

Smart Home

Overview

This is the documentation for smart home web rest api by [reem alghamdi](#)

Version information

Version : 1.0

Contact information

Contact Email : reem.brain@gmail.com

License information

License : MIT

License URL : <https://opensource.org/licenses/MIT>

Terms of service : null

URI scheme

Host : gp.reem-codes.com

BasePath : /api

Schemes : HTTPS, HTTP

Tags

- **Command** : What user wants raspberry to do. It is a mapping between the time, the hardware and the configuration
- **Configuration** : All the possible configuration and status for a given hardware. eg: on, off, red, extended ...
- **Hardware** : Everything about the sensors and actuators connected to raspberry pi
- **Response** : What the raspberry wants the user to know. It is a mapping between the time of execution, the command triggered and whether the action was done or not
- **Schedule** : scheduling information for command if any. Specifys the days and time of day the user would like a command to be triggered

Paths

Add a new command

POST /command

Parameters

Type	Name	Description	Schema
Body	command <i>optional</i>	The command to create.	command

command

Name	Schema
configurationId <i>optional</i>	integer
hardwareId <i>optional</i>	integer
scheduleId <i>optional</i>	integer

Responses

HTTP Code	Description	Schema
201	Created	Command
405	Invalid input	No Content

Consumes

- `application/json`

Produces

- `application/json`

Tags

- Command

get all commands

GET /command

Responses

HTTP Code	Description	Schema
200	return an array of command objects	< Command > array

Produces

- [application/json](#)

Tags

- Command

get a command by id

GET /command/{commandId}

Parameters

Type	Name	Schema
Path	commandId <i>required</i>	integer

Responses

HTTP Code	Description	Schema
200	get the command	Command
404	Not found	No Content

Produces

- [application/json](#)

Tags

- Command

edit an existing command given its id

```
PUT /command/{commandId}
```

Parameters

Type	Name	Description	Schema
Path	commandId <i>required</i>		integer
Body	command <i>optional</i>	The command to edit.	command

command

Name	Schema
configurationId <i>optional</i>	integer
hardwareId <i>optional</i>	integer
scheduleId <i>optional</i>	integer

Responses

HTTP Code	Description	Schema
200	Edited	Command
404	Not found	No Content
405	Invalid input	No Content

Consumes

- [application/json](#)

Produces

- `application/json`

Tags

- Command

delete a command by id

```
DELETE /command/{commandId}
```

Parameters

Type	Name	Schema
Path	commandId <i>required</i>	integer

Responses

HTTP Code	Description	Schema
204	deleted	No Content
404	Not found	No Content

Tags

- Command

Add a new configuration to a hardware

```
POST /configuration
```

Parameters

Type	Name	Description	Schema
Body	configuration <i>optional</i>	The configuration to create.	configuration

configuration

Name	Description	Schema
description <i>optional</i>		string
hardwareId <i>optional</i>		integer
name <i>optional</i>	the configuration name	string

Responses

HTTP Code	Description	Schema
201	Created	Configuration
405	Invalid input	No Content

Consumes

- `application/json`

Produces

- `application/json`

Tags

- Configuration

get a configuration by id

```
GET /configuration/{configurationId}
```

Parameters

Type	Name	Schema
Path	configurationId <i>required</i>	integer

Responses

HTTP Code	Description	Schema
200	get the configuration	Configuration
404	Not found	No Content

Produces

- `application/json`

Tags

- Configuration

edit an existing configuration given its id

```
PUT /configuration/{configurationId}
```

Parameters

Type	Name	Description	Schema
Path	configurationId <i>required</i>		integer
Body	configuration <i>optional</i>	The configuration to edit.	configuration

configuration

Name	Description	Schema
description <i>optional</i>		string
hardwareId <i>optional</i>		integer
name <i>optional</i>	the configuration name	string

Responses

HTTP Code	Description	Schema
200	Edited	Configuration
404	Not found	No Content
405	Invalid input	No Content

Consumes

- `application/json`

Produces

- `application/json`

Tags

- Configuration

delete a configuration by id

```
DELETE /configuration/{configurationId}
```

Parameters

Type	Name	Schema
Path	configurationId <i>required</i>	integer

Responses

HTTP Code	Description	Schema
204	deleted	No Content
404	Not found	No Content

Tags

- Configuration

Add a new Hardware to the system

POST /hardware

Parameters

Type	Name	Description	Schema
Body	hardware <i>optional</i>	The hardware to create.	hardware

hardware

Name	Description	Schema
description <i>optional</i>		string
gpio <i>optional</i>	the gpio pin the hardware is installed at	integer
icon <i>optional</i>	URL image to desired icon in client	string
name <i>optional</i>	the hardware name	string

Responses

HTTP Code	Description	Schema
201	Created	Hardware
405	Invalid input	No Content

Consumes

- `application/json`

Produces

- `application/json`

Tags

- Hardware

get all hardware connected to raspberry pi

GET /hardware

Responses

HTTP Code	Description	Schema
200	return an array of hardware objects	< Hardware > array

Produces

- `application/json`

Tags

- Hardware

get a hardware by id

GET /hardware/{hardwareId}

Parameters

Type	Name	Schema
Path	hardwareId <i>required</i>	integer

Responses

HTTP Code	Description	Schema
200	get the hardware	Hardware
404	Not found	No Content

Produces

- `application/json`

Tags

- Hardware

edit an existing hardware given its id

```
PUT /hardware/{hardwareId}
```

Parameters

Type	Name	Description	Schema
Path	hardwareId <i>required</i>		integer
Body	hardware <i>optional</i>	The hardware to edit.	hardware

hardware

Name	Description	Schema
description <i>optional</i>		string
gpio <i>optional</i>	the gpio pin the hardware is installed at	integer
icon <i>optional</i>	URL image to desired icon in client	string
name <i>optional</i>	the hardware name	string

Responses

HTTP Code	Description	Schema
200	Edited	Hardware

HTTP Code	Description	Schema
404	Not found	No Content
405	Invalid input	No Content

Consumes

- `application/json`

Produces

- `application/json`

Tags

- Hardware

delete a hardware by id

```
DELETE /hardware/{hardwareId}
```

Parameters

Type	Name	Schema
Path	hardwareId <i>required</i>	integer

Responses

HTTP Code	Description	Schema
204	deleted	No Content
404	Not found	No Content

Tags

- Hardware

get all configuration for a given hardware

GET /hardware/{hardwareId}/configuration

Parameters

Type	Name	Schema
Path	hardwareId <i>required</i>	integer

Responses

HTTP Code	Description	Schema
200	return an array of configuration objects	< Configuration > array

Produces

- [application/json](#)

Tags

- Configuration
- Hardware

Add a new response

POST /response

Parameters

Type	Name	Description	Schema
Body	response <i>optional</i>	The response to create.	response

response

Name	Schema
commandId <i>optional</i>	integer
executionTime <i>optional</i>	string (date-time)
isDone <i>optional</i>	boolean
message <i>optional</i>	string

Responses

HTTP Code	Description	Schema
201	Created	Response
405	Invalid input	No Content

Consumes

- `application/json`

Produces

- `application/json`

Tags

- Response

get all responses

GET /response

Responses

HTTP Code	Description	Schema
200	return an array of response objects	< Response > array

Produces

- `application/json`

Tags

- Response

get a response by id

```
GET /response/{responseId}
```

Parameters

Type	Name	Schema
Path	responseId <i>required</i>	integer

Responses

HTTP Code	Description	Schema
200	get the response	Response
404	Not found	No Content

Produces

- `application/json`

Tags

- Response

edit an existing response given its id

PUT /response/{responseId}

Parameters

Type	Name	Description	Schema
Path	responseId <i>required</i>		integer
Body	response <i>optional</i>	The response to edit.	response

response

Name	Schema
commandId <i>optional</i>	integer
executionTime <i>optional</i>	string (date-time)
isDone <i>optional</i>	boolean
message <i>optional</i>	string

Responses

HTTP Code	Description	Schema
200	Edited	Response
404	Not found	No Content
405	Invalid input	No Content

Consumes

- [application/json](#)

Produces

- `application/json`

Tags

- Response

delete a response by id

```
DELETE /response/{responseId}
```

Parameters

Type	Name	Schema
Path	responseId <i>required</i>	integer

Responses

HTTP Code	Description	Schema
204	deleted	No Content
404	Not found	No Content

Tags

- Response

Add a new schedule

```
POST /schedule
```

Parameters

Type	Name	Description	Schema
Body	schedule <i>optional</i>	The schedule to create.	schedule

schedule

Name	Description	Schema
commandId <i>optional</i>		integer
days <i>optional</i>	value between 0 and 127, representing 7 bits each bit correspond to a day in the order: sun mon tues web thurs fri sat Minimum value : 0 Maximum value : 127	integer (int64)
time <i>optional</i>		string (time)

Responses

HTTP Code	Description	Schema
201	Created	Schedule
405	Invalid input	No Content

Consumes

- `application/json`

Produces

- `application/json`

Tags

- Schedule

get all schedules

GET /schedule

Responses

HTTP Code	Description	Schema
200	return an array of schedule objects	< Schedule > array

Produces

- `application/json`

Tags

- Schedule

get a schedule by id

```
GET /schedule/{scheduleId}
```

Parameters

Type	Name	Schema
Path	scheduleId <i>required</i>	integer

Responses

HTTP Code	Description	Schema
200	get the schedule	Schedule
404	Not found	No Content

Produces

- `application/json`

Tags

- Schedule

edit an existing schedule given its id

```
PUT /schedule/{scheduleId}
```

Parameters

Type	Name	Description	Schema
Path	scheduleId <i>required</i>		integer
Body	schedule <i>optional</i>	The schedule to edit.	schedule

schedule

Name	Description	Schema
commandId <i>optional</i>		integer
days <i>optional</i>	value between 0 and 127, representing 7 bits each bit correspond to a day in the order: sun mon tues web thurs fri sat Minimum value : 0 Maximum value : 127	integer (int64)
time <i>optional</i>		string (time)

Responses

HTTP Code	Description	Schema
200	Edited	Schedule
404	Not found	No Content
405	Invalid input	No Content

Consumes

- [application/json](#)

Produces

- [application/json](#)

Tags

- Schedule

delete a schedule by id

```
DELETE /schedule/{scheduleId}
```

Parameters

Type	Name	Schema
Path	scheduleId <i>required</i>	integer

Responses

HTTP Code	Description	Schema
204	deleted	No Content
404	Not found	No Content

Tags

- Schedule

Definitions

Command

Name	Schema
configuration <i>optional</i>	Configuration
hardware <i>optional</i>	Hardware
id <i>optional</i>	integer (int64)
schedule <i>optional</i>	Schedule

Name	Schema
updatedAt <i>optional</i>	string (date-time)

Configuration

Name	Schema
description <i>optional</i>	string
hardware <i>optional</i>	Hardware
id <i>optional</i>	integer (int64)
name <i>optional</i>	string
updatedAt <i>optional</i>	string (date-time)

Hardware

Name	Schema
description <i>optional</i>	string
gpio <i>optional</i>	integer (int64)
icon <i>optional</i>	string
id <i>optional</i>	integer (int64)
name <i>optional</i>	string

Name	Schema
status <i>optional</i>	Configuration
updatedAt <i>optional</i>	string (date-time)

Response

Name	Schema
command <i>optional</i>	Command
executionTime <i>optional</i>	string (date-time)
id <i>optional</i>	integer (int64)
isDone <i>optional</i>	boolean
message <i>optional</i>	string
updatedAt <i>optional</i>	string (date-time)

Schedule

Name	Description	Schema
command <i>optional</i>		Command
days <i>optional</i>	value between 0 and 127, representing 7 bits each bit correspond to a day in the order: sun mon tues web thurs fri sat Minimum value : 0 Maximum value : 127	integer (int64)
id <i>optional</i>		integer (int64)

Name	Description	Schema
time <i>optional</i>		string (time)
updateAt <i>optional</i>		string (date-time)