

# GestionApp

API versión: 1.0

API Host: <http://localhost:5000/api/1.0>

**Descripción:** Nuestra API es de tipo REST y tiene como finalidad permitir interactuar con la Base de Datos gestionapp permitiendo, agregar, modificar, listar y borrar datos de teléfonos, sensores, apps, operarios e instalaciones realizadas a los teléfonos.

También permite realizar consultas específicas sobre:

- Teléfonos y sus sensores.
- Apps instaladas en un teléfono y que operario realizo la instalación.
- Listar las Apps instaladas discriminándolas por las que fueron agregadas exitosamente y las que no.
- Filtrar la lista de teléfonos en base al sensor del cual dispone o la App instalada.
- Mostrar una lista de operarios y cuantas Apps instalaron en una fecha determinada.

**Interactuar con la API:** se recomienda el uso de Postman para realizar peticiones de forma rápida.

**Formato de respuestas de la API:** siempre que se realice una petición correcta a la API se obtendrá un JSON con el atributo data y un código de respuesta HTTP 200.

Por el contrario, las peticiones incorrectas devolverán un código de error que puede ser un Bad Request o un Not Found dependiendo el caso

En esta documentación encontrara ejemplos para cada recurso de la API con las diferentes posibles respuestas HTTP al realizar una petición.

## End-Points

### • Teléfonos

#### • **GET** [localhost:5000/api/1.0/Telefonos](http://localhost:5000/api/1.0/Telefonos)

Devuelve una lista de los teléfonos que hay en la Base de Datos.

Este End-Point no requiere parámetros.

Ejemplo de la salida:

#### • **GET** [localhost:5000/api/1.0/Telefonos/{id}](http://localhost:5000/api/1.0/Telefonos/{id})

Devuelve la información de un teléfono específica.

Este End-Point toma como parámetro el id del teléfono a devolver.

Ejemplo: [localhost:5000/api/1.0/telefono/1](http://localhost:5000/api/1.0/telefono/1)

Ejemplo de la salida:

#### • **POST** [localhost:5000/api/1.0/telefonos](http://localhost:5000/api/1.0/telefonos)

Permite agregar telefonos a la base de datos. Pasándole los parámetros en el Body. El parámetro SensoresList es obligatorio y no puede ser NULL, sin embargo si puede estar vacío.

En SensoresList se especifica el Id de los sensores que tiene el teléfono.

Parámetro:

Marca: string

Modelo: string

Precio: float

SensoresList: List [int] (PARAMETRO OBLIGATORIO)

Ejemplo:

```
{
  "Marca": "Samsung",
  "Modelo": "Galaxy S9",
  "Precio": 25000.00,
  "SensoresList": [1]
}
```

Ejemplo de salida:

```
{
  "TelefonoId": 9,
  "Marca": "Samsung",
  "Modelo": "Galaxy S9",
  "Precio": 25000.00,
  "SensoresList": [1]
}
```

#### • **PUT** [localhost:5000/api/1.0/Telefonos/{id}](http://localhost:5000/api/1.0/Telefonos/{id})

Permite modificar los datos de un telefono. Pasándole como parámetros el id, marca, modelo, precio, SensoresList del teléfono en el Body. El parámetro SensoresList es obligatorio y no puede ser NULL, sin embargo si puede estar vacío.

El id pasado como parámetro debe ser igual al pasado en el Body.

Parámetros:

Marca: string

Modelo: string

Precio: float

SensoresList: List [int] (PARAMETRO OBLIGATORIO)

Ejemplo:  
**localhost:5000/api/1.0/telefonos/3**

```
{
  "TelefonoId": 3,
  "Marca": "MOTOROLA",
  "Modelo": "OneVision",
  "Precio": 50000.00,
  "SensoresList": [1, 4]
}
```

Ejemplo de salida:

```
{
  "TelefonoId": 3,
  "Marca": "MOTOROLA",
  "Modelo": "OneVision",
  "Precio": 50000.00,
  "SensoresList": [1, 4]
}
```

- **DEL localhost:5000/api/1.0/Telefonos/{id}**

Permite eliminar un teléfono. Pasándole como parámetros el id en la URL.

Parámetros: **Id: Int** (PARAMETRO OBLIGATORIO)

Ejemplo: **localhost:5000/api/1.0/telefonos/3**

Salida: en caso de pasar un Id inexistente devuelve NOT FOUND

- **GET localhost:5000/api/1.0/Telefonos/TelxId?Id={Id}**

Devuelve información de apps instaladas y qué operario lo hizo para un teléfono en particular,

Pasándole por parámetros el Id del teléfono en La URL.

Parámetros: **Id: Int** (PARAMETRO OBLIGATORIO)

Ejemplo: **localhost:5000/api/1.0/telefonos/TelxId?Id=3**

Salida:

```
{
  "marca": "Motorola",
  "modelo": "E7",
  "listaappsinstaladas": [
    {
      "aplicacion": "Instagram",
      "nombre_Operario": "Javier",
      "apellido_Operario": "Rodriguez"
    }
  ]
}
```

- **GET localhost:5000/api/1.0/Telefonos/Filtroappsen?Idsen=xx||Idapp=xx**

Devuelve una lista de telefonos filtrando por sensor o por aplicación, pasándole por parámetros en la URL el Id del sensor o de la app.

Parámetros: **Idsen: Int** (Opcional)

**Idapp: Int** (Opcional)

Ejemplo1: **localhost:5000/api/1.0/telefonos/filtroappsen?idapp=4**

Ejemplo2: **localhost:5000/api/1.0/telefonos/Filtroappsen?Idsen=4**

Salida Ejemplo1:

```
{
  "nombre_App": "TikTok",
  "listatelefonos": [
    {
      "Marca": "Motorola",
      "Modelo": "E7",
      "Precio": 15000.00
    }
  ]
}
```

Salida Ejemplo2:

```
{
  "nombre_Sensor": "Proximidad",
  "listatelefonos": [
    {
      "marca": "Motorola",
      "modelo": "OneVision",
      "precio": 50000.00
    }
  ]
}
```

- **Apps**

- **GET localhost:5000/api/1.0/apps**

Devuelve una lista de las Apps que hay en la Base de Datos.

Este End-Point no requiere parámetros.

- **GET localhost:5000/api/1.0/apps/{id}**

Devuelve la información de una App específica.

Este End-Point toma como parámetro el id de la App a devolver.

Parámetro: **Id: int**

Ejemplo: **localhost:5000/api/1.0/apps/1**

- **POST localhost:5000/api/1.0/apps**

Permite agregar Apps a la base de datos.

Pasándole como parámetro el nombre en el Body

Parámetro: **nombre: string**

El campo instalaciones es una colección que se utiliza más adelante para agregar las Instalaciones en las cuales se agregó la app.

Ejemplo: {  
 "nombre": "WhatsApp"  
 }  
 Ejemplo de salida: {  
 "appld": 1,  
 "nombre": "Whatsapp",  
 "instalaciones": null  
 }

- **PUT** `localhost:5000/api/1.0/Apps/{id}`  
 Permite modificar los datos de una app, pasándole como parámetros el id y el nombre de la app en el Body. El id pasado como parámetro debe ser igual al pasado en el Body.  
 Parámetro: **Appld: int** (campo obligatorio)  
**Nombre: string**  
 Instalaciones es una colección, se utiliza más adelante para agregar las Instalaciones en las cuales se agregó la app.

Ejemplo:  
`localhost:5000/api/1.0/Apps/1`  

```
{
  "appld": 1,
  "nombre": "Facebook",
  "instalaciones": null
}
```

- **DEL** `localhost:5000/api/1.0/Apps/{id}`  
 Permite eliminar una app. Pasándole como parámetros el id en la URL.  
 Parámetros: **Id: Int** (PARAMETRO OBLIGATORIO)  
 Ejemplo: `localhost:5000/api/1.0/apps/6`  
 Salida: en caso de pasar un Id inexistente devuelve NOT FOUND

## • Operarios

- **GET** `localhost:5000/api/1.0/operarios`  
 Devuelve una lista de los operarios que hay en la Base de Datos.  
 Este End-Point no requiere parámetros.

- **GET** `localhost:5000/api/1.0/operarios/{id}`  
 Devuelve la información de un operario específico.  
 Este End-Point toma como parámetro el id del operario.  
 Parámetro: **Id: int**  
 Ejemplo: `localhost:5000/api/1.0/operarios/1`

- **POST** `localhost:5000/api/1.0/operarios`  
 Permite agregar operarios a la base de datos.  
 Pasándole como parámetro el nombre y Apellido en el Body  
 Parámetros: **nombre: string**  
**Apellido: string**

Ejemplo: 

```
{
  "nombre": "Miguel",
  "apellido": "Pérez"
}
```

  
 Ejemplo de salida: 

```
{
  "operarioid": 9,
  "nombre": "Miguel",
  "apellido": "Pérez"
}
```

- **PUT** `localhost:5000/api/1.0/operarios/{id}`  
 Permite modificar los datos de un operario.  
 Pasándole como parámetros el id, el nombre y el apellido del operario en el Body.  
 El id pasado como parámetro debe ser igual al pasado en el Body.  
 Parámetros:  
**Operarioid: int** (campo obligatorio)  
**Nombre: string**  
**Apellido: string**

Ejemplo:  
`localhost:5000/api/1.0/operarios/9`  

```
{
  "operarioid": 9,
  "nombre": "Roque",
  "apellido": "Pérez"
}
```

- **DEL** `localhost:5000/api/1.0/operarios/{id}`  
 Permite eliminar un operario. Pasándole como parámetros el id en la URL.  
 Parámetros: **Id: Int** (PARAMETRO OBLIGATORIO)  
 Ejemplo: `localhost:5000/api/1.0/operarios/8`  
 Salida: en caso de pasar un Id inexistente devuelve NOT FOUND

## • Sensores

- **GET** `localhost:5000/api/1.0/sensores`  
 Devuelve una lista de los sensores que hay en la Base de Datos.  
 Este End-Point no requiere parámetros.

- **GET** `localhost:5000/api/1.0/sensores/{id}`  
 Devuelve la información de un sensor específico.  
 Este End-Point toma como parámetro el id del sensor.  
 Parámetro: **Id: int**  
 Ejemplo: `localhost:5000/api/1.0/sensores/1`

- **POST** `localhost:5000/api/1.0/sensores`  
 Permite agregar sensores a la base de datos.  
 Pasándole como parámetro el nombre en el Body  
 Parámetros: **nombre: string**

Ejemplo: 

```
{
  "nombre": "ProxSensor"
}
```

  
 Ejemplo de salida: 

```
{
  "Sensorid": 6,
  "nombre": "ProxSensor"
}
```

<ul style="list-style-type: none"> <li>• <b>PUT</b> <code>localhost:5000/api/1.0/sensores/{id}</code>          Permite modificar los datos de un sensor, pasándole como parámetros el id y el nombre en el Body. El Id pasado como parámetro debe ser igual al pasado en el Body.          Parámetros: <b>SensorId: int</b> (campo obligatorio)  <b>Nombre: string</b></li> </ul>	Ejemplo: <code>localhost:5000/api/1.0/operarios/6</code> <pre>{   "SensorId": 6,   "nombre": "Acelerometro" }</pre>
---	---

<ul style="list-style-type: none"> <li>• <b>DEL</b> <code>localhost:5000/api/1.0/sensores/{id}</code>          Permite eliminar un sensor. Pasándole como parámetros el id en la URL.          Parámetros: <b>Id: Int</b> (PARAMETRO OBLIGATORIO)          Ejemplo: <code>localhost:5000/api/1.0/sensores/5</code>          Salida: en caso de pasar un Id inexistente devuelve NOT FOUND</li> </ul>
--

## • Instalaciones

<ul style="list-style-type: none"> <li>• <b>GET</b> <code>localhost:5000/api/1.0/instalaciones</code>          Devuelve una lista de las instalaciones que hay en la Base de Datos.          Este End-Point no requiere parámetros.</li> </ul>
--

<ul style="list-style-type: none"> <li>• <b>GET</b> <code>localhost:5000/api/1.0/instalaciones/{id}</code>          Devuelve la información de una instalación específica.          Este End-Point toma como parámetro el id de la instalación.          Parámetro: <b>Id: int</b>          Ejemplo: <code>localhost:5000/api/1.0/instalaciones/1</code></li> </ul>
---

<ul style="list-style-type: none"> <li>• <b>GET</b> <code>localhost:5000/api/1.0/instalaciones/filtroinst?exitosa={boolean}</code>          Devuelve una lista de instalaciones filtrándolas por exitosas y no exitosas, toma por parámetro un Booleano en la URL.          Parámetros: <b>Exitosa: boolean</b> (parámetro obligatorio)          Ejemplo1: <b>GET</b> <code>localhost:5000/api/1.0/instalaciones/filtroinst?exitosa=true</code>          Ejemplo2: <b>GET</b> <code>localhost:5000/api/1.0/instalaciones/filtroinst?exitosa=False</code></li> </ul>
---

Ejemplo1:

```
{
  "exitosa": true,
  "date": "2021-11-24T00:00:00",
  "aplicacion": "dk",
  "operario": "Javier",
  "apellido": "Rodriguez"
}
```

Ejemplo2:

```
{
  "exitosa": false,
  "date": "2021-11-26T00:00:00",
  "aplicacion": "TikTok",
  "operario": "Gabriel",
  "apellido": "Bueno"
}
```

<ul style="list-style-type: none"> <li>• <b>POST</b> <code>localhost:5000/api/1.0/instalaciones</code>          Permite agregar instalaciones a la base de datos.          Pasándole los parámetros en el Body.          La fecha se carga con el formato YYYY-MM-DD.          Parámetro:  <b>Exitosa: Boolean</b>  <b>Fecha: Datetime</b>  <b>AppId: Int</b>  <b>TelefonoId: Int</b>  <b>OperarioId: Int</b></li> </ul>	Ejemplo: <code>localhost:5000/api/1.0/instalaciones</code> Salida: <pre>{   "Exitosa": 0,   "Fecha": "2021-11-24",   "AppId": 1,   "TelefonoId": 3,   "OperarioId": 1 }</pre>
--	---

<ul style="list-style-type: none"> <li>• <b>PUT</b> <code>localhost:5000/api/1.0/instalaciones/{id}</code>          Permite modificar los datos de una instalación.          Pasándole como parámetros el id, exitosa, fecha, appld, Telefonold y Operariold en el Body. El id pasado como parámetro debe ser igual al pasado en el Body.          Parámetro:  <b>InstalacionId: Int</b> (parámetro obligatorio)  <b>Exitosa: Boolean</b>  <b>Fecal: datetime</b>  <b>AppId: int</b>  <b>TelefonoId: int</b>  <b>OperarioId: int</b></li> </ul>	Ejemplo: <code>localhost:5000/api/1.0/instalaciones/5</code> <pre>{   "InstalacionId": 5,   "Exitosa": 1,   "Fecha": "2021-11-26",   "AppId": 1,   "TelefonoId": 3,   "OperarioId": 1 }</pre>
---	---

**DEL** `localhost:5000/api/1.0/instalaciones/{id}`

Permite eliminar una instalación. Pasándole como parámetros el id en la URL.

Parámetros: `Id`: `Int` (PARAMETRO OBLIGATORIO)

Ejemplo: `localhost:5000/api/1.0/instalaciones/5`

Salida: en caso de pasar un Id inexistente devuelve NOT FOUND

Integrantes:

Sergio Barboza

Daniel Barrera

Maximiliano Madrid