

UNIVERSIDAD NACIONAL DEL LITORAL - FICH
PROYECTO FINAL DE CARRERA – ANALISTA INFORMÁTICA APLICADA

Manual de API

Este manual detalla la implementación de la API (Application Programming Interface) del proyecto SAPPCon. Esta API corresponde al backend del desarrollo.

General

Esta API aplica los fundamentos **API-REST** y está implementada con un modelo de ejecución **serverless**:



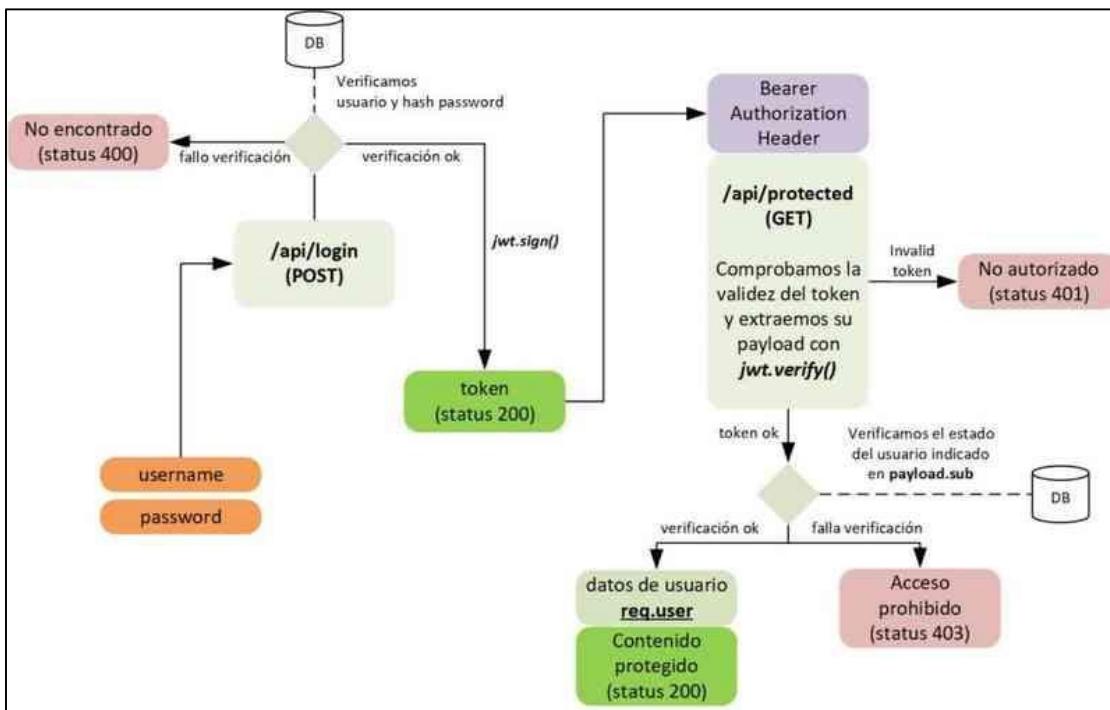
- Utiliza métodos de http específicos por endpoint: **GET** (cuando se obtienen datos), **POST** (cuando se cargan nuevos datos), **PUT** (cuando se modifican datos), **DELETE** (cuando se borran datos).
- Cada función de la API se traduce a un endpoint.
- Las funciones son lógicamente sencillas, minimizando el tiempo de respuesta/ejecución. Las funciones son independientes entre sí, es decir no necesitan consumir de otra función para funcionar.
- Emplea códigos de estado http en los response.

La URL base de la API es la siguiente:

URL: <https://sappcon.site:3000>

Método de Autenticación

La autenticación a los endpoints de la API se realiza mediante un bearer token asociado a al usuario y a la sesión de este. Dicho token se envía en los encabezados de la solicitud http.



Modelo: Usuario (/user)

Nuevo Usuario (/new)

Este método permite registrar un nuevo usuario en el sistema. En este método se crean

URL: <https://sappcon.site:3000/user/register>

Categorías y Unidades de Medida predeterminadas para el usuario.

Especificación Request

- Método HTTP: POST
- Headers:
 - **Content-Type:** application/json
 - **Accept:** */*
- Body:

```
{
    "email": "(string) dirección email",
    "password": "(string) contraseña",
    "name": "(string) nombre",
    "surname": "(string) apellido",
    "alias": "(string) alias",
    "address": "(string) dirección"
}
```

UNIVERSIDAD NACIONAL DEL LITORAL - FICH
PROYECTO FINAL DE CARRERA – ANALISTA INFORMÁTICA APLICADA

Especificación Response

- 200 – OK

```
{  
    "token": "(string) token de autenticación"  
}
```

- 418 – Error al crear el usuario

```
{  
    "message": "Error al crear el usuario.",  
    "messageinfo": "Ya existe un usuario con el email  
proporcionado"  
}
```

- 500 – Error en el servidor

```
{  
    "message": "Error en el servidor.",  
    "messageinfo": "Mensaje de error del servidor  
proporcionado por try/catch."  
}
```

Iniciar sesión (/login)

Este método permite iniciar sesión para un usuario en el sistema.

URL: <https://sappcon.site:3000/user/login>

Especificación Request

- Método HTTP: POST
- Headers:
 - **Content-Type:** application/json
 - **Accept:** */*
- Body:

UNIVERSIDAD NACIONAL DEL LITORAL - FICH
PROYECTO FINAL DE CARRERA – ANALISTA INFORMÁTICA APLICADA

```
{  
    "email": "(string) dirección email",  
    "password": "(string) contraseña"  
}
```

Especificación Response

- 200 – OK

```
{  
    "token": "(string) token de autenticación"  
}
```

- 404 – El usuario no existe

```
{  
    "message": "El usuario no existe.",  
    "messageinfo": "El email proporcionado no está registrado."  
}
```

- 403 – Contraseña incorrecta

```
{  
    "message": "Contraseña incorrecta",  
    "messageinfo": "La contraseña proporcionada no es correcta."  
}
```

- 500 – Error en el servidor

```
{  
    "message": "Error en el servidor.",  
    "messageinfo": "Mensaje de error del servidor proporcionado por try/catch."  
}
```

Cierre sesión (/logout)

Este método permite cerrar la sesión de un usuario en el sistema. Invalida el token de sesión

URL: <https://sappcon.site:3000/user/logout>

asegurando de que a futuro este no sea mal utilizado.

Especificación Request

- **Método HTTP:** POST
- **Headers:**
 - **Content-Type:** application/json
 - **Accept:** */*
 - **Authorization:** "Bearer " + token sesión

Especificación Response

- 200 – OK

```
{  
    "message": "Logout exitoso",  
    "messageinfo": "Token añadido a la lista negra."  
}
```

- 401 – No hay token

```
{  
    "message": "No hay token.",  
    "messageinfo": "No se proporcionó un token de sesión."  
}
```

- 500 – Error en el logout

```
{  
    "message": "Error en el logout.",  
    "messageinfo": "Mensaje de error del servidor  
proporcionado por try/catch de la inserción en la tabla de  
lista negra."  
}
```

- 500 – Error en el servidor

```
{  
    "message": "Error en el servidor.",  
    "messageinfo": "Mensaje de error del servidor  
proporcionado por try/catch."  
}
```

Obtener datos del usuario

Este método permite los datos del usuario que inició sesión. Para obtener los datos del usuario

URL: <https://sappcon.site:3000/user/info>

utiliza el token de autenticación.

Especificación Request

- **Método HTTP:** GET
- **Headers:**
 - **Content-Type:** application/json
 - **Accept:** */*
 - **Authorization:** "Bearer " + token sesión

Especificación Response

- 200 – OK

```
{  
    "message": "Usuario obtenido exitosamente.",  
    "user": {  
        "_id": "(string) id de usuario",  
        "email": "(string) mail usuario",  
        "password": "(string) contraseña cifrada",  
        "name": "(string) nombre usuario",  
        "surname": "(string) apellido",  
        "alias": "(string) alias",  
        "address": "(string) dirección",  
        "__v": 0  
    }  
}
```

- 401 – No autorizado

```
{  
    "message": "No autorizado.",  
    "messageinfo": "No se ha proporcionado token."  
}
```

- 404 – Usuario no encontrado

```
{  
    "message": "Usuario no encontrado.",  
    "messageinfo": "No se encontró información del usuario en  
la base de datos."  
}
```

UNIVERSIDAD NACIONAL DEL LITORAL - FICH
PROYECTO FINAL DE CARRERA – ANALISTA INFORMÁTICA APLICADA

- 500 – Error en el servidor

```
{  
    "message": "Error en el servidor.",  
    "messageinfo": "Mensaje de error del servidor  
proporcionado por try/catch."  
}
```

Modificar datos del usuario (/modify)

Este método permite modificar datos de un usuario en el sistema.

URL: <https://sappcon.site:3000/user/modify>

Especificación Request

- **Método HTTP:** PUT
- **Headers:**
 - **Content-Type:** application/json
 - **Accept:** */*
 - **Authorization:** "Bearer " + token sesión
- **Body:** los tokens que se envían en el JSON son opcionales, sino se envían, ese dato no se actualiza.

```
{  
    "email": "(string) dirección email",  
    "password": "(string) contraseña",  
    "name": "(string) nombre",  
    "surname": "(string) apellido",  
    "alias": "(string) alias",  
    "address": "(string) dirección"  
}
```

Especificación Response

- 200 – OK

UNIVERSIDAD NACIONAL DEL LITORAL - FICH
PROYECTO FINAL DE CARRERA – ANALISTA INFORMÁTICA APLICADA

```
{  
    "message": "Usuario actualizado.",  
    "user": {  
        "_id": "(string) id de usuario",  
        "email": "(string) mail usuario",  
        "password": "(string) contraseña cifrada",  
        "name": "(string) nombre usuario",  
        "surname": "(string) apellido",  
        "alias": "(string) alias",  
        "address": "(string) dirección",  
        "__v": 0  
    }  
}
```

- 400 – El email ya está en uso

```
{  
    "message": "El email ya está en uso.",  
    "messageinfo": "Por favor, utiliza otro email."  
}
```

- 401 – Usuario no autorizado

```
{  
    "message": "Usuario no autorizado.",  
    "messageinfo": "No se ha proporcionado token válido para el usuario."  
}
```

- 404 – Usuario no encontrado

```
{  
    "message": "Usuario no encontrado.",  
    "messageinfo": "No se ha encontrado un usuario con el id proporcionado."  
}
```

- 500 – Error en el servidor

```
{  
    "message": "Error en el servidor.",  
    "messageinfo": "Mensaje de error del servidor proporcionado por try/catch."  
}
```

UNIVERSIDAD NACIONAL DEL LITORAL - FICH
PROYECTO FINAL DE CARRERA – ANALISTA INFORMÁTICA APLICADA

Reestablecer contraseña (*/forgotPassword*)

Este método permite enviar un email para recuperar la contraseña olvidada.

URL: <https://sappcon.site:3000/user/forgotPassword>

Especificación Request

- **Método HTTP:** PUT
- **Headers:**
 - **Content-Type:** application/json
 - **Accept:** */*
- **Body:**

```
{  
    "email": "(string) dirección email"  
}
```

Especificación Response

- 200 – OK

```
{  
    "message": "Correo enviado.",  
    "messageinfo": "Correo de restablecimiento de contraseña  
enviado con éxito."  
}
```

- 404 – El usuario no existe

```
{  
    "message": "El usuario no existe.",  
    "messageinfo": "El email proporcionado no está  
registrado."  
}
```

- 500 – Error en el servidor

```
{  
    "message": "Error en el servidor.",  
    "messageinfo": "Mensaje de error del servidor  
proporcionado por try/catch."  
}
```

UNIVERSIDAD NACIONAL DEL LITORAL - FICH
PROYECTO FINAL DE CARRERA – ANALISTA INFORMÁTICA APLICADA

Modelo: Categoría (/category)

Nueva Categoría (/new)

Este método permite registrar una nueva categoría en el sistema.

URL: <https://sappcon.site:3000/category/new>

Especificación Request

- **Método HTTP:** POST
- **Headers:**
 - **Content-Type:** application/json
 - **Accept:** */*
 - **Authorization:** "Bearer " + token sesión
- **Body:**

```
{  
    "name": "(string) nombre de la categoría"  
}
```

Especificación Response

- 200 – OK

```
{  
    "message": "Categoría creada.",  
    "messageinfo": {  
        "user_id": "(string) id del usuario",  
        "name": "(string) nombre de la categoría",  
        "_id": "(string) id de la categoría",  
        "createdAt": "(datetime) fecha creación",  
        "updatedAt": "(datetime) fecha actualización",  
        "__v": 0  
    }  
}
```

- 401 – Usuario no autorizado

```
{  
    "message": "Usuario no autorizado.",  
    "messageinfo": "No se ha proporcionado token válido para  
el usuario."  
}
```

UNIVERSIDAD NACIONAL DEL LITORAL - FICH
PROYECTO FINAL DE CARRERA – ANALISTA INFORMÁTICA APLICADA

- 500 – Error en el servidor

```
{  
    "message": "Error en el servidor.",  
    "messageinfo": "Mensaje de error del servidor  
proporcionado por try/catch."  
}
```

Obtener Categorías

Este método permite obtener todas las categorías que tiene registradas el usuario

URL: <https://sappcon.site:3000/category>

Especificación Request

- **Método HTTP:** GET
- **Headers:**
 - **Content-Type:** application/json
 - **Accept:** */*
 - **Authorization:** "Bearer " + token sesión

Especificación Response

- 200 – OK

```
[  
    {  
        "_id": "(string) id de categoría",  
        "user_id": "(string) id de usuario",  
        "name": "(string) nombre categoría",  
        "createdAt": "(datetime) fecha creación",  
        "updatedAt": "(datetime) fecha actualización",  
        "__v": 0  
    },  
    ...  
    {  
        "_id": "(string) id de categoría ",  
        "user_id": "(string) id de usuario",  
        "name": "(string) nombre categoría",  
        "createdAt": "(datetime) fecha creación",  
        "updatedAt": "(datetime) fecha actualización"  
        "__v": 0  
    }  
]
```

UNIVERSIDAD NACIONAL DEL LITORAL - FICH
PROYECTO FINAL DE CARRERA – ANALISTA INFORMÁTICA APLICADA

- 401 – Usuario no autorizado

```
{  
    "message": "Usuario no autorizado.",  
    "messageinfo": "No se ha proporcionado token válido para  
el usuario."  
}
```

- 500 – Error en el servidor

```
{  
    "message": "Error en el servidor.",  
    "messageinfo": "Mensaje de error del servidor  
proporcionado por try/catch."  
}
```

Obtener Categoria por ID (/{{id de Categoría}})

Este método permite obtener el detalle de una categoría a partir del id de esta.

URL: <https://sappcon.site:3000/category/{{idCategoria}}>

Especificación Request

- **Método HTTP:** GET
- **Headers:**
 - **Content-Type:** application/json
 - **Accept:** */*
 - **Authorization:** “Bearer ” + token sesión

Especificación Response

- 200 – OK

```
{  
    "user_id": "(string) id del usuario",  
    "name": "(string) nombre de la categoría",  
    "_id": "(string) id de la categoría",  
    "createdAt": "(datetime) fecha creación",  
    "updatedAt": "(datetime) fecha actualización",  
    "__v": 0  
}
```

UNIVERSIDAD NACIONAL DEL LITORAL - FICH
PROYECTO FINAL DE CARRERA – ANALISTA INFORMÁTICA APLICADA

- 401 – Usuario no autorizado

```
{  
    "message": "Usuario no autorizado.",  
    "messageinfo": "No se ha proporcionado token válido para  
el usuario."  
}
```

- 404 – Categoría no encontrada

```
{  
    "message": "Categoría no encontrada.",  
    "messageinfo": "No se ha encontrado la categoría con el  
id proporcionado."  
}
```

- 500 – Error en el servidor

```
{  
    "message": "Error en el servidor.",  
    "messageinfo": "Mensaje de error del servidor  
proporcionado por try/catch."  
}
```

Actualizar Categoría (/modify/{id de Categoría})

Este método permite modificar el nombre de una categoría a partir del id de esta.

URL: <https://sappcon.site:3000/category/modify/{idCategoria}>

Especificación Request

- **Método HTTP:** PUT
- **Headers:**
 - **Content-Type:** application/json
 - **Accept:** */*
 - **Authorization:** “Bearer ” + token sesión
- **Body:**

```
{  
    "name": "(string) nombre de la categoría"  
}
```

UNIVERSIDAD NACIONAL DEL LITORAL - FICH
PROYECTO FINAL DE CARRERA – ANALISTA INFORMÁTICA APLICADA

Especificación Response

- 200 – OK

```
{  
    "message": "Categoría actualizada.",  
    "messageinfo": {  
        "user_id": "(string) id del usuario",  
        "name": "(string) nombre de la categoría",  
        "_id": "(string) id de la categoría",  
        "createdAt": "(datetime) fecha creación",  
        "updatedAt": "(datetime) fecha actualización",  
        "__v": 0  
    }  
}
```

- 401 – Usuario no autorizado

```
{  
    "message": "Usuario no autorizado.",  
    "messageinfo": "No se ha proporcionado token válido para  
el usuario."  
}
```

- 404 – Categoría no encontrada

```
{  
    "message": "Categoría no encontrada.",  
    "messageinfo": "No se ha encontrado la categoría con el  
id proporcionado."  
}
```

- 500 – Error en el servidor

```
{  
    "message": "Error en el servidor.",  
    "messageinfo": "Mensaje de error del servidor  
proporcionado por try/catch."  
}
```

UNIVERSIDAD NACIONAL DEL LITORAL - FICH
PROYECTO FINAL DE CARRERA – ANALISTA INFORMÁTICA APLICADA

Eliminar Categoría (/delete/{id de categoría})

Este método permite eliminar una categoría a partir del id de esta.

URL: <https://sappcon.site:3000/category/delete/{idCategoria}>

Especificación Request

- **Método HTTP:** **DELETE**
- **Headers:**
 - **Content-Type:** application/json
 - **Accept:** */*
 - **Authorization:** "Bearer " + token sesión

Especificación Response

- 200 – OK

```
{  
    "message": "Categoría eliminada.",  
    "messageinfo": "Category id: {idCategoria}"  
}
```

- 401 – Usuario no autorizado

```
{  
    "message": "Usuario no autorizado.",  
    "messageinfo": "No se ha proporcionado token válido para  
el usuario."  
}
```

- 404 – Categoría no encontrada

```
{  
    "message": "Categoría no encontrada.",  
    "messageinfo": "No se ha encontrado la categoría con el  
id proporcionado."  
}
```

- 400 – Categoría no eliminada

```
{  
    "message": "Categoría no eliminada.",  
    "messageinfo": "No se puede eliminar la categoría porque  
tiene servicios asociados."  
}
```

UNIVERSIDAD NACIONAL DEL LITORAL - FICH
PROYECTO FINAL DE CARRERA – ANALISTA INFORMÁTICA APLICADA

- 500 – Error en el servidor

```
{  
    "message": "Error en el servidor.",  
    "messageinfo": "Mensaje de error del servidor  
proporcionado por try/catch."  
}
```

UNIVERSIDAD NACIONAL DEL LITORAL - FICH
PROYECTO FINAL DE CARRERA – ANALISTA INFORMÁTICA APLICADA

Modelo: Unidad de Medida (/measureUnit)

Nueva Unidad de Medida (/new)

Este método permite registrar una nueva unidad de medida en el sistema.

URL: <https://sappcon.site:3000/measureUnit/new>

Especificación Request

- **Método HTTP:** POST
- **Headers:**
 - **Content-Type:** application/json
 - **Accept:** */*
 - **Authorization:** "Bearer " + token sesión
- **Body:**

```
{  
    "name": "(string) nombre de la unidad de medida"  
}
```

Especificación Response

- 200 – OK

```
{  
    "message": "Unidad de medida creada.",  
    "messageinfo": {  
        "user_id": "(string) id del usuario",  
        "name": "(string) nombre de la unidad",  
        "_id": "(string) id de la unidad",  
        "__v": 0  
    }  
}
```

- 401 – Usuario no autorizado

```
{  
    "message": "Usuario no autorizado.",  
    "messageinfo": "No se ha proporcionado token válido para  
el usuario."  
}
```

- 500 – Error en el servidor

```
{  
    "message": "Error en el servidor.",  
    "messageinfo": "Mensaje de error del servidor  
proporcionado por try/catch."  
}
```

Obtener Unidades de Medida

Este método permite obtener todas las categorías que tiene registradas el usuario

URL: <https://sappcon.site:3000/measureUnit>

Especificación Request

- **Método HTTP:** GET
- **Headers:**
 - **Content-Type:** application/json
 - **Accept:** */*
 - **Authorization:** "Bearer " + token sesión

Especificación Response

- 200 – OK

```
[  
  {  
    "_id": "(string) id de unidad de medida",  
    "user_id": "(string) id de usuario",  
    "name": "(string) nombre de unidad",  
    "__v": 0  
  },  
  ...  
  {  
    "_id": "(string) id de unidad de medida",  
    "user_id": "(string) id de usuario",  
    "name": "(string) nombre de unidad",  
    "__v": 0  
  }  
]
```

- 401 – Usuario no autorizado

```
{  
  "message": "Usuario no autorizado.",  
  "messageinfo": "No se ha proporcionado token válido para  
el usuario."  
}
```

- 500 – Error en el servidor

```
{  
  "message": "Error en el servidor.",  
  "messageinfo": "Mensaje de error del servidor  
proporcionado por try/catch."  
}
```

UNIVERSIDAD NACIONAL DEL LITORAL - FICH
PROYECTO FINAL DE CARRERA – ANALISTA INFORMÁTICA APLICADA

Obtener Unidad de Medida por ID (/{{id de Unidad de Medida}})

Este método permite obtener el detalle de una unidad de medida a partir del id de esta.

URL: <https://sappcon.site:3000/measureUnit/{idUnidadMedida}>

Especificación Request

- Método HTTP: GET
- Headers:
 - Content-Type: application/json
 - Accept: */*
 - Authorization: "Bearer " + token sesión

Especificación Response

- 200 – OK

```
{  
    "_id": "(string) id de unidad de medida",  
    "user_id": "(string) id de usuario",  
    "name": "(string) nombre de unidad",  
    "__v": 0  
}
```

- 401 – Usuario no autorizado

```
{  
    "message": "Usuario no autorizado.",  
    "messageinfo": "No se ha proporcionado token válido para  
el usuario."  
}
```

- 404 – Unidad de Medida no encontrada

```
{  
    "message": "Unidad de Medida no encontrada.",  
    "messageinfo": "No se ha encontrado la unidad de medida  
con el id proporcionado."  
}
```

- 500 – Error en el servidor

```
{  
    "message": "Error en el servidor.",  
    "messageinfo": "Mensaje de error del servidor  
proporcionado por try/catch."  
}
```

UNIVERSIDAD NACIONAL DEL LITORAL - FICH
PROYECTO FINAL DE CARRERA – ANALISTA INFORMÁTICA APLICADA

Actualizar Unidad de Medida (/modify/{ id de Unidad de Medida})

Este método permite modificar el nombre de una unidad de medida a partir del id de esta.

URL: <https://sappcon.site:3000/measureUnit/modify/{idUnidadMedida}>

Especificación Request

- Método HTTP: **PUT**
- Headers:
 - **Content-Type:** application/json
 - **Accept:** */*
 - **Authorization:** "Bearer " + token sesión
- Body:

```
{  
    "name": "(string) nombre de la unidad"  
}
```

Especificación Response

- 200 – OK

```
{  
    "message": "Unidad de Medida actualizada.",  
    "messageinfo": {  
        "user_id": "(string) id del usuario",  
        "name": "(string) nombre de la unidad",  
        "_id": "(string) id de la unidad",  
        "__v": 0  
    }  
}
```

- 401 – Usuario no autorizado

```
{  
    "message": "Usuario no autorizado.",  
    "messageinfo": "No se ha proporcionado token válido para  
el usuario."  
}
```

- 404 – Unidad de Medida no encontrada

```
{  
    "message": "Unidad de Medida no encontrada.",  
    "messageinfo": "No se ha encontrado la unidad de medida  
con el id proporcionado."  
}
```

UNIVERSIDAD NACIONAL DEL LITORAL - FICH
PROYECTO FINAL DE CARRERA – ANALISTA INFORMÁTICA APLICADA

- 500 – Error en el servidor

```
{  
    "message": "Error en el servidor.",  
    "messageinfo": "Mensaje de error del servidor  
proporcionado por try/catch."  
}
```

Eliminar Categoría (/delete/{id de Unidad de medida})

Este método permite eliminar una unidad de medida a partir del id de esta.

URL: <https://sappcon.site:3000/measureUnit/delete/{idUnidadMedida}>

Especificación Request

- **Método HTTP: DELETE**
- **Headers:**
 - **Content-Type:** application/json
 - **Accept:** */*
 - **Authorization:** “Bearer ” + token sesión

Especificación Response

- 200 – OK

```
{  
    "message": "Unidad de Medida eliminada.",  
    "messageinfo": "Measure Unit id: {idUnidadMedida}"  
}
```

- 401 – Usuario no autorizado

```
{  
    "message": "Usuario no autorizado.",  
    "messageinfo": "No se ha proporcionado token válido para  
el usuario."  
}
```

- 404 – Unidad de Medida no encontrada

```
{  
    "message": "Unidad de medida no encontrada.",  
    "messageinfo": "No se ha encontrado la unidad de medida  
con el id proporcionado."  
}
```

UNIVERSIDAD NACIONAL DEL LITORAL - FICH
PROYECTO FINAL DE CARRERA – ANALISTA INFORMÁTICA APLICADA

- 400 – Categoría no eliminada

```
{  
    "message": "Unidad de medida no eliminada.",  
    "messageinfo": "No se puede eliminar la unidad de medida  
porque tiene servicios asociados."  
}
```

- 500 – Error en el servidor

```
{  
    "message": "Error en el servidor.",  
    "messageinfo": "Mensaje de error del servidor  
proporcionado por try/catch."  
}
```

UNIVERSIDAD NACIONAL DEL LITORAL - FICH
PROYECTO FINAL DE CARRERA – ANALISTA INFORMÁTICA APLICADA

Modelo: Cliente (/customer)

Nuevo Cliente (/new)

Este método permite registrar un nuevo cliente del usuario.

URL: <https://sappcon.site:3000/customer/new>

Especificación Request

- **Método HTTP: POST**
- **Headers:**
 - **Content-Type:** application/json
 - **Accept:** */*
 - **Authorization:** "Bearer " + token sesión
- **Body:**

```
{  
    "name": "(string) nombre cliente",  
    "surname": "(string) apellido cliente",  
    "alias": "(string) alias cliente",  
    "address": "(string) dirección cliente",  
    "phone": "(number) teléfono cliente",  
    "email": "(string) email cliente"  
}
```

Especificación Response

- 200 – OK

```
{  
    "message": "Cliente creado.",  
    "messageinfo": {  
        "user_id": "(string) id de usuario",  
        "name": "(string) nombre cliente",  
        "surname": "(string) apellido cliente",  
        "alias": "(string) alias cliente",  
        "address": "(string) dirección cliente",  
        "phone": "(number) teléfono cliente",  
        "email": "(string) email cliente"  
        "_id": "(string) id de cliente",  
        "createdAt": "(datetime) fecha de creación",  
        "updatedAt": "(datetime) fecha de modificación",  
        "__v": 0  
    }  
}
```

UNIVERSIDAD NACIONAL DEL LITORAL - FICH
PROYECTO FINAL DE CARRERA – ANALISTA INFORMÁTICA APLICADA

- 401 – Usuario no autorizado

```
{  
    "message": "Usuario no autorizado.",  
    "messageinfo": "No se ha proporcionado token válido para  
el usuario."  
}
```

- 500 – Error en el servidor

```
{  
    "message": "Error en el servidor.",  
    "messageinfo": "Mensaje de error del servidor  
proporcionado por try/catch."  
}
```

Obtener Clientes

Este método permite obtener todos los clientes que tiene registrados el usuario.

URL: <https://sappcon.site:3000/customer>

Especificación Request

- Método HTTP: GET
- Headers:
 - **Content-Type:** application/json
 - **Accept:** */*
 - **Authorization:** “Bearer ” + token sesión

Especificación Response

- 200 – OK

UNIVERSIDAD NACIONAL DEL LITORAL - FICH
PROYECTO FINAL DE CARRERA – ANALISTA INFORMÁTICA APLICADA

```
[  
  {  
    "_id": "(string) id del cliente",  
    "user_id": "(string) id del usuario",  
    "name": "(string) nombre cliente",  
    "surname": "(string) apellido cliente",  
    "alias": "(string) alias cliente",  
    "address": "(string) dirección cliente",  
    "phone": "(number) teléfono cliente",  
    "email": "(string) email cliente"  
    "createdAt": "(datetime) fecha de creación",  
    "updatedAt": "(datetime) fecha de modificación",  
    "__v": 0  
  },  
  ...  
  {  
    "_id": "(string) id del cliente",  
    "user_id": "(string) id del usuario",  
    "name": "(string) nombre cliente",  
    "surname": "(string) apellido cliente",  
    "alias": "(string) alias cliente",  
    "address": "(string) dirección cliente",  
    "phone": "(number) teléfono cliente",  
    "email": "(string) email cliente"  
    "createdAt": "(datetime) fecha de creación",  
    "updatedAt": "(datetime) fecha de modificación",  
    "__v": 0  
  }  
]
```

- 401 – Usuario no autorizado

```
{  
  "message": "Usuario no autorizado.",  
  "messageinfo": "No se ha proporcionado token válido para  
el usuario."  
}
```

- 500 – Error en el servidor

```
{  
  "message": "Error en el servidor.",  
  "messageinfo": "Mensaje de error del servidor  
proporcionado por try/catch."  
}
```

UNIVERSIDAD NACIONAL DEL LITORAL - FICH
PROYECTO FINAL DE CARRERA – ANALISTA INFORMÁTICA APLICADA

Obtener Cliente por ID (/{{id del Cliente}})

Este método permite obtener un cliente registrado a partir del id del mismo.

URL: <https://sappcon.site:3000/customer/{{id del Cliente}}>

Especificación Request

- **Método HTTP:** GET
- **Headers:**
 - **Content-Type:** application/json
 - **Accept:** */*
 - **Authorization:** "Bearer " + token sesión

Especificación Response

- 200 – OK

```
{  
    "_id": "(string) id del cliente",  
    "user_id": "(string) id del usuario",  
    "name": "(string) nombre cliente",  
    "surname": "(string) apellido cliente",  
    "alias": "(string) alias cliente",  
    "address": "(string) dirección cliente",  
    "phone": "(number) teléfono cliente",  
    "email": "(string) email cliente"  
    "createdAt": "(datetime) fecha de creación",  
    "updatedAt": "(datetime) fecha de modificación",  
    "__v": 0  
}
```

- 401 – Usuario no autorizado

```
{  
    "message": "Usuario no autorizado.",  
    "messageinfo": "No se ha proporcionado token válido para  
el usuario."  
}
```

- 404 – Cliente no encontrado

```
{  
    "message": "Cliente no encontrado.",  
    "messageinfo": "No se ha encontrado el cliente con el id  
proporcionado."  
}
```

UNIVERSIDAD NACIONAL DEL LITORAL - FICH
PROYECTO FINAL DE CARRERA – ANALISTA INFORMÁTICA APLICADA

- 500 – Error en el servidor

```
{  
    "message": "Error en el servidor.",  
    "messageinfo": "Mensaje de error del servidor  
proporcionado por try/catch."  
}
```

Actualizar Cliente (/modify)

Este método permite modificar un cliente, identificándolo a través de su id enviado como

URL: <https://sappcon.site:3000/customer/modify/{id del Cliente}>

parámetro.

Especificación Request

- Método HTTP: **PUT**
- Headers:
 - **Content-Type:** application/json
 - **Accept:** */*
 - **Authorization:** "Bearer " + token sesión
- Body: solo es suficiente mandar el token y valor a modificar.

```
{  
    "name": "(string) nuevo nombre del cliente",  
    "address": "(string) nueva dirección cliente"  
}
```

Especificación Response

- 200 – OK

UNIVERSIDAD NACIONAL DEL LITORAL - FICH
PROYECTO FINAL DE CARRERA – ANALISTA INFORMÁTICA APLICADA

```
{  
    "message": "Cliente actualizado.",  
    "messageinfo": {  
        "_id": "(string) id del cliente",  
        "user_id": "(string) id del usuario",  
        "name": "(string) nombre cliente",  
        "surname": "(string) apellido cliente",  
        "alias": "(string) alias cliente",  
        "address": "(string) dirección cliente",  
        "phone": "(number) teléfono cliente",  
        "email": "(string) email cliente"  
        "createdAt": "(datetime) fecha de creación",  
        "updatedAt": "(datetime) fecha de modificación",  
        "__v": 0  
    }  
}
```

- 401 – Usuario no autorizado

```
{  
    "message": "Usuario no autorizado.",  
    "messageinfo": "No se ha proporcionado token válido para  
el usuario."  
}
```

- 404 – Cliente no encontrado

```
{  
    "message": "Cliente no encontrado.",  
    "messageinfo": "No se ha encontrado el cliente con el id  
proporcionado."  
}
```

- 500 – Error en el servidor

```
{  
    "message": "Error en el servidor.",  
    "messageinfo": "Mensaje de error del servidor  
proporcionado por try/catch."  
}
```

UNIVERSIDAD NACIONAL DEL LITORAL - FICH
PROYECTO FINAL DE CARRERA – ANALISTA INFORMÁTICA APLICADA

Eliminar Cliente (/delete)

Este método permite eliminar un cliente, identificándolo a través de su id enviado como parámetro.

URL: <https://sappcon.site:3000/customer/delete/{id del Cliente}>

Especificación Request

- **Método HTTP: DELETE**
- **Headers:**
 - **Content-Type:** application/json
 - **Accept:** */*
 - **Authorization:** "Bearer " + token sesión

Especificación Response

- 200 – OK

```
{  
    "message": "Cliente eliminado.",  
    "messageinfo": {  
        "_id": "(string) id del cliente",  
        "user_id": "(string) id del usuario",  
        "name": "(string) nombre cliente",  
        "surname": "(string) apellido cliente",  
        "alias": "(string) alias cliente",  
        "address": "(string) dirección cliente",  
        "phone": "(number) teléfono cliente",  
        "email": "(string) email cliente"  
        "createdAt": "(datetime) fecha de creación",  
        "updatedAt": "(datetime) fecha de modificación",  
        "__v": 0  
    }  
}
```

- 401 – Usuario no autorizado

```
{  
    "message": "Usuario no autorizado.",  
    "messageinfo": "No se ha proporcionado token válido para el usuario."  
}
```

- 404 – Cliente no encontrado

```
{  
    "message": "Cliente no encontrado.",  
    "messageinfo": "No se ha encontrado el cliente con el id proporcionado."  
}
```

UNIVERSIDAD NACIONAL DEL LITORAL - FICH
PROYECTO FINAL DE CARRERA – ANALISTA INFORMÁTICA APLICADA

- 500 – Error en el servidor

```
{  
    "message": "Error en el servidor.",  
    "messageinfo": "Mensaje de error del servidor  
proporcionado por try/catch."  
}
```

UNIVERSIDAD NACIONAL DEL LITORAL - FICH
PROYECTO FINAL DE CARRERA – ANALISTA INFORMÁTICA APLICADA

Modelo: Servicio (/service)

Nuevo Servicio (/new)

Este método permite registrar un nuevo servicio del usuario.

URL: <https://sappcon.site:3000/service/new>

Especificación Request

- **Método HTTP: POST**
- **Headers:**
 - **Content-Type:** application/json
 - **Accept:** */*
 - **Authorization:** "Bearer " + token sesión
- **Body:**

```
{  
    "measure_unit_id": "(string) id de unidad medida",  
    "category_id": "(string) id de categoría",  
    "name": "(string) nombre del servicio",  
    "description": "(string) descripción ampliada",  
    "cost": (decimal) costo,  
    "price": (decimal) precio  
}
```

Especificación Response

- 200 – OK

```
{  
    "message": "Servicio creado.",  
    "messageinfo": {  
        "user_id": "(string) id de usuario",  
        "measure_unit_id": "(string) id de u. medida",  
        "category_id": "(string) id de categoría",  
        "name": "(string) nombre del servicio",  
        "description": "(string) descripción ampliada",  
        "cost": (decimal) costo,  
        "price": (decimal) precio,  
        "_id": "(string) id del servicio",  
        "createdAt": "(datetime) fecha creación.",  
        "updatedAt": "(datetime) fecha modificación.",  
        "__v": 0  
    }  
}
```

UNIVERSIDAD NACIONAL DEL LITORAL - FICH
PROYECTO FINAL DE CARRERA – ANALISTA INFORMÁTICA APLICADA

- 401 – Usuario no autorizado

```
{  
    "message": "Usuario no autorizado.",  
    "messageinfo": "No se ha proporcionado token válido para  
el usuario."  
}
```

- 500 – Error en el servidor

```
{  
    "message": "Error en el servidor.",  
    "messageinfo": "Mensaje de error del servidor  
proporcionado por try/catch."  
}
```

Obtener Servicios

Este método permite obtener todos los servicios que tiene registrados el usuario.

URL: <https://sappcon.site:3000/service>

Especificación Request

- **Método HTTP:** GET
- **Headers:**
 - **Content-Type:** application/json
 - **Accept:** */*
 - **Authorization:** “Bearer ” + token sesión

Especificación Response

- 200 – OK

UNIVERSIDAD NACIONAL DEL LITORAL - FICH
PROYECTO FINAL DE CARRERA – ANALISTA INFORMÁTICA APLICADA

```
[  
  {  
    "user_id": "(string) id de usuario",  
    "measure_unit_id": "(string) id de u. medida",  
    "category_id": "(string) id de categoría",  
    "name": "(string) nombre del servicio",  
    "description": "(string) descripción ampliada",  
    "cost": (decimal) costo,  
    "price": (decimal) precio,  
    "_id": "(string) id del servicio",  
    "createdAt": "(datetime) fecha creación.",  
    "updatedAt": "(datetime) fecha modificación.",  
    "__v": 0  
  },  
  ...  
  {  
    "user_id": "(string) id de usuario",  
    "measure_unit_id": "(string) id de u. medida",  
    "category_id": "(string) id de categoría",  
    "name": "(string) nombre del servicio",  
    "description": "(string) descripción ampliada",  
    "cost": (decimal) costo,  
    "price": (decimal) precio,  
    "_id": "(string) id del servicio",  
    "createdAt": "(datetime) fecha creación.",  
    "updatedAt": "(datetime) fecha modificación.",  
    "__v": 0  
  }  
]
```

- 401 – Usuario no autorizado

```
{  
  "message": "Usuario no autorizado.",  
  "messageinfo": "No se ha proporcionado token válido para  
el usuario."  
}
```

- 500 – Error en el servidor

```
{  
  "message": "Error en el servidor.",  
  "messageinfo": "Mensaje de error del servidor  
proporcionado por try/catch."  
}
```

UNIVERSIDAD NACIONAL DEL LITORAL - FICH
PROYECTO FINAL DE CARRERA – ANALISTA INFORMÁTICA APLICADA

Obtener Servicio por ID (/{{id del Servicio}})

Este método permite obtener un servicio registrado a partir del id del mismo, el cual se pasa

URL: <https://sappcon.site:3000/service/{{id del Servicio}}>

como parámetro en la URL de la petición.

Especificación Request

- Método HTTP: **GET**
- Headers:
 - **Content-Type:** application/json
 - **Accept:** */*
 - **Authorization:** "Bearer " + token sesión

Especificación Response

- 200 – OK

```
{  
    "user_id": "(string) id de usuario",  
    "measure_unit_id": "(string) id de u. medida",  
    "category_id": "(string) id de categoría",  
    "name": "(string) nombre del servicio",  
    "description": "(string) descripción ampliada",  
    "cost": "(decimal) costo",  
    "price": "(decimal) precio",  
    "_id": "(string) id del servicio",  
    "createdAt": "(datetime) fecha creación.",  
    "updatedAt": "(datetime) fecha modificación.",  
    "__v": 0  
}
```

- 401 – Usuario no autorizado

```
{  
    "message": "Usuario no autorizado.",  
    "messageinfo": "No se ha proporcionado token válido para  
el usuario."  
}
```

- 404 – Servicio no encontrado

```
{  
    "message": "Servicio no encontrado.",  
    "messageinfo": "No se ha encontrado el servicio con el id  
proporcionado."  
}
```

UNIVERSIDAD NACIONAL DEL LITORAL - FICH
PROYECTO FINAL DE CARRERA – ANALISTA INFORMÁTICA APLICADA

- 500 – Error en el servidor

```
{  
    "message": "Error en el servidor.",  
    "messageinfo": "Mensaje de error del servidor  
proporcionado por try/catch."  
}
```

Actualizar Servicio (/modify)

Este método permite modificar un servicio, identificándolo a través de su id enviado como

URL: <https://sappcon.site:3000/service/modify/{id del Servicio}>

parámetro de la URL.

Especificación Request

- Método HTTP: **PUT**
- Headers:
 - **Content-Type:** application/json
 - **Accept:** */*
 - **Authorization:** "Bearer " + token sesión
- Body: solo es suficiente mandar el token y valor a modificar.

```
{  
    "name": "(string) nuevo nombre del servicio",  
    "cost": "(decimal) nuevo costo del servicio"  
}
```

Especificación Response

- 200 – OK

UNIVERSIDAD NACIONAL DEL LITORAL - FICH
PROYECTO FINAL DE CARRERA – ANALISTA INFORMÁTICA APLICADA

```
{  
    "message": "Servicio actualizado.",  
    "messageinfo": {  
        "user_id": "(string) id de usuario",  
        "measure_unit_id": "(string) id de u. medida",  
        "category_id": "(string) id de categoría",  
        "name": "(string) nombre del servicio",  
        "description": "(string) descripción ampliada",  
        "cost": "(decimal) costo",  
        "price": "(decimal) precio",  
        "_id": "(string) id del servicio",  
        "createdAt": "(datetime) fecha creación.",  
        "updatedAt": "(datetime) fecha modificación.",  
        "__v": 0  
    }  
}
```

- 401 – Usuario no autorizado

```
{  
    "message": "Usuario no autorizado.",  
    "messageinfo": "No se ha proporcionado token válido para  
el usuario."  
}
```

- 404 – Servicio no encontrado

```
{  
    "message": "Servicio no encontrado.",  
    "messageinfo": "No se ha encontrado el cliente con el id  
proporcionado."  
}
```

- 500 – Error en el servidor

```
{  
    "message": "Error en el servidor.",  
    "messageinfo": "Mensaje de error del servidor  
proporcionado por try/catch."  
}
```

UNIVERSIDAD NACIONAL DEL LITORAL - FICH
PROYECTO FINAL DE CARRERA – ANALISTA INFORMÁTICA APLICADA

Eliminar Servicio (/delete)

Este método permite eliminar un servicio, identificándolo a través de su id enviado como

URL: <https://sappcon.site:3000/service/delete/{id del Servicio}>

parámetro de la URL.

Especificación Request

- Método HTTP: DELETE
- Headers:
 - **Content-Type:** application/json
 - **Accept:** */*
 - **Authorization:** "Bearer " + token sesión

Especificación Response

- 200 – OK

```
{  
    "message": "Servicio eliminado.",  
    "messageinfo": {  
        "user_id": "(string) id de usuario",  
        "measure_unit_id": "(string) id de u. medida",  
        "category_id": "(string) id de categoría",  
        "name": "(string) nombre del servicio",  
        "description": "(string) descripción ampliada",  
        "cost": (decimal) costo,  
        "price": (decimal) precio,  
        "_id": "(string) id del servicio",  
        "createdAt": "(datetime) fecha creación.",  
        "updatedAt": "(datetime) fecha modificación.",  
        "__v": 0  
    }  
}
```

- 401 – Usuario no autorizado

```
{  
    "message": "Usuario no autorizado.",  
    "messageinfo": "No se ha proporcionado token válido para  
el usuario."  
}
```

- 404 – Servicio no encontrado

```
{  
    "message": "Servicio no encontrado.",  
    "messageinfo": "No se ha encontrado el cliente con el id  
proporcionado."  
}
```

UNIVERSIDAD NACIONAL DEL LITORAL - FICH
PROYECTO FINAL DE CARRERA – ANALISTA INFORMÁTICA APLICADA

- 405 – No se puede eliminar el servicio

```
{  
    "message": "No se puede eliminar el servicio.",  
    "messageinfo": "Este servicio está siendo utilizado en una  
    linea de presupuesto."  
}
```

- 500 – Error en el servidor

```
{  
    "message": "Error en el servidor.",  
    "messageinfo": "Mensaje de error del servidor  
    proporcionado por try/catch."  
}
```

UNIVERSIDAD NACIONAL DEL LITORAL - FICH
PROYECTO FINAL DE CARRERA – ANALISTA INFORMÁTICA APLICADA

Modelo: Obra (/building)

Nueva Obra (/new)

Este método permite registrar una nueva obra/construcción del usuario.

URL: <https://sappcon.site:3000/building/new>

Especificación Request

- **Método HTTP: POST**
- **Headers:**
 - **Content-Type:** application/json
 - **Accept:** */*
 - **Authorization:** "Bearer " + token sesión
- **Body:**

```
{  
    "customer_id": "(string) id del cliente/dueño de la  
    obra",  
    "name": "(string) nombre de la obra",  
    "address": "(string) dirección",  
    "description": "(string) descripción ampliada",  
}
```

Especificación Response

- 200 – OK

```
{  
    "message": "Construcción creada.",  
    "messageinfo": {  
        "user_id": "(string) id de usuario",  
        "customer_id": "(string) id del cliente",  
        "name": "(string) nombre de la obra",  
        "address": "(string) dirección de la obra",  
        "description": "(string) descripción ampliada",  
        "_id": "(string) id de la obra",  
        "__v": 0  
    }  
}
```

- 401 – Usuario no autorizado

```
{  
    "message": "Usuario no autorizado.",  
    "messageinfo": "No se ha proporcionado token válido para  
el usuario."  
}
```

UNIVERSIDAD NACIONAL DEL LITORAL - FICH
PROYECTO FINAL DE CARRERA – ANALISTA INFORMÁTICA APLICADA

- 500 – Error en el servidor

```
{  
    "message": "Error en el servidor.",  
    "messageinfo": "Mensaje de error del servidor  
proporcionado por try/catch."  
}
```

Obtener Obras

Este método permite obtener todas las obras que tiene registradas el usuario.

URL: <https://sappcon.site:3000/building>

Especificación Request

- **Método HTTP:** GET
- **Headers:**
 - **Content-Type:** application/json
 - **Accept:** */*
 - **Authorization:** "Bearer " + token sesión

Especificación Response

- 200 – OK

```
[  
    {  
        "_id": "(string) id de obra",  
        "user_id": "(string) id de usuario",  
        "customer_id": "(string) id de cliente",  
        "name": "(string) nombre de la obra",  
        "address": "(string) dirección",  
        "description": "(string) descripción ampliada",  
        "__v": 0  
    },  
    ...  
    {  
        "_id": "(string) id de obra",  
        "user_id": "(string) id de usuario",  
        "customer_id": "(string) id de cliente",  
        "name": "(string) nombre de la obra",  
        "address": "(string) dirección",  
        "description": "(string) descripción ampliada",  
        "__v": 0  
    }  
]
```

UNIVERSIDAD NACIONAL DEL LITORAL - FICH
PROYECTO FINAL DE CARRERA – ANALISTA INFORMÁTICA APLICADA

- 401 – Usuario no autorizado

```
{  
    "message": "Usuario no autorizado.",  
    "messageinfo": "No se ha proporcionado token válido para  
el usuario."  
}
```

- 500 – Error en el servidor

```
{  
    "message": "Error en el servidor.",  
    "messageinfo": "Mensaje de error del servidor  
proporcionado por try/catch."  
}
```

Obtener Obra por ID (/{{id de la Obra}})

Este método permite obtener una obra a partir de su id, el cual se pasa como parámetro en la URL de la petición.

URL: <https://sappcon.site:3000/building/{{id de la Obra}}>

Especificación Request

- Método HTTP: GET
- Headers:
 - **Content-Type:** application/json
 - **Accept:** */*
 - **Authorization:** “Bearer ” + token sesión

Especificación Response

- 200 – OK

```
{  
    "_id": "(string) id de obra",  
    "user_id": "(string) id de usuario",  
    "customer_id": "(string) id de cliente",  
    "name": "(string) nombre de la obra",  
    "address": "(string) dirección",  
    "description": "(string) descripción ampliada",  
    "__v": 0  
}
```

UNIVERSIDAD NACIONAL DEL LITORAL - FICH
PROYECTO FINAL DE CARRERA – ANALISTA INFORMÁTICA APLICADA

- 401 – Usuario no autorizado

```
{  
    "message": "Usuario no autorizado.",  
    "messageinfo": "No se ha proporcionado token válido para  
el usuario."  
}
```

- 404 – Construcción no encontrada

```
{  
    "message": "Construcción no encontrada.",  
    "messageinfo": "No se ha encontrado la construcción con  
el id proporcionado."  
}
```

- 500 – Error en el servidor

```
{  
    "message": "Error en el servidor.",  
    "messageinfo": "Mensaje de error del servidor  
proporcionado por try/catch."  
}
```

Actualizar Obra (/modify)

Este método permite modificar una obra, identificándola a través de su id enviado como parámetro de la URL.

URL: <https://sappcon.site:3000/building/modify/{id de la Obra}>

Especificación Request

- **Método HTTP: PUT**
- **Headers:**
 - **Content-Type:** application/json
 - **Accept:** */*
 - **Authorization:** “Bearer ” + token sesión
- **Body:** solo es suficiente mandar el token y valor a modificar.

```
{  
    "name": "(string) nuevo nombre de la obra",  
    "description": "(string) nueva descripción"  
}
```

Especificación Response

- 200 – OK

UNIVERSIDAD NACIONAL DEL LITORAL - FICH
PROYECTO FINAL DE CARRERA – ANALISTA INFORMÁTICA APLICADA

```
{  
    "message": "Construcción actualizada.",  
    "messageinfo": {  
        "_id": "(string) id de obra",  
        "user_id": "(string) id de usuario",  
        "customer_id": "(string) id de cliente",  
        "name": "(string) nombre de la obra",  
        "address": "(string) dirección",  
        "description": "(string) descripción ampliada",  
        "__v": 0  
    }  
}
```

- 401 – Usuario no autorizado

```
{  
    "message": "Usuario no autorizado.",  
    "messageinfo": "No se ha proporcionado token válido para el usuario."  
}
```

- 404 – Construcción no encontrada

```
{  
    "message": "Construcción no encontrado.",  
    "messageinfo": "No se ha encontrado la construcción con el id  
proporcionado."  
}
```

- 500 – Error en el servidor

```
{  
    "message": "Error en el servidor.",  
    "messageinfo": "Mensaje de error del servidor proporcionado por  
try/catch."  
}
```

Eliminar Obra (/delete)

Este método permite eliminar una obra/construcción, identificándola a través de su id enviado como parámetro de la URL.

URL: <https://sappcon.site:3000/building/delete/{id de la Obra}>

Especificación Request

- Método HTTP: DELETE

UNIVERSIDAD NACIONAL DEL LITORAL - FICH
PROYECTO FINAL DE CARRERA – ANALISTA INFORMÁTICA APLICADA

- Headers:
 - **Content-Type:** application/json
 - **Accept:** */*
 - **Authorization:** "Bearer " + token sesión

Especificación Response

- 200 – OK

```
{  
    "message": "Construcción eliminada.",  
    "messageinfo": {  
        "_id": "(string) id de la obra eliminada",  
        "user_id": "(string) id del usuario",  
        "customer_id": "(string) id del cliente",  
        "name": "(string) nombre de la obra",  
        "address": "(string) dirección de la obra",  
        "description": "(string) descripción ampliada",  
        "__v": 0  
    }  
}
```

- 401 – Usuario no autorizado

```
{  
    "message": "Usuario no autorizado.",  
    "messageinfo": "No se ha proporcionado token válido para el usuario."  
}
```

- 404 – Construcción no encontrada

```
{  
    "message": "Construcción no encontrada.",  
    "messageinfo": "No se ha encontrado la construcción con el id proporcionado."  
}
```

- 400 – Construcción no eliminada

```
{  
    "message": "Construcción no eliminada.",  
    "messageinfo": "No se puede eliminar la construcción porque tiene presupuestos asociados."  
}
```

- 500 – Error en el servidor

```
{  
    "message": "Error en el servidor.",  
    "messageinfo": "Mensaje de error del servidor proporcionado por try/catch."  
}
```

UNIVERSIDAD NACIONAL DEL LITORAL - FICH
PROYECTO FINAL DE CARRERA – ANALISTA INFORMÁTICA APLICADA

Modelo: Presupuesto (/budget)

Nuevo Presupuesto (/new)

Este método permite registrar un nuevo presupuesto, solo a partir de los datos de la cabecera.

URL: <https://sappcon.site:3000/budget/new>

Especificación Request

- **Método HTTP: POST**
- **Headers:**
 - **Content-Type:** application/json
 - **Accept:** */*
 - **Authorization:** "Bearer" + token sesión
- **Body:**

```
{  
    "Building_id": "(string) id de la obra",  
    "status": "(enum) estado",  
    "date": "(date) fecha del presupuesto"  
}
```

**valores enum "status": ["Pendiente", "Aprobado", "Rechazado", "En Construcción", "Finalizado"]

Especificación Response

- 200 – OK

```
{  
    "message": "Presupuesto creado.",  
    "messageinfo": {  
        "user_id": "(string) id del usuario",  
        "building_id": "(string) id de la obra",  
        "customer_id": "(string) id del cliente",  
        "customer_name": "(string) nombre + apellido cliente",  
        "status": "(enum) status",  
        "amount": (decimal) sumatoria de importe,  
        "date": "(date) fecha del presupuesto",  
        "_id": "(string) id del presupuesto",  
        "createdAt": "(datetime) fecha y hora creación",  
        "updatedAt": "(datetime) fecha y hora modificación",  
        "__v": 0  
    }  
}
```

- 401 – Usuario no autorizado

UNIVERSIDAD NACIONAL DEL LITORAL - FICH
PROYECTO FINAL DE CARRERA – ANALISTA INFORMÁTICA APLICADA

```
{  
    "message": "Usuario no autorizado.",  
    "messageinfo": "No se ha proporcionado token válido para el  
usuario."  
}
```

- 500 – Error en el servidor

```
{  
    "message": "Error en el servidor.",  
    "messageinfo": "Mensaje de error del servidor proporcionado por  
try/catch."  
}
```

Nueva Línea de Presupuesto (/newLine)

Este método permite ingresar una nueva línea de detalle a un presupuesto, cada línea de detalle asocia un solo servicio y una cantidad para este, de acuerdo a la unidad de medida del servicio. Y obtiene de la configuración del servicio, el precio unitario.

URL: <https://sappcon.site:3000/budget/newLine>

Especificación Request

- **Método HTTP:** POST
- **Headers:**
 - **Content-Type:** application/json
 - **Accept:** */*
 - **Authorization:** “Bearer ” + token sesión
- **Body:**

```
{  
    "budget_id": "(string) id del presupuesto",  
    "line_no": "(number) número de linea",  
    "service_id": "(string) id del servicio",  
    "quantity": (decimal) cantidad del servicio  
}
```

Especificación Response

- 200 – OK

UNIVERSIDAD NACIONAL DEL LITORAL - FICH
PROYECTO FINAL DE CARRERA – ANALISTA INFORMÁTICA APLICADA

```
{  
    "message": "Línea de presupuesto creada.",  
    "messageinfo": {  
        "user_id": "(string) id del usuario",  
        "budget_id": "(string) id del presupuesto",  
        "line_no": "(number) número de línea",  
        "service_id": "(string) id del servicio",  
        "service_name": "(string) nombre del servicio",  
        "measure_unit_id": "(string) id unidad de medida",  
        "measure_unit_name": "(string) nombre de la unidad de medida",  
        "quantity": "(decimal) cantidad",  
        "price": "(decimal) precio",  
        "amount": "(decimal) importe",  
        "_id": "(string) id de la línea",  
        "createdAt": "(datetime) fecha y hora de creación.",  
        "updatedAt": "(datetime) fecha y hora de modificación.",  
        "__v": 0  
    }  
}
```

- 401 – Usuario no autorizado

```
{  
    "message": "Usuario no autorizado.",  
    "messageinfo": "No se ha proporcionado token válido para el  
usuario."  
}
```

- 404 – Servicio no encontrado

```
{  
    "message": "Servicio no encontrado.",  
    "messageinfo": "El servicio con el ID proporcionado no existe."  
}
```

- 404 – Presupuesto no encontrado

```
{  
    "message": "Presupuesto no encontrado.",  
    "messageinfo": "El presupuesto con el ID proporcionado no  
existe."  
}
```

- 404 – Unidad de medida no encontrada

```
{  
    "message": "Unidad de medida no encontrada.",  
    "messageinfo": "La unidad de medida con el ID proporcionado no  
existe."  
}
```

UNIVERSIDAD NACIONAL DEL LITORAL - FICH
PROYECTO FINAL DE CARRERA – ANALISTA INFORMÁTICA APLICADA

- 500 – Error en el servidor

```
{  
    "message": "Error en el servidor.",  
    "messageinfo": "Mensaje de error del servidor  
proporcionado por try/catch."  
}
```

Obtener Cabeceras de Presupuestos (/headers)

Este método permite obtener solo las cabeceras de todos los presupuestos. Se utiliza para listar los presupuestos sin la necesidad de tener que cargar en memoria presupuestos y sus respectivas líneas.

URL: <https://sappcon.site:3000/budget/headers>

Especificación Request

- Método HTTP: GET
- Headers:
 - **Content-Type:** application/json
 - **Accept:** */*
 - **Authorization:** “Bearer ” + token sesión

Especificación Response

- 200 – OK

```
[  
    {  
        "_id": "(string) id del presupuesto",  
        "building_id": "(string) id de la obra",  
        "customer_id": "(string) id del cliente",  
        "status": "(enum) estado del presupuesto",  
        "amount": "(decimal) importe total",  
        "date": "(date) fecha del presupuesto"  
    },  
    ...  
    {  
        "_id": "(string) id del presupuesto",  
        "building_id": "(string) id de la obra",  
        "customer_id": "(string) id del cliente",  
        "status": "(enum) estado del presupuesto",  
        "amount": "(decimal) importe total",  
        "date": "(date) fecha del presupuesto"  
    }  
]
```

UNIVERSIDAD NACIONAL DEL LITORAL - FICH
PROYECTO FINAL DE CARRERA – ANALISTA INFORMÁTICA APLICADA

- 401 – Usuario no autorizado

```
{  
    "message": "Usuario no autorizado.",  
    "messageinfo": "No se ha proporcionado token válido para el  
    usuario."  
}
```

- 500 – Error en el servidor

```
{  
    "message": "Error en el servidor.",  
    "messageinfo": "Mensaje de error del servidor proporcionado por  
    try/catch."  
}
```

Obtener Presupuesto por ID (/{{id presupuesto}})

Este método retorna un presupuesto, el cual se envía su id por parámetro en la URL de solicitud, detallando sus datos de cabecera y sus líneas en un arreglo.

URL: <https://sappcon.site:3000/budget/{{id del Presupuesto}}>

Especificación Request

- Método HTTP: GET
- Headers:
 - **Content-Type:** application/json
 - **Accept:** */*
 - **Authorization:** “Bearer ” + token sesión

Especificación Response

- 200 – OK

UNIVERSIDAD NACIONAL DEL LITORAL - FICH
PROYECTO FINAL DE CARRERA – ANALISTA INFORMÁTICA APLICADA

```
{  
    "budget": {  
        "_id": "(string) id del presupuesto",  
        "user_id": "(string) id del usuario",  
        "building_id": "(string) id de la obra",  
        "customer_id": "(string) id del cliente",  
        "customer_name": "(string) nombre + apellido del cliente",  
        "status": "(enum) estado del presupuesto",  
        "amount": (decimal) importe del presupuesto,  
        "date": "(date) fecha del presupuesto",  
        "createdAt": "(datetime) fecha y hora creación.",  
        "updatedAt": "(datetime) fecha y hora modificación.",  
        "__v": 0  
    },  
    "budgetLines": [  
        {  
            "user_id": "(string) id del usuario",  
            "budget_id": "(string) id del presupuesto",  
            "line_no": (number) número de linea,  
            "service_id": "(string) id del servicio",  
            "service_name": "(string) nombre del servicio",  
            "measure_unit_id": "(string) id unidad de medida",  
            "measure_unit_name": "(string) nombre de la unidad de medida",  
            "quantity": (decimal) cantidad,  
            "price": (decimal) precio,  
            "amount": (decimal) importe,  
            "_id": "(string) id de la línea",  
            "createdAt": "(datetime) fecha y hora de creación.",  
            "updatedAt": "(datetime) fecha y hora de modificación.",  
            "__v": 0  
        },  
        {  
            "user_id": "(string) id del usuario",  
            "budget_id": "(string) id del presupuesto",  
            "line_no": (number) número de linea,  
            "service_id": "(string) id del servicio",  
            "service_name": "(string) nombre del servicio",  
            "measure_unit_id": "(string) id unidad de medida",  
            "measure_unit_name": "(string) nombre de la unidad de medida",  
            "quantity": (decimal) cantidad,  
            "price": (decimal) precio,  
            "amount": (decimal) importe,  
            "_id": "(string) id de la línea",  
            "createdAt": "(datetime) fecha y hora de creación.",  
            "updatedAt": "(datetime) fecha y hora de modificación.",  
            "__v": 0  
        }  
    ]  
}
```

UNIVERSIDAD NACIONAL DEL LITORAL - FICH
PROYECTO FINAL DE CARRERA – ANALISTA INFORMÁTICA APLICADA

- 401 – Usuario no autorizado

```
{  
    "message": "Usuario no autorizado.",  
    "messageinfo": "No se ha proporcionado token válido para el  
    usuario."  
}
```

- 404 – Presupuesto no encontrado

```
{  
    "message": "Presupuesto no encontrado.",  
    "messageinfo": "El presupuesto con el ID proporcionado no  
    existe."  
}
```

- 500 – Error en el servidor

```
{  
    "message": "Error en el servidor.",  
    "messageinfo": "Mensaje de error del servidor proporcionado por  
    try/catch."  
}
```

Modificar Presupuesto (/modify/{id presupuesto})

Este método permite modificar algún dato de la cabecera del presupuesto, el cual se identifica a través de su id enviado como parámetro de la URL.

URL: <https://sappcon.site:3000/budget/modify/{id del Presupuesto}>

Especificación Request

- Método HTTP: PUT
- Headers:
 - **Content-Type:** application/json
 - **Accept:** */*
 - **Authorization:** “Bearer ” + token sesión
- Body: solo es suficiente mandar el token y valor a modificar.

```
{  
    "status": "(enum) nuevo estado del presupuesto",  
    "date": "(date) nueva fecha del presupuesto"  
}
```

UNIVERSIDAD NACIONAL DEL LITORAL - FICH
PROYECTO FINAL DE CARRERA – ANALISTA INFORMÁTICA APLICADA

Especificación Response

- 200 – OK

```
{  
    "message": "Presupuesto actualizado.",  
    "messageinfo": {  
        "_id": "(string) id del presupuesto",  
        "user_id": "(string) id del usuario",  
        "building_id": "(string) id de la obra",  
        "customer_id": "(string) id del cliente",  
        "customer_name": "(string) nombre + apellido del cliente",  
        "status": "(enum) estado del presupuesto",  
        "amount": "(decimal) importe del presupuesto",  
        "date": "(date) fecha del presupuesto",  
        "createdAt": "(datetime) fecha y hora creación.",  
        "updatedAt": "(datetime) fecha y hora modificación.",  
        "__v": 0  
    }  
}
```

- 401 – Usuario no autorizado

```
{  
    "message": "Construcción no encontrado.",  
    "messageinfo": "No se ha encontrado la construcción con el id proporcionado."  
}
```

- 404 – Presupuesto no encontrado

```
{  
    "message": "Presupuesto no encontrado.",  
    "messageinfo": "El presupuesto con el ID proporcionado no existe."  
}
```

- 500 – Error en el servidor

```
{  
    "message": "Error en el servidor.",  
    "messageinfo": "Mensaje de error del servidor proporcionado por try/catch."  
}
```

Modificar Línea de Presupuesto (`/modify/{id presupuesto}/{id línea}`)

Este método permite modificar algún dato de una línea de presupuesto. La línea a modificar se identifica a través de la combinación de ID de presupuesto e ID de línea como parámetros en la URL.

URL: <https://sappcon.site:3000/budget/modify/{id del Presupuesto}/{id de la Línea}>

Especificación Request

- Método HTTP: **PUT**
- Headers:
 - **Content-Type:** application/json
 - **Accept:** */*
 - **Authorization:** "Bearer " + token sesión
- Body: solo es suficiente mandar el token y valor a modificar.

```
{  
    "quantity": (decimal) nueva cantidad  
}
```

Especificación Response

- 200 – OK

```
{  
    "message": "Línea de Presupuesto actualizada.",  
    "messageinfo": {  
        "user_id": "(string) id del usuario",  
        "budget_id": "(string) id del presupuesto",  
        "line_no": (number) número de linea,  
        "service_id": "(string) id del servicio",  
        "service_name": "(string) nombre del servicio",  
        "measure_unit_id": "(string) id unidad de medida",  
        "measure_unit_name": "(string) nombre de la unidad de medida",  
        "quantity": (decimal) cantidad,  
        "price": (decimal) precio,  
        "amount": (decimal) importe,  
        "_id": "(string) id de la línea",  
        "createdAt": "(datetime) fecha y hora de creación.",  
        "updatedAt": "(datetime) fecha y hora de modificación.",  
        "__v": 0  
    }  
}
```

UNIVERSIDAD NACIONAL DEL LITORAL - FICH
PROYECTO FINAL DE CARRERA – ANALISTA INFORMÁTICA APLICADA

- 401 – Usuario no autorizado

```
{  
    "message": "Construcción no encontrado.",  
    "messageinfo": "No se ha encontrado la construcción con el id  
proporcionado."  
}
```

- 404 – Presupuesto no encontrado

```
{  
    "message": "Presupuesto no encontrado.",  
    "messageinfo": "El presupuesto con el ID proporcionado no existe."  
}
```

- 404 – Línea de Presupuesto no encontrada

```
{  
    "message": "Línea de Presupuesto no encontrada.",  
    "messageinfo": "La linea de presupuesto con el ID proporcionado no  
existe."  
}
```

- 500 – Error en el servidor

```
{  
    "message": "Error en el servidor.",  
    "messageinfo": "Mensaje de error del servidor proporcionado por  
try/catch."  
}
```

Eliminar Presupuesto (/delete/{id presupuesto})

Este método permite eliminar todo un presupuesto (cabecera y líneas). El presupuesto a

URL: <https://sappcon.site:3000/budget/delete/{id del Presupuesto}>

eliminar se identifica pasando su ID como parámetro en la URL.

Especificación Request

- Método HTTP: DELETE
- Headers:
 - **Content-Type:** application/json
 - **Accept:** */*
 - **Authorization:** “Bearer ” + token sesión

UNIVERSIDAD NACIONAL DEL LITORAL - FICH
PROYECTO FINAL DE CARRERA – ANALISTA INFORMÁTICA APLICADA

Especificación Response

- 200 – OK

```
{  
    "message": "Presupuesto eliminado.",  
    "messageinfo": {  
        "_id": "(string) id del presupuesto",  
        "user_id": "(string) id del usuario",  
        "building_id": "(string) id de la obra",  
        "customer_id": "(string) id del cliente",  
        "customer_name": "(string) nombre + apellido del cliente",  
        "status": "(enum) estado del presupuesto",  
        "amount": "(decimal) importe del presupuesto",  
        "date": "(date) fecha del presupuesto",  
        "createdAt": "(datetime) fecha y hora creación.",  
        "updatedAt": "(datetime) fecha y hora modificación.",  
        "__v": 0  
    }  
}
```

- 401 – Usuario no autorizado

```
{  
    "message": "Construcción no encontrado.",  
    "messageinfo": "No se ha encontrado la construcción con el id proporcionado."  
}
```

- 404 – Presupuesto no encontrado

```
{  
    "message": "Presupuesto no encontrado.",  
    "messageinfo": "El presupuesto con el ID proporcionado no existe."  
}
```

- 500 – Error en el servidor

```
{  
    "message": "Error en el servidor.",  
    "messageinfo": "Mensaje de error del servidor proporcionado por try/catch."  
}
```

Eliminar Línea de Presupuesto (/delete/{id presupuesto}/{id línea})

Este método permite eliminar una línea de presupuesto. La línea a eliminar se identifica pasándola como parámetro en la URL.

URL: <https://sappcon.site:3000/budget/delete/{id del Presupuesto}/{id de la Línea}>

Especificación Request

- Método HTTP: **DELETE**
- Headers:
 - **Content-Type:** application/json
 - **Accept:** */*
 - **Authorization:** "Bearer " + token sesión

Especificación Response

- 200 – OK

```
{  
    "message": "Línea de Presupuesto eliminada.",  
    "messageinfo": {  
        "user_id": "(string) id del usuario",  
        "budget_id": "(string) id del presupuesto",  
        "line_no": "(number) número de línea",  
        "service_id": "(string) id del servicio",  
        "service_name": "(string) nombre del servicio",  
        "measure_unit_id": "(string) id unidad de medida",  
        "measure_unit_name": "(string) nombre de la unidad de medida",  
        "quantity": "(decimal) cantidad",  
        "price": "(decimal) precio",  
        "amount": "(decimal) importe",  
        "_id": "(string) id de la linea",  
        "createdAt": "(datetime) fecha y hora de creación.",  
        "updatedAt": "(datetime) fecha y hora de modificación.",  
        "__v": 0  
    }  
}
```

- 401 – Usuario no autorizado

```
{  
    "message": "Construcción no encontrado.",  
    "messageinfo": "No se ha encontrado la construcción con el id proporcionado."  
}
```

UNIVERSIDAD NACIONAL DEL LITORAL - FICH
PROYECTO FINAL DE CARRERA – ANALISTA INFORMÁTICA APLICADA

- 404 – Línea de Presupuesto no encontrada

```
{  
    "message": "Línea de Presupuesto no encontrada.",  
    "messageinfo": "No existe una línea de presupuesto para la combinación  
de Línea de Presupuesto y Presupuesto."  
}
```

- 500 – Error en el servidor

```
{  
    "message": "Error en el servidor.",  
    "messageinfo": "Mensaje de error del servidor proporcionado por  
try/catch."  
}
```

UNIVERSIDAD NACIONAL DEL LITORAL - FICH
PROYECTO FINAL DE CARRERA – ANALISTA INFORMÁTICA APLICADA

Reporte (/reporter)

Obtener Presupuesto en PDF (*/{id presupuesto}*)

Este método retorna el reporte PDF de un presupuesto que se identifica por su id en el parámetro de la URL.

URL: <https://sappcon.site:3000/building/{id de Presupuesto}>

Especificación Request

- **Método HTTP:** GET
- **Headers:**
 - **Content-Type:** application/json
 - **Accept:** */*
 - **Authorization:** "Bearer " + token sesión

Especificación Response

- 200 – OK (ejemplo)

Nº Linea	Servicio	Unidad Medida	Cantidad	Precio Unitario	Total Linea
1	Calefon	Unidad	1	\$ 150000.00	\$ 150000.00
Total					\$ 150000.00

UNIVERSIDAD NACIONAL DEL LITORAL - FICH
PROYECTO FINAL DE CARRERA – ANALISTA INFORMÁTICA APLICADA

- 401 – Usuario no autorizado

```
{  
    "message": "Usuario no autorizado.",  
    "messageinfo": "No se ha proporcionado token válido para  
el usuario."  
}
```

- 500 – Error en el servidor

```
{  
    "message": "Error en el servidor.",  
    "messageinfo": "Mensaje de error del servidor  
proporcionado por try/catch."  
}
```

UNIVERSIDAD NACIONAL DEL LITORAL - FICH
PROYECTO FINAL DE CARRERA – ANALISTA INFORMÁTICA APLICADA

Estadísticas (/stats)

Estados por Presupuestos (/budgetStatus)

Este método retorna la estadística de cuántos presupuestos hay para cada estado de

URL: <https://sappcon.site:3000/stats/budgetStatus>

presupuesto, y el porcentaje que representa.

Especificación Request

- Método HTTP: GET
- Headers:
 - **Content-Type:** application/json
 - **Accept:** */*
 - **Authorization:** "Bearer " + token sesión

Especificación Response

- 200 – OK (ejemplo)

```
{
  "statusList": [
    {
      "status": "Pendiente",
      "count": 1,
      "percentage": 100
    },
    {
      "status": "Aprobado",
      "count": 0,
      "percentage": 0
    },
    {
      "status": "Rechazado",
      "count": 0,
      "percentage": 0
    },
    {
      "status": "En Construcción",
      "count": 0,
      "percentage": 0
    },
    {
      "status": "Finalizado",
      "count": 0,
      "percentage": 0
    }
  ]
}
```

UNIVERSIDAD NACIONAL DEL LITORAL - FICH
PROYECTO FINAL DE CARRERA – ANALISTA INFORMÁTICA APLICADA

Importes por Presupuestos (/budgetAmount)

Este método retorna medidas estadísticas para los presupuestos: importe mínimo, importe máximo, importe promedio, mediana de importe.

URL: <https://sappcon.site:3000/stats/budgetAmount>

Especificación Request

- **Método HTTP:** GET
- **Headers:**
 - **Content-Type:** application/json
 - **Accept:** */*
 - **Authorization:** "Bearer " + token sesión

Especificación Response

- 200 – OK (ejemplo)

```
{  
    "minAmount": 150000,  
    "maxAmount": 150000,  
    "avgAmount": 150000,  
    "medianAmount": 150000  
}
```

Servicios más populares (/topServices)

Este método retorna un arreglo de hasta 10 elementos, listando en orden, los 10 servicios más populares (es decir aquellos que tengan mayor cantidad de aparición en la totalidad de los presupuestos).

URL: <https://sappcon.site:3000/stats/topServices>

Especificación Request

- **Método HTTP:** GET
- **Headers:**
 - **Content-Type:** application/json
 - **Accept:** */*
 - **Authorization:** "Bearer " + token sesión

Especificación Response

- 200 – OK (ejemplo)

UNIVERSIDAD NACIONAL DEL LITORAL - FICH
PROYECTO FINAL DE CARRERA – ANALISTA INFORMÁTICA APLICADA

```
[  
  {  
    "rank": 1,  
    "serviceName": "Pared",  
    "quantity": 3,  
    "percentage": 75  
  }, {  
    "rank": 2,  
    "serviceName": "Techo",  
    "quantity": 1,  
    "percentage": 25  
  }]  
]
```

Servicios por Categoría (/servicesPerCategory)

Este método retorna cuántos Servicios hay registrados para cada categoría.

URL: <https://sappcon.site:3000/stats/servicesPerCategory>

Especificación Request

- Método HTTP: GET
- Headers:
 - **Content-Type:** application/json
 - **Accept:** */*
 - **Authorization:** “Bearer ” + token sesión

UNIVERSIDAD NACIONAL DEL LITORAL - FICH
PROYECTO FINAL DE CARRERA – ANALISTA INFORMÁTICA APLICADA

Especificación Response

- 200 – OK (ejemplo)

```
[  
  {  
    "categoryName": "Diseño mod",  
    "count": 1  
  },  
  {  
    "categoryName": "General",  
    "count": 2  
  },  
  {  
    "categoryName": "Acabados",  
    "count": 1  
  },  
  {  
    "categoryName": "Diseño",  
    "count": 2  
  },  
  {  
    "categoryName": "Tareas de preparación",  
    "count": 5  
  },  
  {  
    "categoryName": "Instalaciones eléctricas",  
    "count": 1  
  },  
  {  
    "categoryName": "Unknown",  
    "count": 1  
  }]
```

UNIVERSIDAD NACIONAL DEL LITORAL - FICH
PROYECTO FINAL DE CARRERA – ANALISTA INFORMÁTICA APLICADA

Imágenes (/image)

Subir Imagen (/upload)

Este método se utiliza para cargar una imagen para un Cliente o para el Usuario en la base de datos.

La imagen se envía y almacena en la base de datos encriptada en base64.

URL: <https://sappcon.site:3000/image/upload>

Especificación Request

- **Método HTTP:** POST
- **Headers:**
 - **Content-Type:** application/json
 - **Accept:** */*
 - **Authorization:** "Bearer " + token sesión
- **Body:**

```
{  
    "object_type": "(enum) User/Customer",  
    "object_id": "(string) id del Usuario o del Cliente",  
    "image_b64": "(string) imagen codificada en base64"  
}
```

Especificación Response

200 – OK

```
{  
    "message": "OK",  
    "messageinfo": "Imagen subida exitosamente."  
}
```

UNIVERSIDAD NACIONAL DEL LITORAL - FICH
PROYECTO FINAL DE CARRERA – ANALISTA INFORMÁTICA APLICADA

Descargar Imagen (/download)

Este método se utiliza para descargar una imagen para un Cliente o para el Usuario en la base de datos.

La imagen se recibe encriptada en base64.

URL: <https://sappcon.site:3000/image/download>

Especificación Request

- **Método HTTP: POST**
- **Headers:**
 - **Content-Type:** application/json
 - **Accept:** */*
 - **Authorization:** "Bearer " + token sesión
- **Body:**

```
{  
    "object_type": "(enum) User/Customer",  
    "object_id": "(string) id del Usuario o del Cliente"  
}
```

Especificación Response

200 – OK

```
{  
    "image_b64": "(string) imagen codificada en base64"  
}
```