

# Interfaz hardware control de accesos

Esta aplicación levanta un servidor en Node.js con conexión cifrada que recibe peticiones para realizar las diferentes acciones que puede ejecutar el hardware de control de puertas, como por ejemplo abrir una puerta o grabar huellas en los sensores de las puertas.

## Huellas

### Huellas - Borrar huellas almacenadas en el sensor

PUT

/huellas/borrar

#### Parámetro

Campo	Tipo	Descripción
destino	Number	ID de la o las huellas que se desean borrar de o de los sensores
huellas	Number[]	ID o IDs de las huellas que se desean borrar

- Ejemplo [#parameter-examples-Huellas-PutHuellasBorrar-2\_0\_0-0]

```
{
  "destino": 0,
  "huellas": [
    1,
    21,
    45
  ]
}
```

#### Success 200

Campo	Tipo	Descripción
mensaje	String	Mensaje de eliminación correcta

- Success [#success-examples-Huellas-PutHuellasBorrar-2\_0\_0-0]

```
HTTP/1.1 200 OK
{
  "mensaje": "Eliminando huellas 1,2"
}
```

#### Error 4xx

Nombre	Tipo	Descripción
409	json	Servidor ocupado HTTP/1.1 409 Conflict { "error": "Servidor ocupado" }

- 409 Servidor ocupado [#error-examples-Huellas-PutHuellasBorrar-2\_0\_0-0]

```
HTTP/1.1 409 Conflict
{
  "error": "Servidor ocupado"
}
```

## Huellas - Grabar nuevas huellas

PUT

/huellas/enviar

### Parámetro

Campo	Tipo	Descripción
destino	Number	ID del sensor en el cual se grabará la o las nuevas huellas, "0" en caso de enviar a todos los sensores
huellas	Number[]	ID o IDs de las huellas que se desean enviar

- Ejemplo [#parameter-examples-Huellas-PutHuellasEnviar-2\_0\_0-0]

```
{
  "destino": 0,
  "huellas": [
    1,
    21,
    45
  ]
}
```

### Success 200

Campo	Tipo	Descripción
mensaje	String	Mensaje de envío correcto

- Success [#success-examples-Huellas-PutHuellasEnviar-2\_0\_0-0]

```
HTTP/1.1 200 OK
{
  "mensaje": "Enviando huellas 1,2"
}
```

### Error 4xx

Nombre	Tipo	Descripción
409	json	Servidor ocupado HTTP/1.1 409 Conflict { "error": "Servidor ocupado" }

- 409 Servidor ocupado [#error-examples-Huellas-PutHuellasEnviar-2\_0\_0-0]

```
HTTP/1.1 409 Conflict
{
  "error": "Servidor ocupado"
}
```

# Huellas - Verificar la disponibilidad del servidor

GET

/huellas

## Success 200

Campo	Tipo	Descripción
mensaje	String	Mensaje de disponibilidad

- Success [#success-examples-Huellas-GetHuellas-2\_0\_0-0]

```
HTTP/1.1 200 OK
{
  "mensaje": "Servicio disponible"
}
```

## Error 4xx

Nombre	Tipo	Descripción
409	json	Servidor ocupado HTTP/1.1 409 Conflict { "error": "Servidor ocupado" }

- 409 Servidor ocupado [#error-examples-Huellas-GetHuellas-2\_0\_0-0]

```
HTTP/1.1 409 Conflict
{
  "error": "Servidor ocupado"
}
```

# MonitorMicroservicio

## MonitorMicroservicio - Sincronizar manualmente ID's y usuarios desde el microservicio

GET

/monitorMicroservicio/sincronizar

## Success 200

Campo	Tipo	Descripción
mensaje	String	Mensaje de disponibilidad

- Success [#success-examples-MonitorMicroservicio-GetMonitormicroservicioSincronizar-2\_0\_0-0]

```
HTTP/1.1 200 OK
{
```

```
"mensaje": "Sincronizando"
}
```

Error 4xx

Nombre	Tipo	Descripción
409	json	Servidor ocupado HTTP/1.1 409 Conflict { "error": "Servidor ocupado" }

- 409 Servidor ocupado [#error-examples-MonitorMicroservicio-GetMonitormicroservicioSincronizar-2\_0\_0-0]

```
HTTP/1.1 409 Conflict
{
  "error": "Servidor ocupado"
}
```

# Programas

## Programas - Programa para grabar el arduino

GET

/programas/:arduinoId

Parámetro

Campo	Tipo	Descripción
arduinoId	Number	ID del arduino de control de puertas o interfaz sensorial

- Success [#success-examples-Programas-GetProgramasArduinoid-2\_0\_0-0]

```
HTTP/1.1 200 OK
#include <SPI.h>
#include <UIPEthernet.h>
...
void setup() {
...
void loop() {
...
...

```

Error 4xx

Nombre	Tipo	Descripción
403	json	Datos erróneos HTTP/1.1 409 Forbidden { "error": "Datos erróneos" }
500	json	Error interno HTTP/1.1 500 Internal Error { "error": "Error interno del servidor" }

- 409 Servidor ocupado [#error-examples-Programas-GetProgramasArduinoid-2\_0\_0-0]

```
HTTP/1.1 409 Conflict
{

```

```
{
  "error": "Datos erróneos"
}
```

# Puertas

## Puertas - Abrir puertas

POST

/puertas

### Parámetro

Campo	Tipo	Descripción
persona	Number	ID de la persona que desea acceder
puerta	Number	ID de la puerta que se desea abrir

- Ejemplo [#parameter-examples-Puertas-PostPuertas-2\_0\_0-0]

```
{
  "persona": 1,
  "puerta": 4
}
```

### Success 200

Campo	Tipo	Descripción
mensaje	String	Mensaje de apertura correcta

- Success [#success-examples-Puertas-PostPuertas-2\_0\_0-0]

```
HTTP/1.1 200 OK
{
  "mensaje": "Abriendo puerta 4"
}
```

### Error 4xx

Nombre	Tipo	Descripción
401	json	No autorizado HTTP/1.1 401 Unauthorized { "error": "Usuario 1 no autorizado" }

- 401 No autorizado [#error-examples-Puertas-PostPuertas-2\_0\_0-0]

```
HTTP/1.1 401 Unauthorized
{
  "error": "Usuario 1 no autorizado"
}
```

# Sensores

## Sensores - Verificar la conexión de los sensores

GET

/sensores/verificarConexion/:idArduino

### Success 200

Campo	Tipo	Descripción
mensaje	String	Mensaje del estado de la conexión

- Success [#success-examples-Sensores-GetSensoresVerificarconexionIdarduino-2\_0\_0-0]

```
HTTP/1.1 200 OK
{
  "mensaje": "Verificación en curso"
}
```

### Error 4xx

Nombre	Tipo	Descripción
409	json	Servidor ocupado HTTP/1.1 409 Conflict { "error": "Servidor ocupado" }

- 409 Servidor ocupado [#error-examples-Sensores-GetSensoresVerificarconexionIdarduino-2\_0\_0-0]

```
HTTP/1.1 409 Conflict
{
  "error": "Servidor ocupado"
}
```