**ESPECIFICACIONES DEL SERVICIO**

**Integrantes:**

**Rodney Zapata Palacio**

**Presentado al instructor:**

**Gustavo Adolfo Rodriguez**

**Servicio Nacional de aprendizaje SENA**

**Centro de Comercio y Servicios (Regional Cauca)**

**Cauca - Popayán**

**Tecnólogo en Análisis y Desarrollo de Software**

**Ficha: 2675810**

**07 de septiembre del 2024**

**Tabla Contenido**

Tabla de contenido

[1. DESCRIPCION GENERAL 3](#_Toc177112169)

[2. SERVICIOS API 3](#_Toc177112170)

[2.1 Consultar clientes por Id 3](#_Toc177112171)

[2.2 Creación de un cliente 4](#_Toc177112172)

[2.3 Eliminar un cliente 5](#_Toc177112173)

[2.4 Consultar técnicos por Id 5](#_Toc177112174)

[2.5 Consultar actividades por Id 6](#_Toc177112175)

[2.6 Eliminar actividades por Id 6](#_Toc177112176)

# DESCRIPCION GENERAL

Esta aplicación permite a los usuarios registrar, consultar, eliminar y actualizar los clientes a los que se les presta el servicio de mantenimiento, también podemos administrar los técnicos, las actividades a realizar, los refrigerantes que son utilizados en los aires acondicionado, así como las ordenes de servicio.

# SERVICIOS API

# Consultar clientes por Id

Método: GET

URL: http://localhost:8080/customer/Id

Descripción: Consulta un cliente por Id, se debe colocar en Id el numero del registro buscado por ejemplo el registro 1

Entrada:

<http://localhost:8080/customer/1>

Salida:

{

    "message": "Registro encontrado",

    "status": 200,

    "data": {

        "id": 1,

        "fullName": "Evelin Palacio ",

        "bornDate": "2023-12-28T00:00:00.000+00:00",

        "state": **null**,

        "email": "evelin2005@gmail.com",

        "phone": "3024694523",

        "address": "Calle 10 No. 15-24",

        "city": "Barranquilla"

    }

}

# Creación de un cliente

Método: POST

URL: <http://localhost:8080/customer/>

Descripción: Creamos un cliente, la API genera el Id automáticamente.

Entrada:

{

"fullName": "Gelkin Zapata ",

"bornDate": "1977-12-05",

"email": "gelkin2010@gmail.com",

"phone": "3024694523",

"address": "Via 40 No. 75-26",

"city": "Barranquilla"

}

Salida:

{

    "message": "Registro creado con exito",

    "status": 200,

    "data": {

        "id": 5,

        "fullName": "Gelkin Zapata ",

        "bornDate": "1977-12-05T00:00:00.000+00:00",

        "state": **null**,

        "email": "gelkin2010@gmail.com",

        "phone": "3024694523",

        "address": "Via 40 No. 75-26",

        "city": "Barranquilla"

    }

}

# Eliminar un cliente

Método: DELETE

URL: <http://localhost:8080/customer/>

Descripción: Eliminamos un cliente por Id.

Ejemplo: http://localhost:8080/customer/6

Entrada:

http://localhost:8080/customer/6

Salida:

{

    "message": "Registro eliminado",

    "status": 200,

    "data": **true**

}

# Consultar técnicos por Id

Método: GET

URL: http://localhost:8080/technician

Descripción: Consulta un técnico por Id, se debe colocar en Id el número del registro buscado por ejemplo el registro 1

Entrada:

<http://localhost:8080/technician/1>

Salida:

{

    "message": "Registro encontrado",

    "status": 200,

    "data": {

        "id": 1,

        "fullName": "tecnico1 Samuel ",

        "email": "samuel1@gmail.com",

        "phone": "3100000000",

        "hireDate": "1979-12-30T00:00:00.000+00:00"

    }

}

# Consultar actividades por Id

Método: GET

URL: http://localhost:8080/activity

Descripción: Consulta todas las actividades por Id

Entrada:

<http://localhost:8080/activity/1>

Salida:

{

    "message": "Registro encontrado",

    "status": 200,

    "data": {

        "id": 1,

        "activityName": "Registrar la presión de entrada y salida de agua de agua"

    }

}

# Eliminar actividades por Id

Método: DELETE

URL: http://localhost:8080/activity

Descripción: Eliminamos la actividad con Id=23

Entrada:

<http://localhost:8080/activity/23>

Salida:

{

    "message": "Registro eliminado",

    "status": 200,

    "data": **true**

}