

POCloud Data Logger

Written by Patrick McDonagh, Henry Pump

Prerequisites

- NodeJS
- Python 2.7
- npm
- MySQL
- Bower

Installation

1. Clone this repository using either HTTPS:

```
git clone http://<USERNAME>@bitbucket.poconsole.net/scm/pocloud/tag-server.git
```

2. Change directory to the "www" folder

```
cd tag-server/www
```

3. Install necessary npm packages

```
npm install
```

4. Install necessary bower packages

```
bower install
```

5. Create init.d scripts

```
INFO HERE COMING SOON
```

6. Test Web Server functionality

```
sails lift
```

Proceed to <http://:3000> to test the web server and set up tags.

7. Enter tag information in web server.

8. Test the Python datalogger functionality.

```
python python/tagserver.py
```

The program should return messages saying it has stored values. If errors occur, check the PLC ip address. If errors persist, contact Henry Pump.

9. Start the init.d scripts so the programs run in the background.

```
/etc/init.d/tagserver start
```

and

```
/etc/init.d/website start  
OR  
forever www/app.js
```

Database Documentation

MySQL Tables

```

+-----+
| Tables_in_poconsole |
+-----+
| config                |
| data_types            |
| device_types          |
| devices               |
| tag_classes           |
| tag_vals              |
| tags                  |
+-----+

```

config Table

Holds parameter/val pairs for configuration parameters

```

+-----+-----+-----+-----+-----+-----+
| Field  | Type           | Null | Key | Default | Extra           |
+-----+-----+-----+-----+-----+-----+
| id      | int(10) unsigned | NO   | PRI | NULL    | auto_increment |
| parameter | varchar(255)    | YES  | UNI | NULL    |                 |
| val     | varchar(255)    | YES  |     | NULL    |                 |
| createdAt | datetime        | YES  |     | NULL    |                 |
| updatedAt | datetime        | YES  |     | NULL    |                 |
+-----+-----+-----+-----+-----+-----+

```

data_types Table

Holds all possible data types

```

+-----+-----+-----+-----+-----+-----+
| Field  | Type           | Null | Key | Default | Extra           |
+-----+-----+-----+-----+-----+-----+
| id      | int(10) unsigned | NO   | PRI | NULL    | auto_increment |
| data_type | varchar(255)    | YES  | UNI | NULL    |                 |
| createdAt | datetime        | YES  |     | NULL    |                 |
| updatedAt | datetime        | YES  |     | NULL    |                 |
+-----+-----+-----+-----+-----+-----+

```

device_types Table

Holds all possible device types

Field	Type	Null	Key	Default	Extra
id	int(10) unsigned	NO	PRI	NULL	auto_increment
dType	varchar(255)	YES	UNI	NULL	
createdAt	datetime	YES		NULL	
updatedAt	datetime	YES		NULL	

devices Table

Holds device configuration.

devices.device_type = device_types.dType

Field	Type	Null	Key	Default	Extra
id	int(10) unsigned	NO	PRI	NULL	auto_increment
device_type	int(11)	YES		NULL	
address	varchar(255)	YES		NULL	
createdAt	datetime	YES		NULL	
updatedAt	datetime	YES		NULL	

tag_classes Table

Holds all possible tag classes. All Data Logger tags are "Custom" (id=5)

Field	Type	Null	Key	Default	Extra
id	int(10) unsigned	NO	PRI	NULL	auto_increment
class_type	varchar(255)	YES	UNI	NULL	
description	varchar(255)	YES		NULL	
createdAt	datetime	YES		NULL	
updatedAt	datetime	YES		NULL	

tag_vals Table

Holds stored measurements of values of tags.

tag_vals.tagID = tags.id

Field	Type	Null	Key	Default	Extra
id	int(10) unsigned	NO	PRI	NULL	auto_increment
tagID	int(11)	YES		NULL	
val	float	YES		NULL	
createdAt	datetime	YES		NULL	
updatedAt	datetime	YES		NULL	

tags Table

Holds the scan-list of tags to be evaluated.

tag.tag_class = tag_classes.id

tag.deviceID = devices.id

tag.data_type = data_types.id

Field	Type	Null	Key	Default	Extra
id	int(10) unsigned	NO	PRI	NULL	auto_increment
name	varchar(255)	YES		NULL	
tag_class	int(11)	YES		NULL	
tag	varchar(255)	YES		NULL	
deviceID	int(11)	YES		NULL	
description	varchar(255)	YES		NULL	
data_type	int(11)	YES		NULL	
change_threshold	float	YES		NULL	
guarantee_sec	int(11)	YES		NULL	
map_function	varchar(255)	YES		NULL	
units	varchar(255)	YES		NULL	
minExpected	float	YES		NULL	
maxExpected	float	YES		NULL	
createdAt	datetime	YES		NULL	
updatedAt	datetime	YES		NULL	

Web API

The web API provides a REST-ful API to access the tags and values stored in the database.

Devices

GET /device

Returns all the devices stored in the database

Returns:

```
[
  {
    "device_type": {
      "id": 1,
      "dType": "CLX",
      "createdAt": "2016-04-27T14:10:35.000Z",
      "updatedAt": "2016-04-29T17:02:15.000Z"
    },
    "id": 1,
    "address": "10.10.10.3",
    "createdAt": "2016-04-27T14:19:20.000Z",
    "updatedAt": "2016-04-29T17:02:15.000Z"
  },
  {
    "device_type": {
      "id": 2,
      "dType": "Micro800",
      "createdAt": "2016-04-27T14:10:43.000Z",
      "updatedAt": "2016-04-27T14:10:43.000Z"
    },
    "id": 11,
    "address": "192.168.1.20",
    "createdAt": "2016-04-29T21:36:28.000Z",
    "updatedAt": "2016-04-29T21:36:28.000Z"
  }
]
```

GET /device/

Returns only the information about the specific device

Returns:

```
[
  {
    "device_type": {
      "id": 1,
      "dType": "CLX",
      "createdAt": "2016-04-27T14:10:35.000Z",
      "updatedAt": "2016-04-29T17:02:15.000Z"
    },
    "id": 1,
    "address": "10.10.10.3",
    "createdAt": "2016-04-27T14:19:20.000Z",
    "updatedAt": "2016-04-29T17:02:15.000Z"
  }
]
```

POST /device

Adds a new device to the database

Parameters:

```
{
  address: < URL of the device >,
  device_type: < ID of device type>
}
```

UPDATE /device/

Updates a device in the database

Parameters:

```
{
  address: < URL of the device >,
  device_type: < ID of device type>
}
```

DELETE /device/

Deletes a device from the database

Tags

GET /tag

Returns all the tags stored in the database

Returns:

```
[
  {
    "tag_class": {
      "id": 5,
      "class_type": "custom",
      "description": "User-Selected Tags",
      "createdAt": "2016-04-27T14:17:54.000Z",
      "updatedAt": "2016-04-27T14:17:54.000Z"
    },
    "deviceID": {
      "id": 1,
      "address": "10.10.10.3",
      "createdAt": "2016-04-27T14:19:20.000Z",
      "updatedAt": "2016-04-29T17:02:15.000Z",
      "device_type": 1
    },
    "data_type": {
      "id": 1,
      "data_type": "float",
      "createdAt": "2016-04-27T22:37:48.000Z",
      "updatedAt": "2016-04-29T17:02:15.000Z"
    },
    "id": 1,
    "name": "Test Tag",
    "tag": "testtag.val",
    "description": "changed description",
    "change_threshold": 5,
    "guarantee_sec": 3600,
    "map_function": null,
    "units": "degF",
    "minExpected": 0,
    "maxExpected": 1000,
    "createdAt": "2016-04-27T14:21:37.000Z",
    "updatedAt": "2016-04-29T16:17:02.000Z"
  },
  {
    "tag_class": {
      "id": 5,
```



```

    "class_type": "custom",
    "description": "User-Selected Tags",
    "createdAt": "2016-04-27T14:17:54.000Z",
    "updatedAt": "2016-04-27T14:17:54.000Z"
  },
  "deviceID": {
    "id": 1,
    "address": "10.10.10.3",
    "createdAt": "2016-04-27T14:19:20.000Z",
    "updatedAt": "2016-04-29T17:02:15.000Z",
    "device_type": 1
  },
  "data_type": {
    "id": 1,
    "data_type": "float",
    "createdAt": "2016-04-27T22:37:48.000Z",
    "updatedAt": "2016-04-29T17:02:15.000Z"
  },
  "id": 3,
  "name": "Tag 3 Test",
  "tag": "Tag3.val",
  "description": "This is another randomly generated tag",
  "change_threshold": 5,
  "guarantee_sec": 3600,
  "map_function": null,
  "units": "lbs",
  "minExpected": 0,
  "maxExpected": 100,
  "createdAt": "2016-04-29T17:00:54.000Z",
  "updatedAt": "2016-04-29T17:02:15.000Z"
}
]

```

GET /tag/

Returns only the information about the specified tag

Returns:

```
[
  {
    "tag_class": {
      "id": 5,
      "class_type": "custom",
      "description": "User-Selected Tags",
      "createdAt": "2016-04-27T14:17:54.000Z",
      "updatedAt": "2016-04-27T14:17:54.000Z"
    },
    "deviceID": {
      "id": 1,
      "address": "10.10.10.3",
      "createdAt": "2016-04-27T14:19:20.000Z",
      "updatedAt": "2016-04-29T17:02:15.000Z",
      "device_type": 1
    },
    "data_type": {
      "id": 1,
      "data_type": "float",
      "createdAt": "2016-04-27T22:37:48.000Z",
      "updatedAt": "2016-04-29T17:02:15.000Z"
    },
    "id": 1,
    "name": "Test Tag",
    "tag": "testtag.val",
    "description": "changed description",
    "change_threshold": 5,
    "guarantee_sec": 3600,
    "map_function": null,
    "units": "degF",
    "minExpected": 0,
    "maxExpected": 1000,
    "createdAt": "2016-04-27T14:21:37.000Z",
    "updatedAt": "2016-04-29T16:17:02.000Z"
  }
]
```

POST /tag

Adds a new tag to the scan list

Parameters

```

{
  tag: < Tag name in PLC >,
  name: < Vanity Name >,
  tag_class: 5,
  deviceID: < device ID of the source device >,
  units: < units >,
  minExpected: < Minimum Expected value (used for gauges) >,
  maxExpected: < Maximum Expected value (used for gauges) >,
  guarantee_sec: < Maximum time between stored measurements >,
  change_threshold: < amount change to require data to be stored >,
  description: < description of the tag >,
  data_type: < data type ID for the tag >
}

```

UPDATE /tag/

Updates a tag in the scan list

Parameters

```

{
  tag: < Tag name in PLC >,
  name: < Vanity Name >,
  tag_class: 5,
  deviceID: < device ID of the source device >,
  units: < units >,
  minExpected: < Minimum Expected value (used for gauges) >,
  maxExpected: < Maximum Expected value (used for gauges) >,
  guarantee_sec: < Maximum time between stored measurements >,
  change_threshold: < amount change to require data to be stored >,
  description: < description of the tag >,
  data_type: < data type ID for the tag >
}

```

DELETE /tag/

Deletes a tag in the scan list

Tag Values

GET /tag_val

Gets all values in the database

Returns:

```
[
  {
    "tagID": {
      "id": 1,
      "name": "Test Tag",
      "tag": "testtag.val",
      "description": "changed description",
      "change_threshold": 5,
      "guarantee_sec": 3600,
      "map_function": null,
      "units": "degF",
      "minExpected": 0,
      "maxExpected": 1000,
      "createdAt": "2016-04-27T14:21:37.000Z",
      "updatedAt": "2016-04-29T16:17:02.000Z",
      "tag_class": 5,
      "deviceID": 1,
      "data_type": 1
    },
    "id": 1,
    "val": 3.14159,
    "createdAt": "2016-04-27T14:26:58.000Z",
    "updatedAt": "2016-04-27T14:26:58.000Z"
  },
  {
    "tagID": {
      "id": 1,
      "name": "Test Tag",
      "tag": "testtag.val",
      "description": "changed description",
      "change_threshold": 5,
      "guarantee_sec": 3600,
      "map_function": null,
      "units": "degF",
      "minExpected": 0,
      "maxExpected": 1000,
      "createdAt": "2016-04-27T14:21:37.000Z",
      "updatedAt": "2016-04-29T16:17:02.000Z",
      "tag_class": 5,
      "deviceID": 1,
      "data_type": 1
    },
    "id": 2,
    "val": 12.2334,
    "createdAt": "2016-04-27T22:40:38.000Z",
```

```
"updatedAt": "2016-04-27T22:40:38.000Z"
},
{
  "tagID": {
    "id": 1,
    "name": "Test Tag",
    "tag": "testtag.val",
    "description": "changed description",
    "change_threshold": 5,
    "guarantee_sec": 3600,
    "map_function": null,
    "units": "degF",
    "minExpected": 0,
    "maxExpected": 1000,
    "createdAt": "2016-04-27T14:21:37.000Z",
    "updatedAt": "2016-04-29T16:17:02.000Z",
    "tag_class": 5,
    "deviceID": 1,
    "data_type": 1
  },
  "id": 3,
  "val": 10.6255,
  "createdAt": "2016-04-27T22:40:48.000Z",
  "updatedAt": "2016-04-27T22:40:48.000Z"
},
{
  "tagID": {
    "id": 1,
    "name": "Test Tag",
    "tag": "testtag.val",
    "description": "changed description",
    "change_threshold": 5,
    "guarantee_sec": 3600,
    "map_function": null,
    "units": "degF",
    "minExpected": 0,
    "maxExpected": 1000,
    "createdAt": "2016-04-27T14:21:37.000Z",
    "updatedAt": "2016-04-29T16:17:02.000Z",
    "tag_class": 5,
    "deviceID": 1,
    "data_type": 1
  },
  "id": 4,
  "val": 6.06918,
  "createdAt": "2016-04-27T22:40:58.000Z",
  "updatedAt": "2016-04-27T22:40:58.000Z"
```

```
}  
]
```

GET /tag_val/latest

Returns the latest stored value for all tags

Returns:

```
[  
  {  
    "id": 13931,  
    "dtime": "2016-05-02T23:04:17.000Z",  
    "t_id": 1,  
    "name": "Test Tag",  
    "tag": "testtag.val",  
    "val": 162.312,  
    "units": "degF",  
    "description": "changed description",  
    "minExpected": 0,  
    "maxExpected": 1000  
  },  
  {  
    "id": 13932,  
    "dtime": "2016-05-02T23:04:17.000Z",  
    "t_id": 3,  
    "name": "Tag 3 Test",  
    "tag": "Tag3.val",  
    "val": -2.93347,  
    "units": "lbs",  
    "description": "This is another randomly generated tag",  
    "minExpected": 0,  
    "maxExpected": 100  
  }  
]
```

DELETE /tag_val/

Deletes value for the specified ID.

Web API Advanced Parameters

Reference the [Waterline Documentation](#) to learn about the advanced query parameters.