Rapid Prototyping of IoT Applications for the Industry

Katalin Ferencz
Department of Electrical Engineering
Faculty of Technical and Human Sciences, Sapientia Hungarian University of Transylvania
Tîrgu Mureș, Romania
ferenczkatalin@ms.sapientia.ro

József Domokos
Department of Electrical Engineering
Faculty of Technical and Human Sciences, Sapientia Hungarian University of Transylvania
Tîrgu Mureș, Romania
domi@ms.sapientia.ro

***Abstract*— In this article a novel approach to rapid IoT application prototype design and development is presented using an existing experimental dataset and a functional model. Using the existing data, we populate our NoSQL Apache Cassandra database cluster with legacy data and generate similar data using a python code, considering the Mosquitto MQTT protocol implementation and Node-RED node.js development environment. Using Node-RED, we display the data already collected, and dynamically create new data that can be monitored in real-time in the provided dashboard. The possibilities and utility of this approach are explored in the article, and a simple prototype application for modeling the open access Combined Cycle Power Plant (CCPP) dataset provided by the UCI Machine Learning Repository is presented to prove the efficiency and rapidity of IoT application development. The presented system development approach can be used in industrial environment for rapid development of IIoT applications.**

Keywords— Industry 4.0, IIoT, prototyping, Node-RED, MQTT, data storage

I. INTRODUCTION

The world around us is changing very fast, and as a result we are witnessing a very strong wave of technical development. One of the most important features of the 21th century is digitalization. We equip our environment with smart devices, so we collect and store data about everything. As a result, we can control the temperature and lighting of our home remotely, even from a smart phone, or even instruct our coffee machine to wait for the steaming coffee to reach the kitchen. In this case we can talk about smart home. Many digital innovations have made our daily lives more comfortable, shortened certain activities, allowing us to spend more time on other tasks. But what did we have to do to have this luxury? First of all, we had to bring smart devices, IoT (Internet of Things), into our daily lives. For this reason, we surround ourselves with sensors that collect and transmit the necessary data to a broker or IoT hub to store, process and display the data. If we buy unfinished systems we build a prototype that requires a smaller amount of money based on our preliminary plans and then, if necessary, rebuild it several times, and when it proves successful we buy and build the complete system tool kit.

The integration of smart devices has changed, transformed and generates positive results in our daily lives, and based on this model we need to integrate these tools in industry as well, so we will talk about Industry 4.0. If you look at it in detail, the same need arises in the industry as in the ordinary man, that is, work must be continuous, comfortable and efficient, which can bring about cost savings and create new values. Based on the above, we also need to choose the industry where we want to apply the new technology, and then create a prototype and model of it at a low cost and in a short time. This allows us to measure the impact this innovation will have on overall production, the cost of deploying the entire system, and whether it will provide greater profit in the long run than the current system. Analysts say that Industry 4.0 is transforming the market. But how is Industry 4.0 transforming business? And the question is what added value can the industry have in using smart devices?

This article examines the potential of industrial IoT application prototyping and its potential effects on the market. Using the data from an existing database we have created a prototype that can be used to test the complexity of creating such a prototype. To build a prototype, we use the data set provided by the UCI Machine Learning Repository [1] and generate our own virtual data based on it. We use a NoSQL Apache Cassandra database cluster to store data and a Node-RED development environment for modeling and rendering. The paper is structured as follows: after the first introductory section, related-works are presented in section II; section III bring in focus the connections between industry and IoT; section IV presents Node-RED; section V presents an architectural and functional overview of the designed and developed system; after conclusions the following sections are the ordinary acknowledgements and references.

II. RELATED-WORKS

Recognizing that the integration of their IIoT (Industrial IoT) tools and the recognition and exploitation of their positive benefits are becoming a key factor in the industry as well, more and more research have been conducted in different areas and more scientific papers were published. In the following, let us examine in detail the results of the study of the selected data set, further writings on predictive maintenance, and also fault detection and prediction.

The open access Combined Cycle Power Plant (CCPP) dataset provided by the UCI Machine Learning Repository [1] contains 6 years of data, which is 9568 data points measured on 674 different days. Based on the dataset, Tüfecki published two articles [2][3] on the results of his

978-1-7281-7166-1/20/$31.00 ©2020 IEEE

Authorized licensed use limited to: University of Melbourne. Downloaded on July 26,2020 at 15:10:17 UTC from IEEE Xplore. Restrictions apply.

study of the dataset. In the papers published in the International Journal of Electrical Power & Energy Systems [2] and at the International Conference on Emerging Trends in Computer and Electronics Engineering [3] authors describes in detail the operation of the combined cycle power plant system, the processed sensor data. The sensor values are recorded every 1 second, but only hourly averages are displayed in the processed data set. The relationship between the change of the sensor values and the electrical energy output is described in detail. The aim of Tüfeck's research was to examine and compare the operation of some machine learning regression methods to develop a more accurate predictive model. Examine whether the CCPP can predict the electrical performance of an hourly full load system based on the existing data set.

Zhe Li, in an article in Intelligent Predictive Maintenance for Fault Diagnosis and Prediction in Machine Centers - Industry 4.0 Scenario [4], says that the detection and prediction of machine failures has become a very important topic of research and development over the past decade, we design and use in industrial production. The paper presents data mining approaches (the collection, classification and induction) to illustrate the importance, capabilities and complex frame of predictive maintenance. In an article for the IEEE ICDSDIS 2015 conference, Cheng Wang [5] also examines the potential and benefits of IoT data collected in the manufacturing industry in fault diagnosis and prediction.

In a short presentation by SYS TEC Electronic, Node-RED in industrial applications [6], the company representative presents the potential of their industrial products to integrate IoT technologies easily in the industrial environment. This document provides information that the Node-RED modular development environment can be connected to industrial devices by simple steps, using the MQTT (Message Queue Transfer Protocol) communication protocol. In addition, we get information that we can easily transfer the data we collect to Cloud storage and emphasize Node-Red's capabilities for everyday (Smart Home) and industrial (Smart Industry) use.

III. INDUSTRY AND IOT

The industry as we known it is going through a transformation. Until the start of the 4th Industrial Revolution, industry was characterized by strict and rigid standards that provided a sense of security. Unfortunately, these qualities result in a slow development process and adaptability. With these in mind we may have thought the use of smart devices in industry, but that's the reality. More and more industries are trying to integrate smart devices (such as sensors) or technologies into their manufacturing processes in the hope that they can do more profitable business or even produce value-added products. Thanks to these smart devices, or IoT devices, it is possible to collect large amounts of data that can be analyzed to optimize production, machine utilization and predict the need to replace a component, called predictive maintenance. That is, manufacturers can increase productivity and efficiency with customized digital solutions. Santhosh's study [7] also shows exactly what impact IoT has on smart manufacturing.

The integration of IIoT devices also brings with him the application of new technologies. The collected large amounts of data of various types, known as big data, cannot be stored in classical relational SQL databases, resulting the need of NoSQL-type database usage such as the Apache Cassandra [8] database. With these databases it is possible to store the data not on your own local servers which require a lot of financial investment, but you can rent Cloud server parks, where you can expand your storage without interruption.

What we do with the amount of data we collect is important. If you store it, it will largely increase your spending, but if you analyze it with different algorithms that can provide you with analytics, gathering data will add value to the manufacturer and help you make informed and real-time decisions. Analyzes require machine learning algorithms. Depending on the kind of analysis we need, there are many algorithms that just need to be customized, such as Principal Component Analysis (PCA) / SVD, Constrained Linear Regression, Feed-Forward Neural Networks, Recurrent Neural Networks (RNNs), etc. [9]

Thanks to the technologies mentioned above, there is an increase in manufacturing quality, as smart devices / machines are accurate, their use reduces human contact and error which increases quality and safety. Machines are taking up space in production, that is, they take on tasks that can be reprogrammed, machines can do. In this way, the human factor is displaced from the production lines, greater accuracy is achieved and the role of people in factories is changed. As a result, you will need more office workers (data analysts, system designers, machine controllers, etc.) than line workers.

Prototyping can be seen as a form of problem solving that offers several benefits along with its use, such as testing and rapid implementation of new capabilities, market advantage by helping the product get to market quickly and helping to test a variety of solutions.

For a new software product, final production can be very time consuming and also expensive, but a sample made using a third-party development environment can be done more quickly and is more cost effective. The prototype application can be easily re-designed multiple times and can be developed at low cost. With the introduction of a new technology, e.g. IIoT solutions, it's an important step to build a prototype, see if your idea is feasible, and you need to invest in very expensive software licenses. Is there any return on investment in the long run? In examining these aspects, our prototype can provide an answer.

As mentioned above, an important feature of prototyping is its cost-effective implementation. Product designing has the problem of learning the various licenses and complicated software that may not be needed during the design process, so we have wasted so much money. According to this, the goal is to look for software (free) that can be used for the purpose and can replace the licensed versions in the design process. An example of such an environment is the Node-RED Node.js development environment which provides an easy way to connect different industry sensors, devices, databases like modules on the interface and perform various operations or data visualization tasks.

Furthermore, in order to make a prototype, it is necessary to simulate machines, processes. Thanks to today's advanced technologies, it is possible to model the machine or process by software, or even generate real sensor data using software applications. Thanks to this we do not have to execute real machines or complex processes, thus generating high costs,

Authorized licensed use limited to: University of Melbourne. Downloaded on July 26,2020 at 15:10:17 UTC from IEEE Xplore. Restrictions apply.

but can simulate their operation as desired. Here too, the Node-RED development environment can be a useful and cost-effective help for us.

IV. NODE-RED

Node-RED is an open source JavaScript based Node.js development environment developed by IBM and used to develop IoT systems. [10] This is a flow-based programming tool that is capable of modeling different processes and supporting information flow. Node-RED consist of three panels: node panel, flow panel and info & debug panel and it also contains a UI dashboard. A very important feature is that it comes from library modules, as well as thousands of modules and streams made by the open source community for download and use. These modules are very easy to integrate into our system and can be used to simulate hardware devices or connect real devices to our own Node-RED development environment, for example. Raspberry Pi, MQTT Broker [11], PLCs, Cloud Interfaces etc. [12] Thanks to its increasing use, we were able to create processes that conform to a number of industry port standards and enable reading and writing of SQL or NoSQL databases. It can replace prototypes used in industry, either TIA Portal [13] or Sysmac Studio [14]. The connection between industrial technologies and Node-RED is already being addressed by a large number of researchers, such as Toc in his 2018 study [15] on the communication between Modbus-OPC UA and Node-RED.

Thanks to the built-in dashboard interface, it is possible to create transparent user interfaces without the need for any special HTML or CSS programming knowledge. Thanks to its simplicity and easy-to-learn system, it is becoming more and more usable in the Smart Industry as we can easily create and configure real-time applications. However, it is only used for prototyping because it has not yet reached the required minimum threshold in the field of security.

V. FUNCTIONAL OVERVIEW

Based on the theoretical background and studies presented above we have developed our own prototype IoT application using the Combined Cycle Power Plant (CCPP) data. Figure 1. shows the architecture of our system.

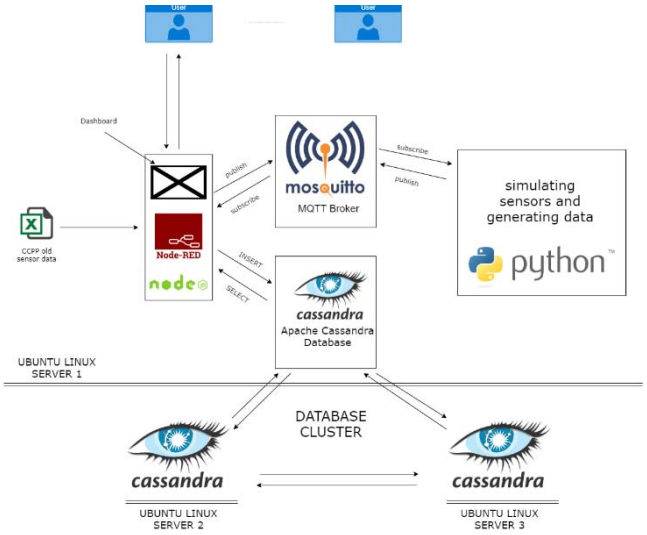

Fig. 1. System architecture

Accordingly, the existing data set was written to the Apache Cassandra database cluster and based on the information presented in Tüfekci's article [2] we have created a simple application in Python that can generate sensor values similar to the existing data set. This eliminates the need for physical tools to reproduce the plant's base load state and allows simulation. Simulated sensor data is generated with the python code and with the help of the Mosquitto MQTT broker implementation the data is transferred in Node-RED and modified if necessary. The generated data was also stored in the Cassandra database cluster, and a Node-RED user interface was created, where we can continuously monitor, modify, and visualize the current values of the sensors. Figure 2. shows exactly how the sensor data looks in the user interface and how we can change their values.

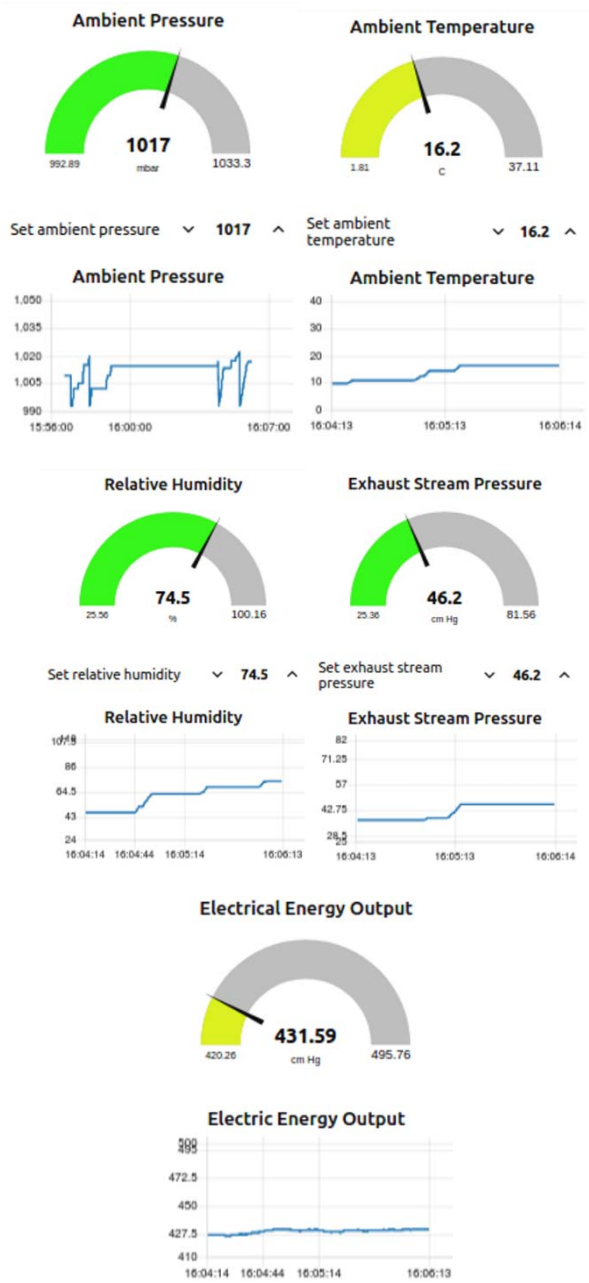

Fig. 2. Generated sensor data monitorisation

In order to import the existing data set containing 9568 data points each with 5 sensor values into the NoSQL Apache Cassandra database cluster using Node-RED, we have created a simple data stream that reads the rows from a file and assigns an id (ascending number) to the database table created for this

Authorized licensed use limited to: University of Melbourne. Downloaded on July 26,2020 at 15:10:17 UTC from IEEE Xplore. Restrictions apply.

purpose (Figure 3.) Given that we want to use this database table sorted by time, we have created the table using the WITH CLUSTERING ORDER BY (id ASC) parameter used in the Cassandra database system, and all data tables with this parameter we will create in a complementary way. Without this, we can't use our data tables sorted by time or time which is important when studying changes in sensor values. In the case of the old data set, it is not important to associate a time stamp with the data as these represent the hourly average for a 6 year period. Existing data set was visualized using the Dashboard module, so we can retrospectively interpret changes in sensor data for 6 years.

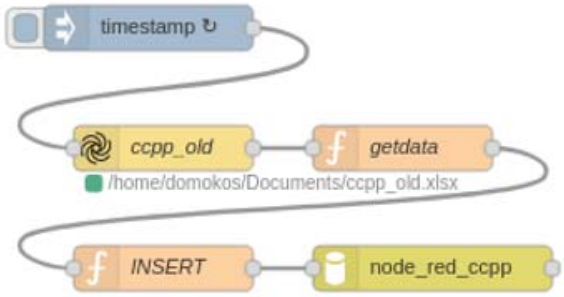

Fig. 3. Import old sensor data from excel to database cluster

Definitions of abbreviations used below:

AT – Ambient Temperature;
AP – Ambient Pressure;
RH – Relative Humidity;
V – Vacuum;
EP – Electrical Power;

An important part of our work is the analysis of the data set and based on the information presented in Tüfekci's article[2], we have created a model in Node-RED that can generate approximately real data, thus simulating the operation of the plant. When generating data, we consider the following:

- when the AT rises, the EP decreases, if the AT decreases, the EP increases;
- if the AP increases, the EP increases, if the AT decreases, the EP decreases;
- if the RH increases, EP increases, if RH decreases, EP decreases;
- when the pressure of the exhaust steam, the V increases, the EP decreases; if V decreases, the EP increases.;

The logic presented above was encoded in JavaScript. The following figure (Figure 4.) shows the case of adjusting the electrical energy output based on the change in ambient pressure value.

We implemented similar JavaScript logic for temperature, relative humidity and exhaust steam pressure.

```
var prev_AP = flow.get("prev_AP_flow_v");
var next_AP = msg.payload;
flow.set("next_AP", next_AP);

if(prev_AP < next_AP){
    var get_ep_flow_v = Number(flow.get("ep_flow_v")) + 0.14;
}
else {
    var get_ep_flow_v = Number(flow.get("ep_flow_v")) - 0.14;
}

flow.set("get_ep_flow_v",get_ep_flow_v);
msg.topic = "sensors/value-EP/control/set";
msg.payload = get_ep_flow_v;
return msg;
```

Fig. 4. Setting the value of electrical energy output as a function of ambient pressure

The first step in temperature effect calculation is to retrieve the previous ambient temperature value and then the new setpoint. Comparison of the two shows that the temperature has been increased or decreased. If we lower it, we will increase the electrical energy output, or if we increase it, we will decrease it. (Figure 5.)

```
var prev_AT = flow.get("prev_AT_flow_v");
var next_AT = msg.payload;
flow.set("next_AT", next_AT);

if(prev_AT < next_AT){
    var get_ep_flow_v = Number(flow.get("ep_flow_v")) - 0.21;
}
else {
    var get_ep_flow_v = Number(flow.get("ep_flow_v")) + 0.21;
}

flow.set("get_ep_flow_v",get_ep_flow_v);
msg.topic = "sensors/value-EP/control/set";
msg.payload = get_ep_flow_v;
return msg;
```

Fig. 5. Setting the value of electrical energy output as a function of ambient temperature

Each sensor has a different value to increase or decrease the electrical energy output. These values was also calculated from Tüfekci's article, because it gave values to one unit, and we converted it to the unit by which we increase or decrease the value of the sensors.

```
var prev_RH = flow.get("prev_RH_flow_v");
var next_RH = msg.payload;
flow.set("next_RH", next_RH);

if(prev_RH < next_RH){
    var get_ep_flow_v = Number(flow.get("ep_flow_v")) + 0.1;
}
else {
    var get_ep_flow_v = Number(flow.get("ep_flow_v")) - 0.1;
}

flow.set("get_ep_flow_v",get_ep_flow_v);
msg.topic = "sensors/value-EP/control/set";
msg.payload = get_ep_flow_v;
return msg;
```

Fig. 6. Setting the value of electrical energy output as a function of relative humidity

Authorized licensed use limited to: University of Melbourne. Downloaded on July 26,2020 at 15:10:17 UTC from IEEE Xplore. Restrictions apply.

```
var prev_V = flow.get("prev_V_flow_v");
var next_V = msg.payload;
flow.set("next_V", next_V);

if(prev_V < next_V){
    var get_ep_flow_v = Number(flow.get("ep_flow_v")) - 0.1;
}
else {
    var get_ep_flow_v = Number(flow.get("ep_flow_v")) + 0.1;
}

flow.set("get_ep_flow_v",get_ep_flow_v);
msg.topic = "sensors/value-EP/control/set";
msg.payload = get_ep_flow_v;
return msg;
```

Fig. 7. Setting the value of electrical energy output as a function of pressure of the exhaust steam

The simulation data generated by the five sensors was generated using a python code which runs continuously and is connected by the Mosquitto MQTT broker to the Node-RED development environment and to the user interface that we have created for this purpose. In the code, we provide the MQTT broker information, that are the IP address and port number, so only the specified broker will connect to our python code. We also build the list of sensors and set their values to a default value. (Figure 10.)

The dashboard (Figure 2.) displays the minimum and maximum sensor values read from the database, as well as the current sensor value that we receive continuously through the MQTT broker. We implemented also the possibility of adjustment (increase or decrease) of these values.

```
sensors={"sensor-AT":{"sensor_type":"n_value","interval":2},\
         "sensor-AP":{"sensor_type":"n_value","interval":4},\
         "sensor-RH":{"sensor_type":"n_value","interval":2},\
         "sensor-V":{"sensor_type":"n_value","interval":2},\
         "value-EP":{"sensor_type":"n_value","interval":2}
         }
...          ....          ...          ...          ...

if sensor["sensor_type"]=="n_value":
            sensor["sensor_status"]=1.0
            sensor["sensor_status_old"]=0.0
```

Fig. 8. Python code of the sensor data generation

Depending on whether we want to increase or decrease the value of a particular sensor the electrical power output will change accordingly. This simulated data is generated every second and stored in the database.

```
var payl = msg.payload[0];
var id = payl["id"] + 1;
var at = flow.get("next_AT");
var v = flow.get("next_V");
var ap = flow.get("next_AP");
var rh = flow.get("next_RH");
var pe = flow.get("get_ep_flow_v");
var time = flow.get("date_flow");

msg.topic = "INSERT INTO ccpp_new_dataset (id, at, v, ap, rh, re, stamo, time)
            VALUES
            ("+id+", "+at+", "+v+", "+ap+", "+rh+", "+pe+", 2, '"+time+"');"
msg.payload = "";
```

Fig. 9. Insert sensor data to Cassandra database

Figure 9 shows how to construct the INSERT statement for writing the current sensor values to the database cluster. Table I. shows the structure of a database table used for generated data storage.

TABLE I. STRUCTURE OF GENERATED DATA IN CASSANDRA DATABASE

id	AT	AP	RH	V	PE	time

Figure 10 shows the flow for the dashboard of the system.

Current values of AT, AP, RH, V and EP using the gauges, and the charts for each shows the changes in sensor value that have occurred in the past. For AT, AP, RH, and V, it is possible to adjust the value of the sensor as desired, and the value of EP will increase or decrease with a given value as a function of that. Flow for storage of sensor data in the Cassandra database cluster is also presented on Figure 10. At the bottom of the image. The connections with the MQTT broker are marked as subscribe and publish fields in the flow.

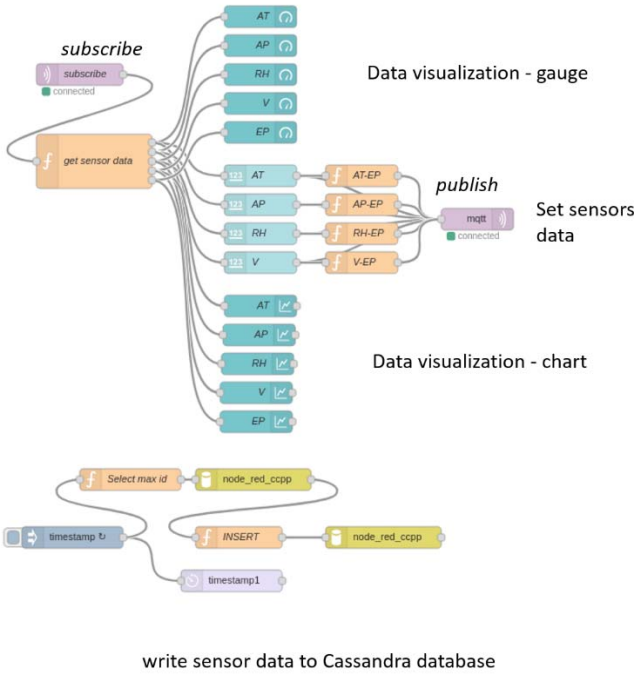

Fig. 10. Flow for subscribe and publish sensor data and for data visualization in the dashboard

VI. CONCLUSION

The presented implementation adequately reflects the fact that the Node-RED development environment makes it easy to communicate with different technologies. It provides cost-effective and time-efficient solution for those in the industry who want to test the effectiveness of a concept, or a new system, so it is perfectly suited to rapid and low-cost prototype development.

The demo application is stable and can be easily used to study the functionality of the modelled CCPP.

With the industry nowadays developing at a high rate and the increasing demand from manufacturers for rapid and cost-effective prototyping, our example proves that we can provide the solution to this, and thus remain competitive. So, the key to successful industry and IoT integration is rapid application prototyping. The security of such systems is crucial therefore we need further investigation on this topic.

We are in the era of the 4th Industrial Revolution, where smart devices are gaining ground in the industry, automation

Authorized licensed use limited to: University of Melbourne. Downloaded on July 26,2020 at 15:10:17 UTC from IEEE Xplore. Restrictions apply.

of production processes, continuous data collection is typical, and data collected is usually taken to the cloud. With our Apache Cassandra database cluster we performed also this step. The cluster can be extended as needed for any application.

In industry, IoT is most important for process optimization, fault detection, and predictive maintenance feasibility. This is important because they can help in costs reduction, and even analyze and re-use the data collected to create a new product or service. With this in mind, we would like to continue our research towards data analysis and information retrieval, while still taking the advantage of the Node-RED environment. So, we want to explore and apply data analysis capabilities provided by machine learning and Cloud service providers for our system, so that we will be able to detect different anomalies and forecast different "events" in the system. In this order we want to implement faulty and outlier sensor data insertion in the sensor data generator python code.

As a result, we want to create a generic system model that can be applied to a variety of different industrial applications.

ACKNOWLEDGMENT

This work was supported by the Collegium Talentum 2019 Programme of Hungary.

REFERENCES

[1] https://archive.ics.uci.edu/ml/datasets/combined+cycle+power+plant

[2] P. Tüfekci, Prediction of full load electrical power output of a base load operated combined cycle power plant using machine learning methods, International Journal of Electrical Power & Energy Systems, Volume 60, September 2014, pp. 126-140

[3] H. Kaya, P. Tüfekci , S. Fikret Gürgen: Local and Global Learning Methods for Predicting Power of a Combined Gas & Steam Turbine, Proceedings of the International Conference on Emerging Trends in Computer and Electronics Engineering ICETCEE 2012, pp. 13-18 (Mar. 2012, Dubai)

[4] L. Zhe, Y. Wang, and K. Wang. "Intelligent predictive maintenance for fault diagnosis and prognosis in machine centers: Industry 4.0 scenario." Advances in Manufacturing 5.4 (2017), pp. 377-387.

[5] W. Chen, H. T. Vo, and P. Ni. "An IoT application for fault diagnosis and prediction"in proceedings of the IEEE International Conference on Data Science and Data Intensive Systems, 2015.

[6] https://www.systec-electronic.com/fileadmin/Redakteur/Unternehmen/Aktuelles/Rueckblick_ew18/20180205_Node_RED_in_industriellen_Anwendungen_Ruprecht_final.pdf

[7] Santhosh, N., Srinivsan, M., & Ragupathy, K. Internet of Things (IoT) in smart manufacturing.

[8] Apache Cassandra: http://cassandra.apache.org/

[9] https://towardsdatascience.com/top-10-machine-learning-algorithms-for-data-science-cdb0400a25f9

[10] Node-RED: https://nodered.org/

[11] MQTT: http://mqtt.org/

[12] K. Ferencz, J. Domokos: “Using Node-RED platform in an industrial environment” XXXV. Jubileumi Kandó Konferencia, Budapest, November 14-15, pp.52-63. 2019

[13] https://new.siemens.com/global/en/products/automation/industry-software/automation-software/tia-portal.html

[14] https://automation.omron.com/en/mx/products/family/SYSSTDIO

[15] Toc, S. I., & Korodi, A. (2018, September). Modbus-OPC UA Wrapper using Node-RED and IoT-2040 with application in the water industry. In 2018 IEEE 16th International Symposium on Intelligent Systems and Informatics (SISY) (pp. 000099-000104). IEEE.

Authorized licensed use limited to: University of Melbourne. Downloaded on July 26,2020 at 15:10:17 UTC from IEEE Xplore. Restrictions apply.