Integration manual
Local API
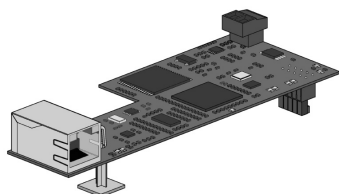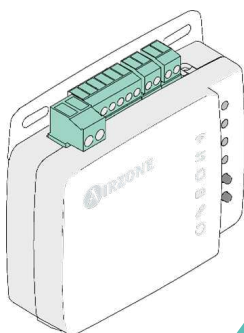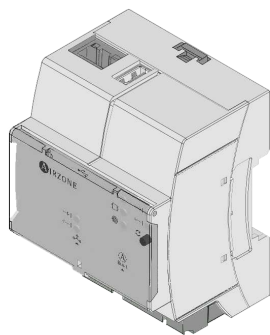
AIRZONE

CONTENTS

Integration with Airzone Systems

ℹ️ **Important:** *Webserver Airzone Cloud Ethernet (AZX6WEBSCLOUDC or AZX6WSCLOUDDINC) must have the 3.1.6 version or higher.*

SYSTEM IDENTIFICATION

To start the configuration process, it is necessary to identify the system, to do this, remove the Blueface thermostat from its base and check the code printed on the label located on the back of the thermostat.
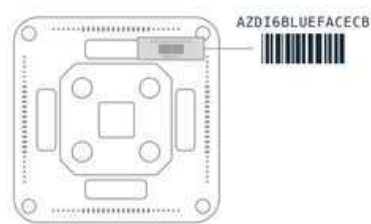

Depending on the code printed on the label the system will be configured in different ways, for further information, please refer to the documentation related to each system:

	Identification		Related documentation
AZCE6	Flexa 3.0 / Innobus Pro6 System	Quick Guide	Installation Manual
AZDI6	Acuazone / Innobus Pro32 System	Quick Guide	Installation Manual
AZRA6	RadianT365 System	Quick Guide	Installation Manual
AZVAF	VAF System	Quick Guide	Installation Manual
AZZBS	ZBS System	Quick Guide	Installation Manual

The Aidoo Pro system (AZAI6WSPxxx) does not have thermostats, but you can identify it by finding your Aidoo Pro device:
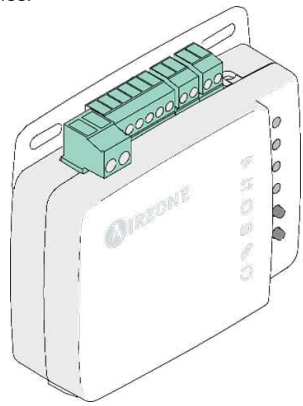
REQUESTS WORKFLOW

Requests are made pointed to an address, port and application.

> e.g http://XXX.XXX.XXX.XX:3000/api/v1/xxx

> Where XXX.XXX.XXX.XX is the IP address of the Airzone Webserver and the port is

3000

The Airzone system allows the control of the zones and parameters via PUT and POST requests.

> **POST** method: extract system data.

> **PUT** method: modify system data.

POST METHOD

The **POST** method is used to extract the data of a specified zone.

This method is used as below:

> **POST**   http://XXX.XXX.XXX.XX:3000/api/v1/hvac

> > Where XXX.XXX.XXX.XX is the IP address of the Airzone Webserver.

> > The port by default is 3000.

> > The application where is pointed is api/v1/hvac.

With the following body

> > {

> > "systemID": n (The system number could be between 1 and 32 depending on the installation),

> > "zoneID": m (The zone number could be between 0 and 32 depending on the installation. Zone 0 gives information on all zones in this system)

> > }

Method	Request URL
POST	http://192.168.0.35:3000/api/v1/hvac

Parameters ∧

Headers	Body	Variables

Body content type: application/json   Editor view: Raw input

FORMAT JSON   MINIFY JSON

```
{
"systemID": 1,
"zoneID": 1
}
```

200 OK 54.60 ms   DETAILS ∨

```
{
    "data": [{
        "systemID": 1,
        "zoneID":   1,
        "name": "ZONA 01",
        "on":    1,
        "maxTemp": 30,
        "minTemp": 18,
        "setpoint": 18.5,
        "roomTemp": 23.200000762939453,
        "modes":    [1, 4, 2, 3],
        "mode": 2,
        "coldStages":   1,
        "coldStage":    1,
        "heatStages":   1,
        "heatStage":    1,
        "humidity": 43,
        "units":    0,
        "errors":   [],
        "air_demand":   1,
        "floor_demand": 0,
        "aq_mode":  2,
        "aq_quality":   0,
        "aq_thrlow":    0,
        "aq_thrhigh":   0
    }]
}
```

POST request parameters

If the **POST** method is correctly requested the response is indicated with code **200** and will give back the following parameters:
A: Flexa 3.0/Innobus Pro6, RadianT365, Acuazone, Easyzone Systems.
B: VAF, ZBS Systems.
C: Aidoo Pro.

A	B	C	Parameters	Type of value	Description	Available value
*	*	*	systemID	Integer	ID of the System to consult	1-32->A,B 1->C
*	*	*	zoneID	Integer	ID of the Zone to consult	1-32->A,B 1->C 0 (All Zones)
*	*	*	name	String	Name of zone	Name of zone
*	*	*	on	Boolean	Zone status. On/Off	true
						false
*	*	*	setpoint	Integer	Setpoint temperature	15-30 ˚C 59-86 ˚F
*	*	*	roomtemp	Integer	Room temperature	0-99 °C 32-210 ˚F
*	*	*	maxtemp	Integer	Upper limit setpoint temperature	15-30 ˚C 59-86 ˚F
*	*	*	mintemp	Integer	Lower limit setpoint temperature	18-30 ˚C 64.4-86 ˚F
	*		coolsetpoint*	Integer	Setpoint temperature	18-30 ˚C 64.4-86 ˚F
	*		coolmaxtemp*	Integer	Upper limit cooling temperature	30 °C 86 ˚F
	*		coolmintemp*	Integer	Lower limit cooling temperature	18 °C 64.4 ˚F
	*		heatsetpoint*	Integer	Setpoint temperature	15-30 °C 59-86 ˚F
	*		heatmaxtemp*	Integer	Upper limit heating temperature	30 °C 86 ˚F
	*		heatmintemp*	Integer	Lower limit heating temperature	15 °C 59 ˚F
*			acs_temp	Integer	DHW tank temperature	20-99 °C 68-210,2 °F
*			acs_setpoint	Integer	DHW setpoint temperature	20-99 °C 68-210,2 °F

A	B	C	Parameters	Type of value	Description	Available value	
*			acs_power	Integer	DHW on/off status	1:On 0:Off	
*			acs_powerful	Integer	on/off status powerfull	1:On 0:Off	
*			power	Integer	Power consumed in the last hour if the installation includes consumption meter (kW/h)	0-10	
*	*	*	modes	Array	Operation modes available in the system	1	Stop
						2	Cooling
						3	Heating
						4	Fan
						5	Dry
						7	Auto*
*	*	*	mode	Integer	Operation mode selected for the system	1	Stop
						2	Cooling
						3	Heating
						4	Fan
						5	Dry
						7	Auto*
*	*	*	speeds	Integer	Fan speeds available in the system	0	Auto
						1	Low speed
						2	Medium speed
						3	High speed
						4...7	Only in individual unit zone modules
*	*	*	speed	Integer	Fan speed selected for the system	0	Auto
						1	Low speed
						2	Medium speed
						3	High speed
						4...7	Only in individual unit zone modules
*	*		coldstages	Integer	Cooling stages available in the system / zone	1	Air

A	B	C	Parameters	Type of value	Description	Available value	
						2	Radiant
						3	Combined
*	*		coldstage	Integer	Cooling stage running	1	Air
						2	Radiant
						3	Combined
*	*		heatstages	Integer	Heating stages available in the system / zone	1	Air
						2	Radiant
						3	Combined
*	*		heatstage	Integer	Heating stage running	1	Air
						2	Radiant
						3	Combined
*	*		humidity	Integer	Relative humidity of the zone	Number (%)	
*	*	*	units	Integer	Temperature measure-ment units	0	CELSIUS
						1	FAHRENHEIT
*	*		air_demand	Integer	System air demand	0	Deactivated
						1	Activated
*	*		floor_demand	Integer	System floor demand	0	Deactivated
						1	Activated
*	*	*	slats_vertical	Integer	Blade position Vertical/ horizontal	0	1 Position
			slats_horizontal			1	2 Position
						2	3 Position
						3	4 Position
*			aq_quality	Integer	Air quality	0	Off
						1	Good
						2	Medium
						3	Low
*			aq_mode	Integer	Ionisation mode	0	Off
						1	On
						2	Auto

The "errors" parameter indicates the error or warning and zone where it is happening.

Parameters	Value type	Description	Available values	
errors	Array	Zone errors	3	Motorized element not connected
errors	Array	Zone errors	4	Motorized element blocked
errors	Array	Zone errors	5	Temperature probe – Open circuit
errors	Array	Zone errors	6	Temperature probe – Short circuit
errors	Array	Zone errors	7	Incompatible element
errors	Array	Zone errors	8	Communication error
errors	Array	System errors	9	Gateway-System communication error
errors	Array	System errors	11	Gateway-AC Unit communication error
errors	Array	System errors	13	Main Board-Control Module of Radiant Elements communication error
errors	Array	System errors	14	Main Control Board-Expansion Module Communication error
errors	Array	System errors	15	Energy Meter communication error
errors	Array	System errors	16	Energy Meter measurement error
errors	Array	System errors	C02	Main Control Board – Production Control Board communication error
errors	Array	System errors	C09	Aerothermal Gateway -Production Control Board Communication Error
errors	Array	warning (zone warnings)	Occupancy	
errors	Array	warning (zone warnings)	Window	
errors	Array	warning (zone warnings)	Anti-freezing	
errors	Array	warning (zone warnings)	Active dew protection	
errors	Array	warning (zone warnings)	Zone without thermostat.	
errors	Array	warning (zone warnings)	Low battery.	
errors	Array	warning (zone warnings)	Active dew.	This warning indicates that there is a risk of water condensation in the floor and that the zone's radiant system has been turned off. If an air system is available, this will be activated to reduce the humidity in the zone. Available in systems with radiant system in cold mode.

If the **POST** request is wrong, the response is indicated with code **500** and will give back the following parameters:

Parameters	Value type	Description	Available values	
errors	Array	Error	request malformed	Wrong request format
errors	Array	Error	zoneid not provided	Zone not present in the request
errors	Array	Error	systemid not provided	System not present in the request
errors	Array	Error	zoneid out of range	Zone not valid (0 –)
errors	Array	Error	systemid out of range	System not valid (0 – 32)
errors	Array	Error	zoneid not available	Zone not available
errors	Array	Error	internal error	Internal error in the application
errors	Array	Error	driver not provided	The driver is not indicated in the request
errors	Array	Error	method not supported	Unsupported method

Exemple:
"errors":   [{
                        "Zone": "Error 1"
}, {
                        "Zone": "Presence alarm"
}, {
                        "system": "Error 9"
}, {
                        "system": "C02"
}]

PUT METHOD

The **PUT** method is used to modify the data of a specified zone.
This method is used as below:

    PUT   http://XXX.XXX.XXX.XX:3000/api/v1/hvac

              Where XXX.XXX.XXX.XX is the IP address of the Airzone Webserver.

              The port by default is 3000.

              The application where is pointed is api/v1/hvac.

    With the following body

```
{
    "systemID": n (system number),
    "zoneID": m (zone number)
     "parameter" (parameter to modify, e.g "setpoint"): f (value),
}
```

PUT request parameters

The PUT method allows to modify the following parameters:
A: Flexa 3.0/Innobus Pro6, RadianT365, Acuazone, Easyzone Systems.
B: VAF, ZBS Systems.
C: Aidoo Pro.

A	B	C	Parameters	Type of value	Description	Available value	
○	○		on	Integer	On/Off	1:On 0:Off	
○	○		name	String	Name of zone	Name of zone	
○	○	○	setpoint	Integer	Setpoint temperature	15-30 ˚C 59-86 ˚F	
	○		coolsetpoint*	Integer	Setpoint temperature for cooling mode	18-30 ˚C 64.4-86 ˚F	
	○		heatsetpoint*	Integer	Setpoint temperature for heating mode	15-30 ˚C 59-86 ˚F	
○	○	○	mode	Integer	Operation mode	1 2 3 4 5 7	Stop Cooling Heating Fan Dry Auto*
○	○	○	speed	Integer	Fan speed	0 1 2 3 4...7	Auto Low speed Medium speed High speed Only in individual unit zone modules
○	○		coldstage	Integer	Cooling stages	1 2 3	Air Radiant Combined
○	○		heatstage	Integer	Heating stages	1 2 3	Air Radiant Combined

If the **PUT** method is correctly requested the response is indicated with code **200** and will give back the system parameters.
If the **PUT** method is requested wrong the response is indicated with code **500** and will give back the system parameters.

Parameters	Value type	Description	Available values	
errors	Array	Error	request malformed	Wrong request format
errors	Array	Error	zoneid not provided	Zone not present in the request
errors	Array	Error	systemid not provided	System not present in the request
errors	Array	Error	zoneid out of range	Zone not valid (0 – 32)
errors	Array	Error	systemid out of range	System not valid (0 – 32)
errors	Array	Error	zoneid not available	Zone not available
errors	Array	Error	internal error	Internal error in the application
errors	Array	Error	driver not provided	The driver is not indicated in the request
errors	Array	Error	method not supported	Unsupported method

INTEGRATION

Check integration

To check the installed driver version.
POST   http://XXX.XXX.XXX.XX:3000/api/v1/integration

               Where XXX.XXX.XXX.XX is the IP address of the Airzone Webserver.

               The port by default is 3000.

               The application where is pointed is api/v1/integration.

If the **POST** method is correctly requested the response is indicated with code **200** and will give back the following parameters:

```
{
        "driver": "integrator"
}
```

Where "integrator" stands for the system to control with Airzone.

Modify integration

To set the integration value.
PUT   http://XXX.XXX.XXX.XX:3000/api/v1/integration

               Where XXX.XXX.XXX.XX is the IP address of the Airzone Webserver.

               The port by default is 3000.

               The application where is pointed is api/v1/integration.

With the following body

```
{
        "driver": "integrator"
}
```

Where "integrator" stands for the system to control with Airzone.

If the **PUT** method is correctly requested the response is indicated with code **200** and will give back the following parameters:

```
{
        "driver": "integrator"
}
```

Where "integrator" stands for the system to control with Airzone.

If the **PUT** method is requested wrong the response is indicated with code **500** and will give back the system parameters.

```
{
        "errors":  [{
         "error": "integration not provided"
        }]
}
```