**Overview**

Detecting Anomalies can be a difficult task and especially in the case of labeled datasets due to some level of human bias introduced while labeling the final product as anomalous or good. These giant manufacturing systems need to be monitored every 10 milliseconds to capture their behavior which brings in lots of information and what we call the *Industrial IoT* (IIOT). Also, hardly a manufacturer wants to create an anomalous product. Hence, the anomalies are like a needle in a haystack which renders the dataset that is significantly Imbalanced.

Capturing such a dataset using a machine learning model and making the model generalize can be fun. In this competition, we bring such a use-case from one of India's leading manufacturers of wafers(semiconductors). The dataset collected was anonymized to hide the feature names, also there are **1558** features that would require some serious domain knowledge to understand them.

However, In the era of Deep Learning, we are challenging the data science community to come up with an anomaly detection model that can generalize well on the unseen set of data**(Test data)**. In this hackathon, you will be creating a machine learning/ deep learning model to classify the anomalies correctly using Area under the curve(AUC) as metric.

**Dataset Description:**

* **Train.csv - 1763 rows x 1559 columns**
* **Test.csv - 756 rows x 1558 columns**
* **Sample Submission.csv -**Please check the **Evaluation** section for more details on how to generate a valid submission

**Attribute Description:**

* **Feature\_1 - Feature\_1558 - Represents the various attributes that were collected from the manufacturing machine**
* **Class - (0 or 1) - Represents Good/Anomalous class labels for the products**

***Skills:***

* ***High Dimensionality Data, Overfitting-vs-Underfitting***
* ***Advanced Classification Techniques, Gradient Boosting, Neural Nets, etc***
* ***Feature engineering, Feature Selection Techniques***
* ***Optimizing Area under the curve(AUC) to generalize well on unseen data***