

Index

1	UPnP discovery	2
2	Avahi/Bonjour	2
3	Requests without Authentication key	2
3.1	GET requests	2
3.2	POST requests	2
4	Requests with Authentication key	2
5	Device (Sensors und Valves) Requests:	2
5.1	GET Requests für Device:	2
5.2	POST Requests für Device:	7
6	Circuit Requests	8
6.1	GET Requests für Circuit:	8
6.2	POST Requests für Circuits	10
7	Extern Requests	12
7.1	GET Requests:	12
7.2	POST Requests:	12
8	System Requests	13
8.1	GET Requests:	13
8.2	POST Requests:	14
9	Logging Requests	15
9.1	POST Requests:	15
10	Objects	17
10.1	Circuit	17
10.2	Devices	17

MIYO Cube HTTP API

1 UPnP discovery

Relevant response parts:

SERVER: miyocube/*VERSION_NUMBER*

USN: uuid:*CUBE_ID*::upnp:rootdevice

2 Avahi/Bonjour

miyocube.local

3 Requests without Authentication key

3.1 GET requests

Endpoint URL `http://*IP*/api/link` - Provides a new apiKey, if a new access is allowed. A button push on the cube or „allowlink“ allow one time request for 15 minutes.

3.2 POST requests

Endpoint URL `http://*IP*/api/system/update` - Schedules the next update. Won't start immediately.

4 Requests with Authentication key

Authentication parameter

apiKey	required	Authentication key	String
--------	----------	--------------------	--------

5 Device (Sensors und Valves) Requests:

5.1 GET Requests für Device:

Endpoint URL `http://*IP*/api/device/all` - Provides information of all devices.

Endpoint URL `http://*IP*/api/device/free` - Provides only unassigned devices (all which are not assigned to an irrigation area).

Example:

```
{
  "id": 0,
  "params": {
    "devices": {
      "{16762df9-c3b7-4f68-b8af-ccd47e7eddb5};1": {
        "channel": 1,
        "deviceTypeId": "valve",
        "firmware": "1.10.0",
        "id": "{16762df9-c3b7-4f68-b8af-ccd47e7eddb5}",
        "ipv6": "fe80::211:7d00:30:edbc%zmd0",
        "lastUpdate": 1527235914,
        "name": "viRaValve",
        "stateTypes": {
          "0": {
            "type": "valveInitialClose",
            "value": false
          },
          "1": {
            "type": "valveStatus",
            "value": false
          },
          "10": {
            "type": "lastChargingTime",
            "value": 1527235914
          },
          "11": {
            "type": "lowPower",
            "value": false
          },
          "12": {
            "type": "otauPossible",
            "value": false
          },
          "13": {
            "type": "otauProgress",
            "value": 0
          },
          "14": {
            "type": "otauStatus",
            "value": ""
          },
          "15": {
            "type": "winterMode",
            "value": false
          },
          "16": {
            "type": "chargingDurationWeekly",
            "value": 81
          },
          "17": {
            "type": "chargingDurationDay",
            "value": 11.571428571429
          }
        }
      }
    }
  }
}
```

```
    },
    "18": {
      "type": "charging",
      "value": true
    },
    "19": {
      "type": "chargingLess",
      "value": false
    },
    "2": {
      "type": "openValve",
      "value": false
    },
    "20": {
      "type": "clientChargingTime",
      "value": 0
    },
    "3": {
      "type": "lastIrrigationStart",
      "value": 0
    },
    "4": {
      "type": "lastIrrigationEnd",
      "value": 1527033928
    },
    "5": {
      "type": "lastIrrigationDuration",
      "value": 1527033928
    },
    "6": {
      "type": "rssi",
      "value": -78
    },
    "7": {
      "type": "reachable",
      "value": true
    },
    "8": {
      "type": "solarVoltage",
      "value": "4.25"
    },
    "9": {
      "type": "sunWithinWeek",
      "value": true
    }
  }
},
"{db31afa8-f851-4f26-81a2-c50f34c9068c};1": {
  "channel": 1,
  "deviceTypeId": "moistureOutdoor",
  "firmware": "",
  "id": "{db31afa8-f851-4f26-81a2-c50f34c9068c}",
  "ipv6": "fe80::211:7d00:30:eba3%zmd0",
```

```
"lastUpdate": 1527033850,
"name": "viRaMoistureSensor",
"stateTypes": {
  "0": {
    "type": "moisture",
    "value": 0
  },
  "1": {
    "type": "brightness",
    "value": 3238
  },
  "10": {
    "type": "sunWithinWeek",
    "value": false
  },
  "11": {
    "type": "lastChargingTime",
    "value": 1526036974
  },
  "12": {
    "type": "lowPower",
    "value": false
  },
  "13": {
    "type": "otauPossible",
    "value": false
  },
  "14": {
    "type": "otauProgress",
    "value": 0
  },
  "15": {
    "type": "otauStatus",
    "value": ""
  },
  "16": {
    "type": "winterMode",
    "value": false
  },
  "17": {
    "type": "chargingDurationWeekly",
    "value": 0
  },
  "18": {
    "type": "chargingDurationDay",
    "value": 0
  },
  "19": {
    "type": "charging",
    "value": false
  },
  "2": {
    "type": "temperature",
```

MIYO Cube HTTP API



```
"value": 28
},
"20": {
  "type": "chargingLess",
  "value": true
},
"21": {
  "type": "clientChargingTime",
  "value": 0
},
"3": {
  "type": "frequency",
  "value": 0
},
"4": {
  "type": "irrigationNecessary",
  "value": false
},
"5": {
  "type": "irrigationPossible",
  "value": false
},
"6": {
  "type": "temperatureOffset",
  "value": 0
},
"7": {
  "type": "rssi",
  "value": -100
},
"8": {
  "type": "reachable",
  "value": false
},
"9": {
  "type": "solarVoltage",
  "value": 0
}
}
}
}
},
"status": "success"
}
```

Array

id		String
status		String
params		Object
devices	List of all devices	Object

5.2 POST Requests für Device:

Endpoint URL `http://*IP*/api/device/remove` - Deletes a device.

deviceId	required	ID of the device	String
----------	----------	------------------	--------

Endpoint URL `http://*IP*/api/device/teachin` - Teaches a device, which is activated with a magnet.

Endpoint URL `http://*IP*/api/device/highlight` - Activates LED on the chosen device.

deviceId	required	ID of the device	String
----------	----------	------------------	--------

Example:

```
{
  "id":0,
  "status":"success"
}
```

or

```
{
  "error": "Device not found, used by circuit or is currently updating",
  "errorLoc": "NOTIFY_ERROR_DEVICENOTFOUND",
  "id": 0,
  "status": "error"
}
```

Array

id		Integer
status		String
error		String
errorLoc		String

6 Circuit Requests

6.1 GET Requests für Circuit:

Endpoint URL http://*IP*/api/circuit/all - Provides information of all irrigation areas.

Endpoint URL http://*IP*/api/circuit/status - Provides information of a irrigation area, analog to „all“ refers to just one area.

circuitId	required	ID of the area	String
-----------	----------	----------------	--------

Endpoint URL http://*IP*/api/circuit/types - Provides all possible types for soil composition, irrigation tools, plants and location.

Endpoint URL http://*IP*/api/circuit/log - Provides a statistic.

circuitId	required	ID of the area	String
-----------	----------	----------------	--------

Example:

```
{
  "id": 0,
  "params": {
    "circuits": {
      "{abc-1234-defg}": {
        "id": "{abc-1234-defg}",
        "name": "Test",
        "params": {
          "automaticMode": false,
          "borderBottom": "60",
          "borderTop": "80",
          "considerMower": false,
          "day0": "",
          "day1": "",
          "day2": ""
        }
      }
    }
  }
}
```



```
"day3": "",
"day4": "",
"day5": "",
"day6": "",
"irrigationType": 0,
"locationType": 0,
"plantType": 1,
"soilType": 0,
"valveStaggering": true
},
"sensor": "0",
"sensorValve": {
  "channel": 1,
  "valve": "{abc-1234-defg}"
},
"stateTypes": {
  "0": {
    "type": "irrigation",
    "value": false
  },
  "1": {
    "type": "automaticMode",
    "value": false
  },
  "2": {
    "type": "winterMode",
    "value": false
  },
  "3": {
    "type": "irrigationNextStart",
    "value": 0
  },
  "4": {
    "type": "irrigationNextEnd",
    "value": 0
  },
  "5": {
    "type": "valveStaggeringIndex",
    "value": 0
  }
},
"valves": {
  "0": {
    "channel": 1,
```

```

    "valve": "{abc-1234-defg}"
  }
}
},
"status": "success"
}

```

Array

params		Object
circuits	List of all circuits	Object
id	ID of the circuit	String
name	Name of the circuit	String

6.2 POST Requests für Circuits

Endpoint URL http://*IP*/api/circuit/add - Adds a new irrigation area.

circuitId	required	ID of the area	String
sensorId	required	ID of the the sensor in the circuit	String
name	required	Name of the area	String

Endpoint URL http://*IP*/api/circuit/edit - Edits an irrigation area.

circuitId	required	ID of the device	String
borderTop	not required	Humidity border	String
borderBottom	not required	Humidity border	String
considerMower	Not required	Consider extern mower	boolean
day0 (day1, day2, day3, day4, day5, day6)	not required	Allowed irrigation time format: 6:00-13:00;14:00-15:21	String

Endpoint URL http://*IP*/api/circuit/irrigation - Starts or stops the irrigation.

MIYO Cube HTTP API

circuitId	required	ID of the area	String
mode	required	start or stop	String
duration	not required	Duration of the irrigation	String

Endpoint URL http://*IP*/api/circuit/remove - Deletes an irrigation area.

circuitId	required	ID of the area	String
-----------	----------	----------------	--------

Endpoint URL http://*IP*/api/circuit/winter - Activates or deactivates the wintermode.

circuitId	required	ID of the device	String
winter	required	Turning off or on the winter mode	Boolean

Endpoint URL http://*IP*/api/circuit/log - Provides a statistic.

circuitId	required	ID of the device	String
timeFilters			Array of Objects
startDate			String
endDate			String

Example:

```
{
  "id": 0,
  "status": "success"
}
```

Array

id		Integer
status		String

7 Extern Requests

7.1 GET Requests:

Endpoint URL `http://*IP*/api/extern/status` - Provides current data of extern devices „rain“, „temperature“, „wind“ and „mower“.

Example:

```
{
  "id": 0,
  "params": {
    "mower": true,
    "rain": false,
    "temperature": false,
    "wind": false
  },
  "status": "success"
}
```

7.2 POST Requests:

Endpoint URL `http://*IP*/api/extern/rain` - Informs MIYO about rain.

rain	required		boolean
------	----------	--	---------

Endpoint URL `http://*IP*/api/extern/wind` - Defines if wind force is too strong.

wind	required		boolean
------	----------	--	---------

Endpoint URL `http://*IP*/api/extern/block` - Permits or prohibits the irrigation, for example during a party.

circuitId	required	ID of the device	String
block	required		boolean

Endpoint URL `http://*IP*/api/extern/mower` - Defines if a mower is currently active or not. The value `considerMower` in `Circuit/edit` must be set to "true".

circuitId	required	ID of the device	String
running	required		boolean

Endpoint URL `http://*IP*/api/extern/temperature` - Provides the temperature of extern devices.

MIYO Cube HTTP API

8 System Requests

8.1 GET Requests:

Endpoint URL `http://*IP*/api/system/status` - Provides information about the cube.

Endpoint URL `http://*IP*/api/system/time` - Provides current local time and timezone.

Endpoint URL `http://*IP*/api/system/timezone` - Provides all timezones.

Endpoint URL `http://*IP*/api/system/weather` - Provides the weather.

Example:

```
{
  "id": 0,
  "params": {
    "cloudAllowed": true,
    "lat": "",
    "linkMode": false,
    "lon": "",
    "updateStatus": 0,
    "uuid": "{abc-1234-defg}",
    "version": "0.90.1"
  },
  "status": "success"
}
```

Array

cloudAllowed	If cloud connection is allowed	boolean
lat	Latitude	
linkMode	Allows access for 15 minutes or one time request.	boolean
lon	Longitude	
updateStatus	"0" - not scheduled "1" - scheduled "2" - checking device "3" - installing "4" - error at last update	String
uuid	UUID	String
version	Currently installed firmware version	String

clock		Object
timestamp	UNIX time	Unsigned Integer
timezone	timezone	String

8.2 POST Requests:

Endpoint URL http://*IP*/api/system/reboot - Restarts the cube.

Endpoint URL http://*IP*/api/system/reset - Reset to factory settings.

Endpoint URL http://*IP*/api/system/reconnect - Reconnects the cube.

Endpoint URL http://*IP*/api/system/position - Sets the GPS Position of the cube (necessary for weather forecast).

lon	required	Longitude	String
lat	required	Latitude	String

Endpoint URL http://*IP*/api/system/allowlink - Allows a new user to add the cube without pressing the button.

Endpoint URL http://*IP*/api/system/cloud - Activates or deactivates the cloud connection.

allowed	required	Activates or deactivates the cloud connection	boolean
---------	----------	---	---------

Endpoint URL http://*IP*/api/system/timezone - Sets timezone of the cube.

timezone	required	startDate and endDate	String
----------	----------	-----------------------	--------

9 Logging Requests

9.1 POST Requests:

Endpoint URL `http://*IP*/api/Logging/entries`

Example:

```
{
  "id": 0,
  "params": {
    "history": [
      {
        "circuitId": "{abc-1234-defg}",
        "deviceId": "{abc-1234-defg}",
        "source": "LoggingSourceState",
        "stateTypeId": "irrigation",
        "timestamp": 1526020572000,
        "value": "false"
      },
      {
        "circuitId": "{abc-1234-defg}",
        "deviceId": "{abc-1234-defg}",
        "source": "LoggingSourceState",
        "stateTypeId": "irrigation",
        "timestamp": 1526025137000,
        "value": "false"
      },
      {
        "circuitId": "{abc-1234-defg}",
        "deviceId": "{abc-1234-defg}",
        "source": "LoggingSourceState",
        "stateTypeId": "irrigation",
        "timestamp": 1526032918000,
        "value": "false"
      },
      {
        "circuitId": "{abc-1234-defg}",
        "deviceId": "{abc-1234-defg}",
        "source": "LoggingSourceState",
        "stateTypeId": "solarVoltage",
        "timestamp": 1526037953000,
        "value": "4.01"
      },
      {
        "circuitId": "{abc-1234-defg}",
        "deviceId": "{abc-1234-defg}",
```

```
"source": "LoggingSourceState",  
"stateTypeId": "lastChargingTime",  
"timestamp": 1526037953000,  
"value": "1526037953"  
}  
],  
"status": "success"  
}
```

Array

circuitId	ID of the area	String
deviceId	ID of the device	String
stateTypeId	ID of the stateType	String
timestamp	UNIX time	Unsigned Integer
value	Value	String

10 Objects

10.1 Circuit

id		
name		String
automaticMode		Boolean
borderBottom	Humidity border	String
borderTop	Humidity border	String
day0		
irrigationType		Integer
locationType		Integer
plantType		Integer
soilType		Integer
valveStaggering		boolean
sensor		
stateTypes		
valves		Object

10.2 Devices

10.2.1 Valve

channel		Unsigned integer
deviceTypeId	Type of the device	String
firmware	Currently installed firmware version	String
id	ID of the device	String
ipv6	IPv6	String
lastUpdate	The UNIX time when the device was last updated	Unsigned integer
name	Name of the device	String
stateTypes	List of all stateTypes	Object

type	Name of the stateType	String
value	Value of the stateType	

10.2.2 Sensor

channel		Unsigned integer
deviceTypeId	Type of the device	String
firmware	Currently installed firmware version	String
id	ID of the device	String
ipv6	IPv6	String
lastUpdate	The UNIX time when the device was last updated	Unsigned integer
name	Name of the device	String
stateTypes	List of all stateTypes	Object
type	Name of the stateType	String
value	Value of the stateType	

Vienna, 19th March 2019