Intel® Internet of Things (IoT) Developer Kit

IoT Cloud-Based Analytics User Guide

September 2014

Reference Number: 000000-000US

Table of Contents

LEGAL

1.0 Introduction

Intel provides a cloud-based analytics system for the Internet-of-Things (IoT) that includes resources for the collection and analysis of sensor data that the Intel® IoT Developer Kit provides. Using this service, Intel Galileo/Edison device developers have the ability to jump-start data acquisition and analysis without having to invest in large-scale storage and processing capacity. Intel provides this guide to allow new and experienced IoT developers to access and take advantage of this valuable resource.

1.1.Considerations and Warnings

The IoT Analytics site is provided as a service to the IoT development community. Keep the following considerations and warnings in mind when using the site:

- Using the `iotkit-admin` program to change your device ID (from the default of `MAC address`) does not make the device anonymous. The device IP address will still be available to Intel and the IoT Analytics site. In addition, if a device collects and submits GPS coordinates, Intel will have access to that information, as well. Please carefully review the Terms and Conditions and the Intel Online Privacy Notice.
- Do not connect to the Galileo development board to any kind of actuator that could cause harm or damage. It is possible that users could unintentionally activate these devices.

2.0 Accessing and Using IoT Analytic Functionality

2.1. IoT Analytics: Login and Account Creation

To begin using the IoT Analytic cloud site, create a new administrative user account that will have the ability to register devices, manage alerts, create accounts, and perform other tasks on the site. The screen in Figure 1 appears when the user goes to the IoT Analytics Dashboard website (https://dashboard.us.enableiot.com).

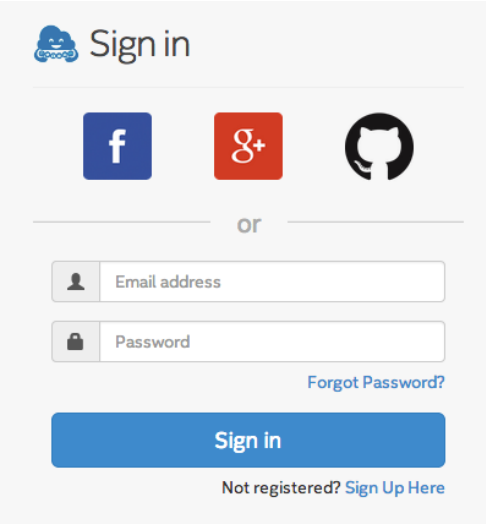

Figure 1. Creating an Account on the IoT Analytics Site

You have the option of Facebook, Google+, or GitHub authentication, or you can create a local account with an email address and password. If you click **Sign Up Here**, the **Create user** page appears (see Figure 2).

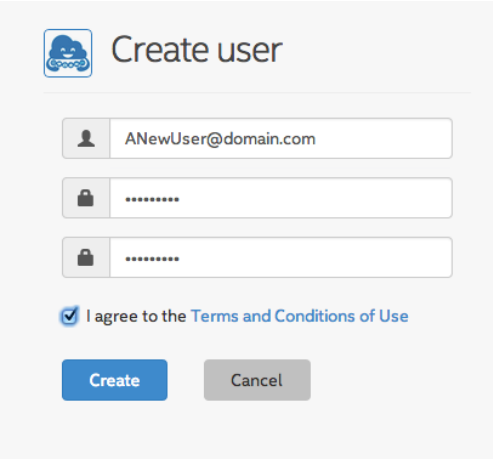

Figure 2. Create a New User

The **Create user** screen requires a valid email address (which is used to send the activation link to complete account creation) and a user-provided password. (There are minimum complexity requirements for the password.) Review the Terms and Conditions, select the **I agree to the Terms and Conditions of Use** check box, and then click **Create**.

After the new user has been created, you are prompted to create an account name (see Figure).

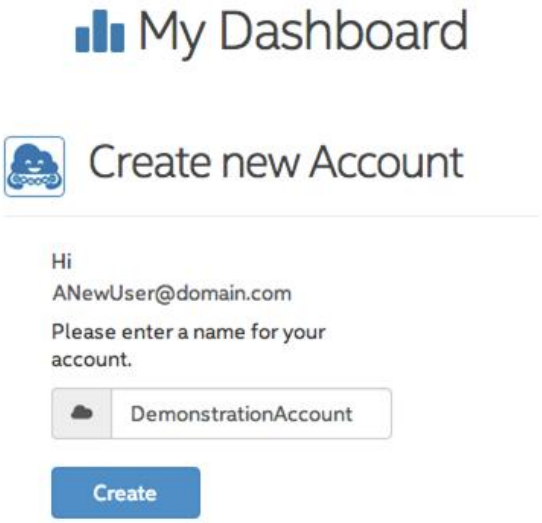

Figure 3. Create a New Account

This name appears on the site when you log in. When you have authenticated, the **My Dashboard** page appears (see Figure). Note the account name in the dashboard header.

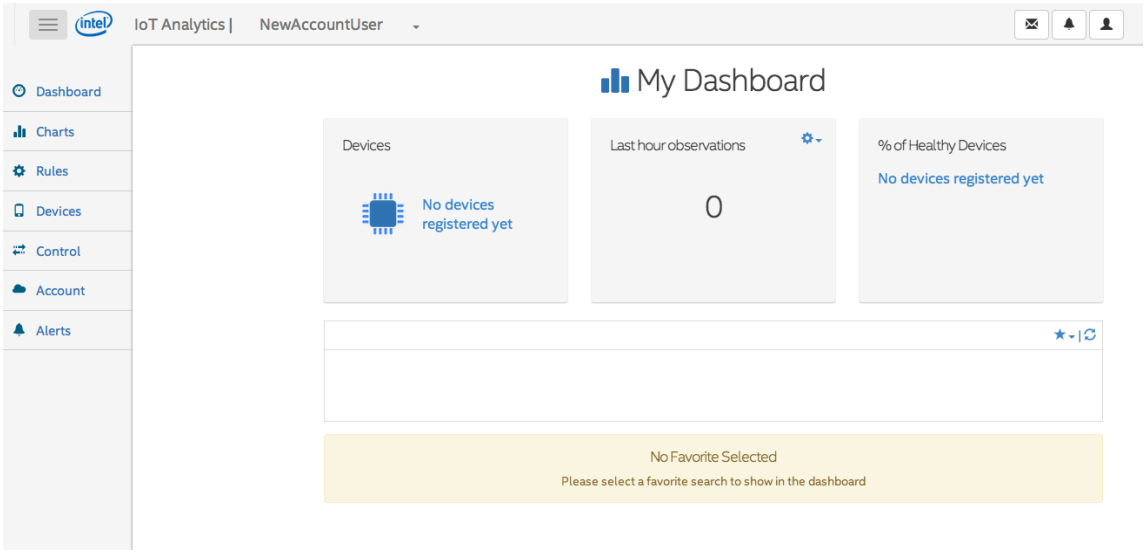

Figure 4. User Dashboard and Navigation Page

My Dashboard provides access to all system functions, including the number of devices, the number of observations received in the past hour, and the percentage of devices that are "healthy," meaning that they have transmitted data recently. You can customize the lower section to show a "favorite" chart, which contains a chart of specific time series from specific devices. This is described more fully in the User Manual.

2.2.Account Administration

Each account that you create on the IoT Analytics site is a separate workspace. You administer the accounts from the **My Account** page (see Figure).

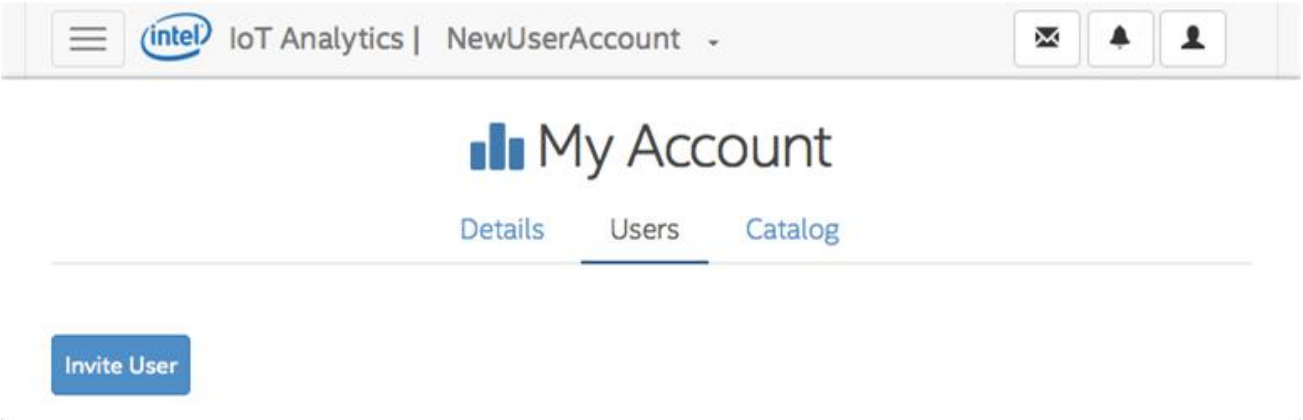

Figure 5. Account Administration

From this page, you can not only invite additional users to participate in the analytics site (by registering their email address) but also view the available catalog of analytic tools and account details.

2.3.IoT Kit Agent

The IoT Gateway Agent abstracts complexities of cloud connectivity, allowing developers to focus on application development and logic for their devices (e.g., sensors, actuators, tags). The Agent transparently implements the necessary message formatting and security during device registration and subsequent data submission. You can download the Agent from the IoT GitHub site.

The IoT Gateway Agent and Administrator are written in JavaScript and can run on any computer for the purpose of setting up an account or to experiment with the interface. When you have successfully installed the Node.js program, you can complete Agent installation by extracting the downloaded GitHub file to a working directory (`iotkit-agent-master` is the default) and run the following installer command to install all of the necessary libraries:

```
npm install
npm install forever
```

Note that if you installed the Agent yourself (without using the installer script), the Agent will not be in the system path. In this case, use the `cd` (change directory) command to go to the working directory that you created when you extracted the `iotkit-agent-master.zip` file and preface all commands with the prefix `./` (this is the standard Bash shell indicator that the following program is to be run from the local directory). Note also that the current version of the Agent requires a directory on the Windows platform of `C:\tmp` for the `agent.log` file. If an error occurs from any of the `iotkit-admin.js` commands (e.g., `Error: ENOENT`), make sure this directory exists, and try the command again.

To verify that the Agent was installed correctly, run the following command from the command line:

```
iotkit-admin.js test
```

This command should produce the following message:

```
2014-08-21T21:50:54.062Z - info: Trying to connect to host ...
2014-08-21T21:50:54.570Z - info: Connected to dashboard.us.enableiot.com
2014-08-21T21:50:54.571Z - info: Environment: prod
2014-08-21T21:50:54.571Z - info: Build: 0.9.1237-fixed
```

If you are attempting the connection from behind a firewall with a proxy server, add the proxy server to the `iotkit-admin` configuration:

```
iotkit-admin.js proxy http://proxy.company.com 8080
```

Repeat the above test command. If you are still having connection problems, you will need to correct them before you can proceed with the installation.

After you have downloaded and extracted the IoT Gateway Agent and loaded it on an appropriate device, several administrative commands are available to check the installation (see the GitHub notes for more information). Note that some Galileo/Edison devices come with the Gateway Agent preinstalled. To determine whether this is the case, run the system service `status` command on the device:

```
systemctl status iotkit-agent
```

If a running status is returned, then the Agent is already installed and running. To stop the Agent for configuration, run the following command:

```
systemctl stop iotkit-agent
```

You will restart the Gateway service in a later step after you have activated it from the IoT Analytics site.

2.3.1. Getting and Changing the Device ID

You should know your device ID because you may need it to find the device in the device list if you have more than one device. You can display the device ID with the following command:

```
iotkit-admin device-id
```

You can change this device ID (if desired) by running the following command.

Note: Changing the ID does not provide anonymity. The source TCP/IP address will still be available to the cloud server.

```
iotkit-admin set-device-id «device_id»
```

It is also important to note that the device ID activation step returns a security token to encode the device ID, which is necessary for correct operation of IoT Analytics security. The device ID should not be changed after activation.

2.3.2. Activating the Device

To complete the device activation, you must supply a specific account activation code from the IoT Analytics account **Details** page (see Figure).

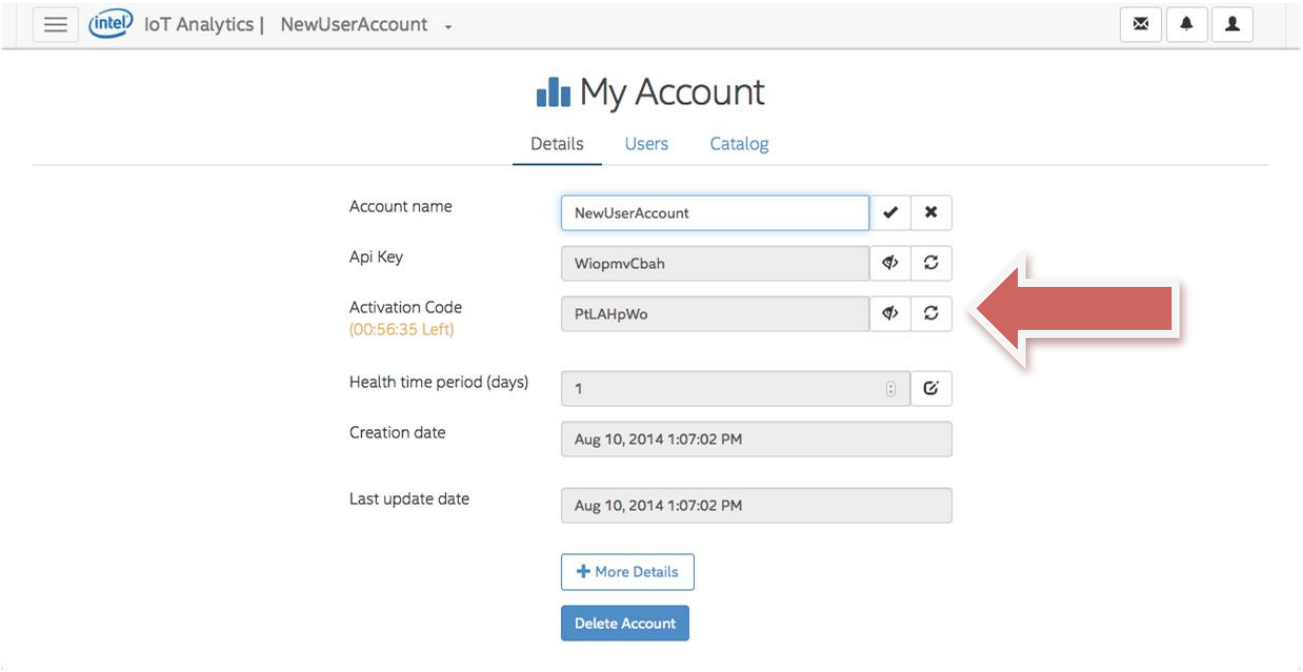

Figure 6. My Account Agent Activation Code

The code is valid for only 60 minutes. After that, you will have to regenerate the code by clicking the button indicated in Figure 6. The `iotkit-admin` Agent supplies the code using the following command:

```
iotkit-admin activate «activation_code»
```

The activation process creates the device record in the cloud, associates the device with the account (where the activation code came from), and provides the device Agent with security credentials.

Note: When a device is activated and associated with a particular user, it is available to all of the users on the account. If you need to reactivate a device on the IoT Analytics site, delete the

previous device registration by logging in to May Dashboard, and then clicking **Devices** in the navigation pane (see Figure).

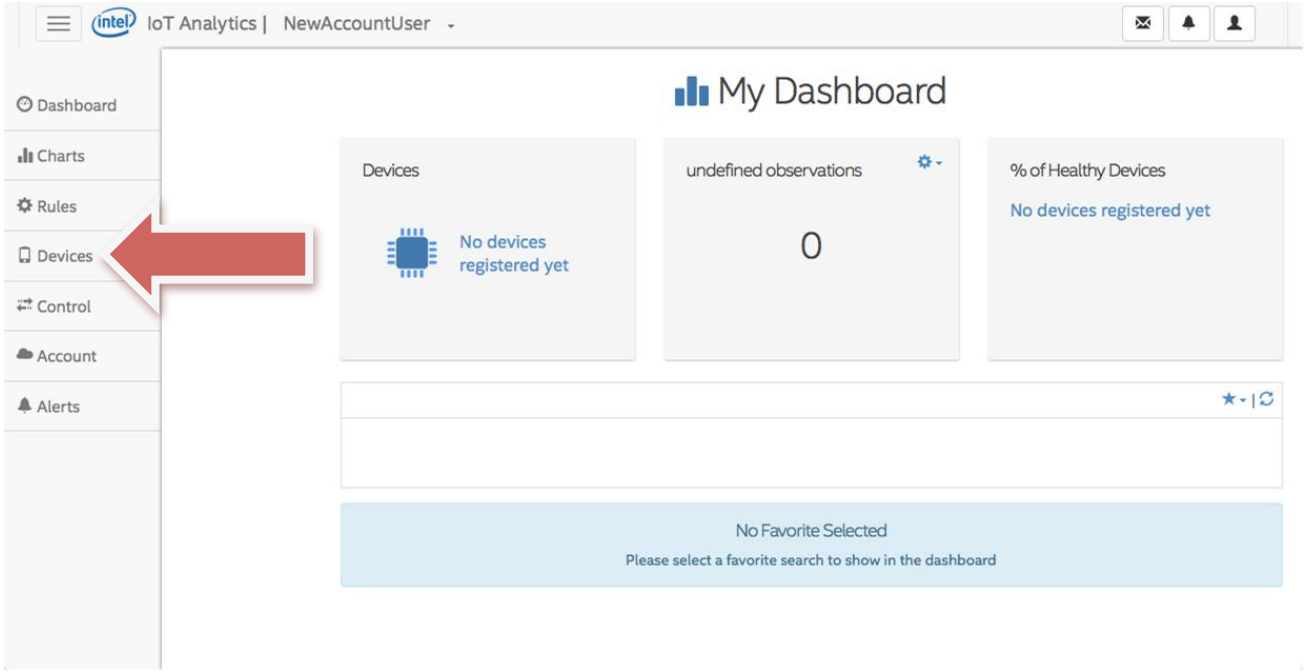

Figure 7. Accessing the Devices page

Doing so brings up the **My Devices** page, where you will be able to see the newly activated device. Click the Device ID (as shown in Figure 8) to see the device's detailed information.

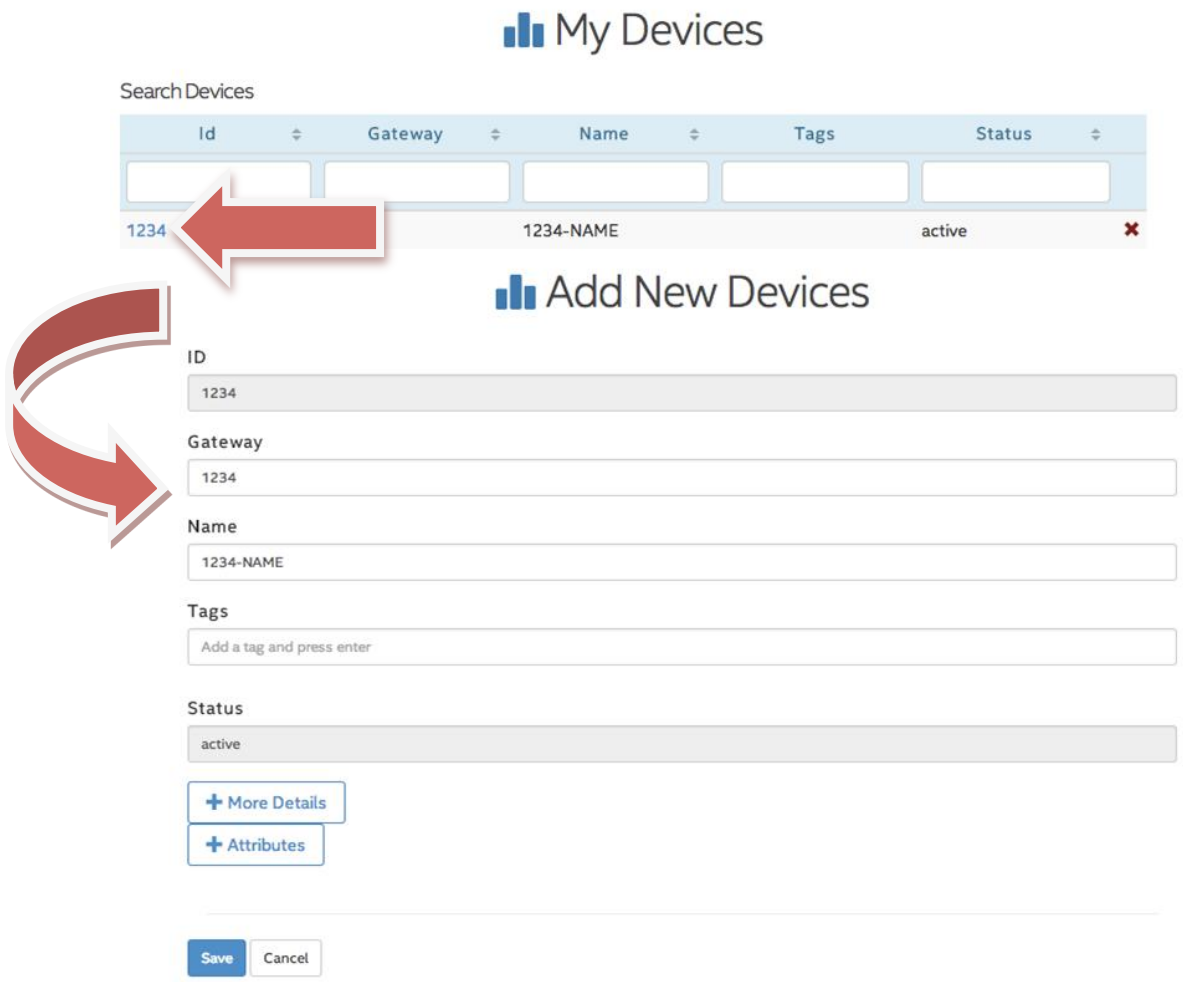

Figure 8. The My Devices Page and the Add New Devices Dialog Box

If the device has already been registered, you must reset the code, and then reinitialize the `iotkit-admin` before attempting to activate the device:

```
iotkit-admin initialize
iotkit-admin activate «device_activation_code»
```

At this point, the device should be activated on the IoT Analytics site. You can confirm this by logging in to My Dashboard and verifying that the device status is active, as shown in Figure .

2.3.3. Starting the Agent

When the device (or devices) has been registered, start the `iotkit-agent` with the following system command:

```
systemctl start iotkit-agent
```

2.3.4. Stopping the Agent

To stop a running Agent (e.g., to configure additional devices), run the following command:

```
systemctl stop iotkit-agent
```

2.3.5. Sensor Registration and Data Submission

Sensor data from a registered device, which you collect using the various Edison and Galileo developer board interfaces connected to one or more monitoring sensors, is sent to the IoT Analytics site when you register each sensor to an appropriate Component type from the Component Catalog. You access the Catalog from My Account, as shown in Figure .

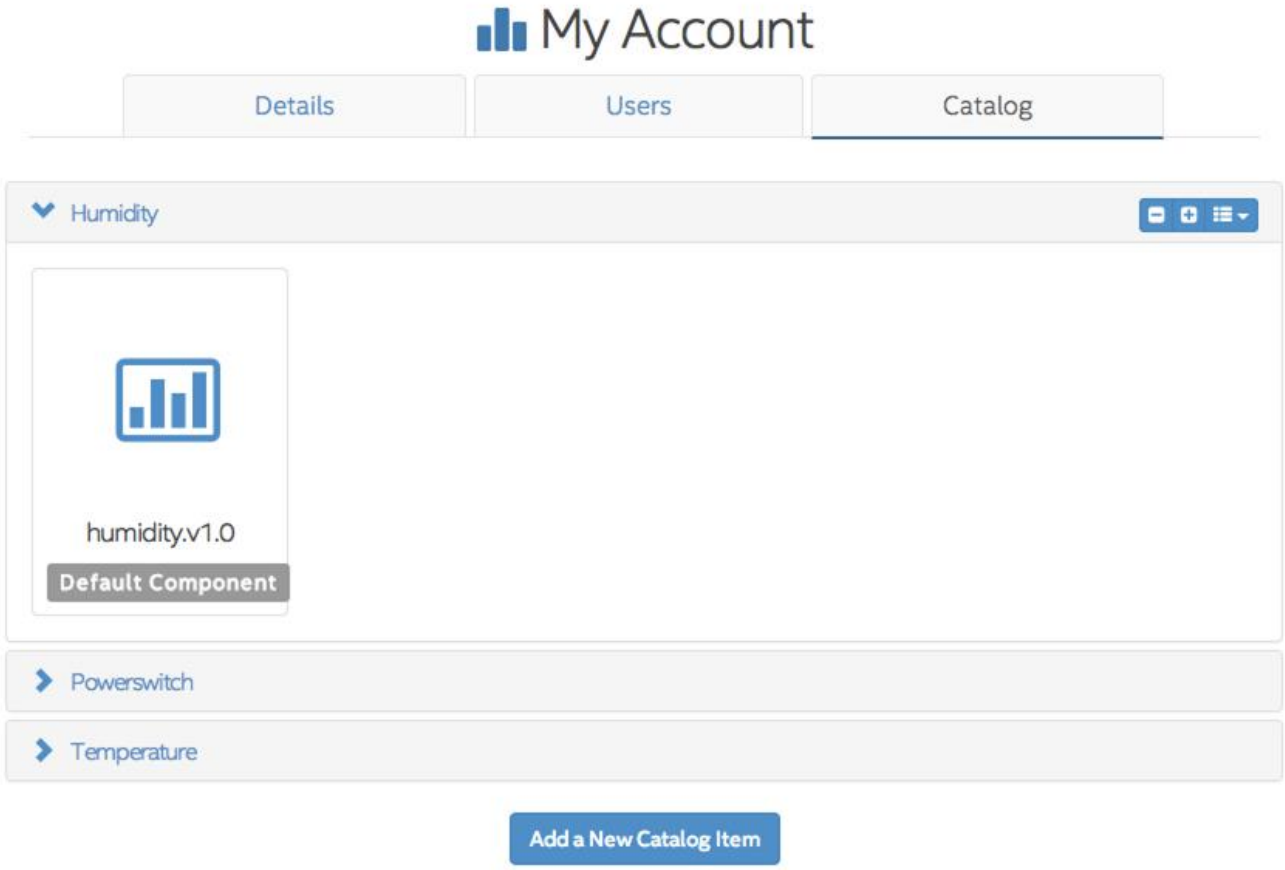

Figure 9. The Component Catalog from the My Account Page

There are three default components: humidity, power, and temperature. You can also add Catalog items from the **Add a New Catalog Item** dialog box.

You perform the registration step only once for each new sensor—for example:

```
iotkit-admin.js register temp temperature.v1.0
```

You can find the `category-id` directly in the IoT Analytics user interface or via the following command:

```
iotkit-admin catalog
```

When the sensor has been registered to an appropriate Component, you can transmit observation data to the IoT Analytics site. To verify that the sensor has been properly registered, run the `observation` command to send individual data elements to the IoT Analytics site:

```
iotkit-admin observation temp 35
iotkit-admin observation temp 30
```

These commands are sent to the IoT Analytics site to sensor data points for the `temperature.v1.0` Component (which you registered as `temp` in the previous step). Switch back to the website, and select **Chart** from the navigation pane to select the device and Component and see the newly sent data elements (Figure).

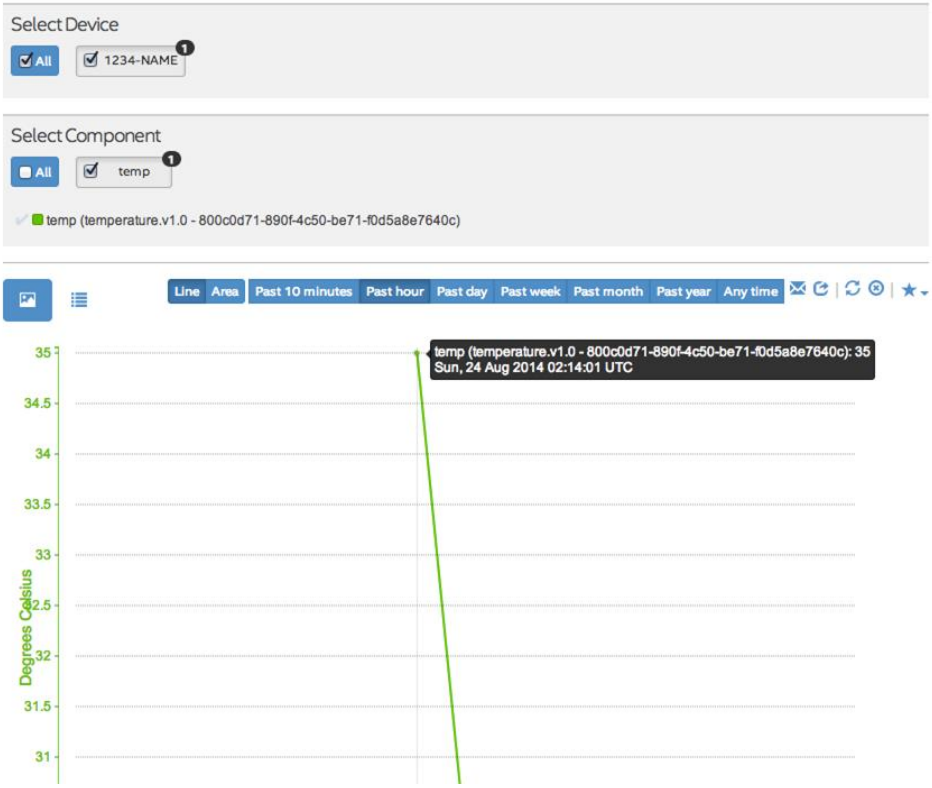

Figure 10. Chart for the Selected Device and Component

2.4.Sending Multiple Data Packets

The `iotkit-admin observation` command is intended to facilitate testing, not for continuous data capture. There are two additional ways to send observation data to the IoT Analytics site:

- A client can use the Representational State Transfer (REST)-ful web service interface documented at https://github.com/enableiot/iotkit-agent/wiki/Api-Home.
- A client can send User Datagram Protocol (UDP) packets to the Agent. The Agent converts these packets into REST calls and sends the request on behalf of the client.

This section explains how to use the Agent.

If the `iotkit-agent` is not running, use the `start` command:

```
systemctl start iotkit-agent
```

You can send UDP-formatted messages to the Agent through the device localhost on port 41234 (e.g., `UDP://localhost:41234`). First, create a JavaScript Object Notation–formatted document, where the key `n` is the name of a time series and `v` is the value:

```
{ "n": "<sensor name>", "v": "<value>" }

Examples:
{ "n": "temp", "v": 21.0 }
{ "n": "humidity", "v": 71.5}
```

Most programming languages can transmit this message, but using the `nc` program on Mac OS X and Linux, you do so as follows:

```
echo -n '{"n": "temp", "v": 21.0}' | nc -u -w1 127.0.0.1 41234
```

This command uses the Agent (which is listening on port 41234 for the UDP messages) to convert the observation to a REST web service call and forward it to the IoT Analytics site.

3.0 Install the IoT Kit Arduino Library and Samples

You can download samples that illustrate how to connect with the `iotkit-agent` from GitHub at https://github.com/enableiot/iotkit-samples.

These samples assume that you have already installed (or have access to) the `iotkit-agent`, which supports UDP and TCP. To install the `iotkit-agent`, complete these steps:

1. From the Arduino integrated development environment, click **Sketch > Import Library > Add Library**.
2. Browse to the `IoTkit.zip` file.
3. Click **Open**.

When installation is complete, you should see IoTkit listed under **File > Examples**.

4.0 Start Working with Cloud

You are now ready to write Arduino scripts to send data to the cloud. Write the scripts on your computer, and then download them to the board and run them. After a few minutes, check My Dashboard to view the data.