**InfluxDB section:**

To store the data, we use the open source time-series database system InfuxDB. A time-series database is ideal because we want data to be constantly added to the database over time. Thus, it makes sense for the data to be indexed by time and use a database that is optimized for this usage. One of the reasons this particular database was chosen is that it provides integration with data analytics and visualization software like Grafana.

The database was created on a laptop which is connected to the same network as the IoT end device. As data is generated, it is sent through HTTP requests with the format shown in figure 4. It is necessary to concatenate an “i” to the number so InfluxDB knows to read that number as an integer. Without this, InfluxDB reads the number as a float by default, which cannot be properly processed later in Grafana.

When an error occurs and the system shuts down because it doesn't have enough information to act properly, it sends a Boolean value of failure to a special column of the database that triggers an alarm to inform the user.

**Grafana section:**

The Grafana software provides a solution for the visualization of the stored time series data. A dashboard is created with a panel for every graph the user wants to display. Here, the user can access and easily interpret the complete history of the monitored environmental parameters.

Setting up Grafana had its challenges. For each panel, we had to enter the metrics for the graph which consists of a query to the database. For testing when the data failed to show up on the graph, I was able to determine whether or not the data reached Grafana through looking at the results of the Query Inspector. When it didn’t, I checked whether or not the data reached the database by searching the same query directly in the database via the command line.

Then, Grafana allows users to customize the label and scale of the axes, how the data is displayed, what time period of data gets displayed, and more. Another helpful feature is the alert. This is useful for the error panel. On that panel, we display a graph which shows each instance that a failure occurs. Since we probably want the user to be notified when this happens, we configure alerts to alert the user, so they know when something is going wrong, like a sensor malfunctioning.