

**Internet of Things**

**Course Project on**

**Predictive Analytics for Motor Care: Unifying Insights with a Single Model**

**Bachelor of Engineering**

**IN**

**COMPUTER SCIENCE AND ENGINEERING**

***Submitted By***

**Team No : A2**

**Div : A**

Omkar gouda S Police Patil 01FE21BCS169 127

Shashank kulkarni 01FE21BCS170 128

Ainapure Darshan 01FE21BCS205 133

Kumar Halingali 01FE21BCS139 168

**Course teacher:** Meenaxi Raikar

**SCHOOL OF COMPUTER SCIENCE & ENGINEERING**

**HUBLI–580 031 (India).**

**Academic year 2023-24**

|  |  |
| --- | --- |
| **Table of Content** |  |
| **Chapters** | **Page**  **No** |
| **1. Introduction**  1.1 Problem Definition  1.2 Objectives | **3** |
| **2. Software and Hardware requirements**  2.1 Schematic diagram using the fritizing tool | **4** |
| **3. Mobile app to remote control the things / Database storage** | **5** |
| **4. Implementation Details / Block diagram** | **6** |
| **5. MQTT / CoAP / Websocket Protocl detail** | **7** |
| **6. Analysis of the sensor data using the tensor flow libraries** | **8** |
| **7. Wireshark IoT Protocol stack capture between the Bluetooth / Wifi module/ Cloud platform and the sensor / network devices.** | **9** |
| **8. Result Analaysis and Screenshots** | **10** |
| **9. Youtube video links submitted** | **11** |
| **10. References** | **12** |
| **11. Conclusion** | **13** |

**1.Introduction:**

Motors serve as the backbone of countless industrial processes, and any unexpected failure can result in significant production losses and increased operational costs.Traditional maintenance approaches, such as preventive or reactive maintenance, often lead to inefficiencies, as they either replace components too early, leading to unnecessary expenses, or too late, resulting in unplanned downtime.This course project delves into the intersection of IoT and machine learning to enhance the reliability and efficiency of motor systems.

* 1. **Problem Definition:**

To develop a robust predictive maintenance system for motors that leverages machine learning and IoT technologies, specifically focusing on a single-model analysis approach.

* 1. **Objectives:**
* To Implement a machine learning model capable of analyzing diverse sensor data from motors for predictive maintenance.
* To apply effective data preprocessing techniques to handle the variability in sensor data.
* To Implement a real-time analysis pipeline that can process and analyze incoming data from IoT sensors attached to motor systems.

**2.Software and Hardware requirements:**

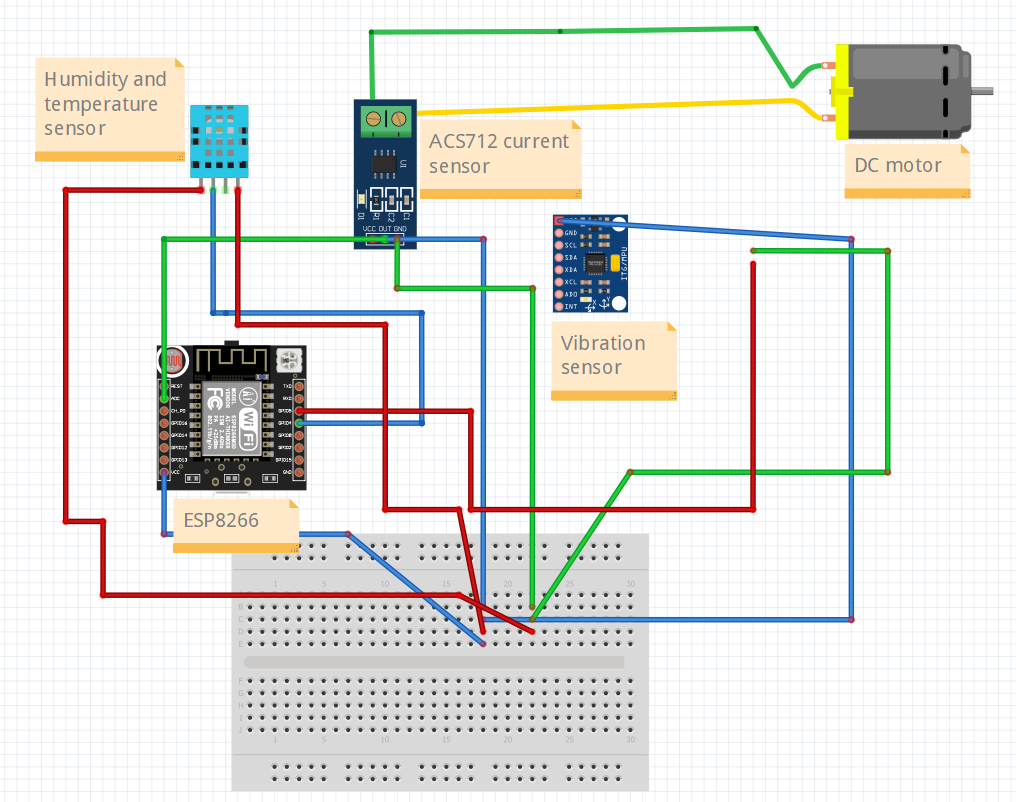
**Software requirements:**

* Arduino IDE – For coding ESP8266 Microcontroller
* Cricuito.io – For Circuit Designing
* ThingSpeak – For Cloud Storage
* VSCode – For ML Code

**Hardware requirements :**

* Single phase motor
* Temperature sensor
* Current sensor
* Voltage sensor
* Humidity sensor
* Vibration sensor
* ESP8266 Wifi-Module

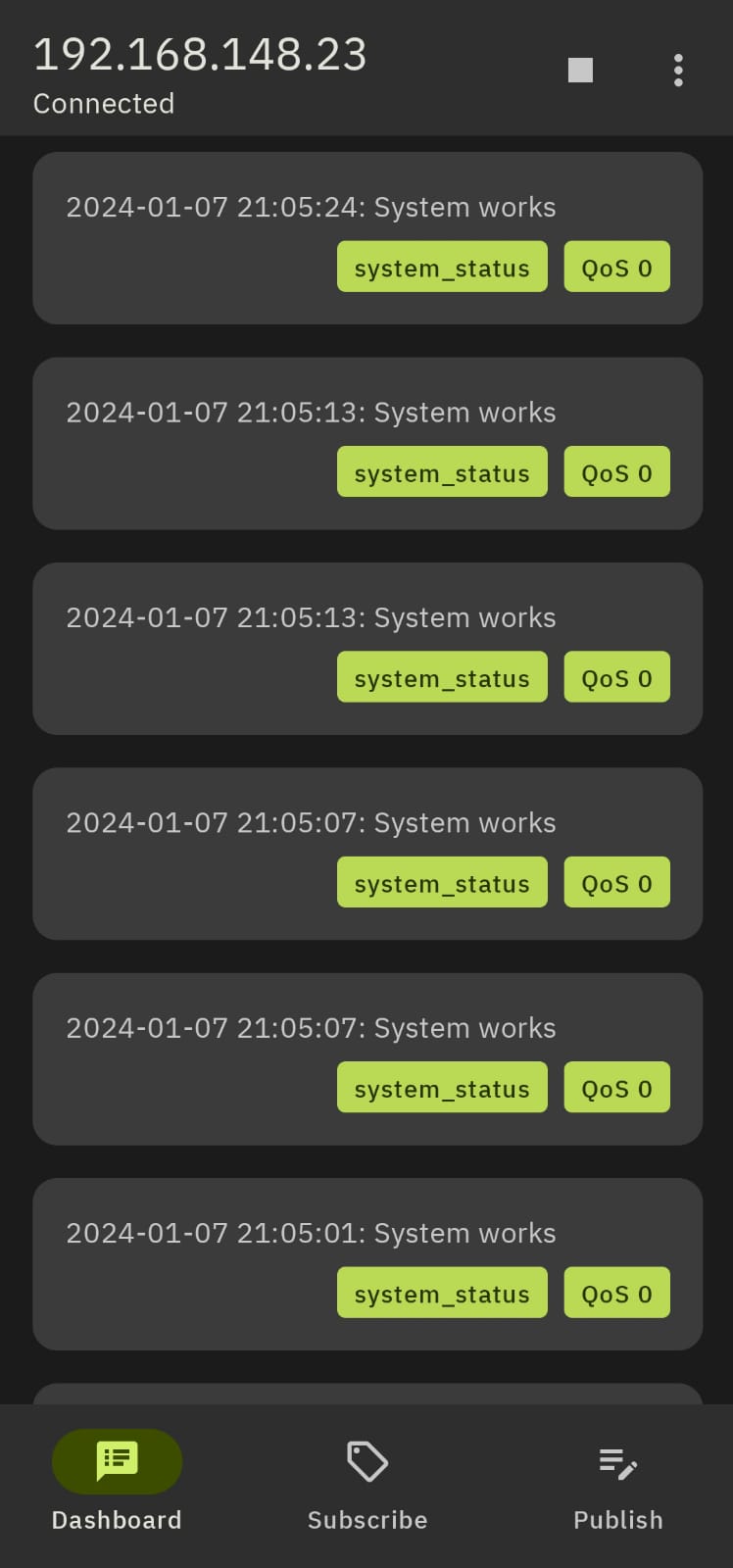
**2.1 Schematic diagram using the fritizing tool:**



**3. Mobile app to remote control the things / Database storage :**

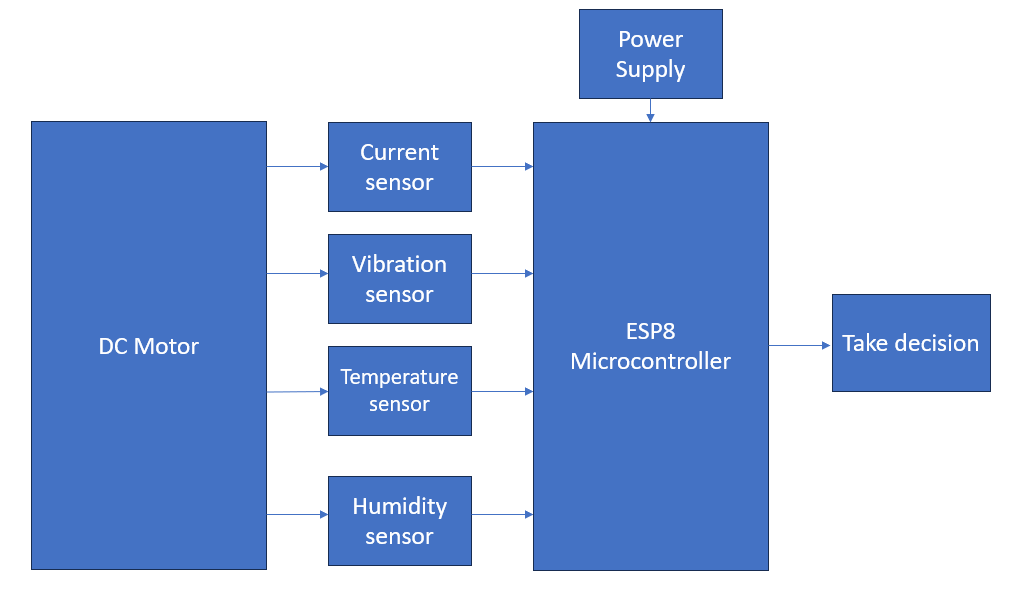
We used the existing My MQTT app to collect the information from all the sensors. We used the app to continuously send the message to user that weather the system is working or not. The message in app is either “system works” or “system fails” with the time stamp. Using this app user can remotely view the status of system and gets to know if the system fails before 5 secs itself as we are using ML model to predict status of system before 5 secs.

**Snapshot of working of Mobile App:**

****

**4.Implementation Details / Block diagram:**

**Block Diagram:**



The Python implementation integrates various libraries and functionalities to enable real-time monitoring and control based on sensor data. The code utilizes pandas for data manipulation, scikit-learn for machine learning tasks, Paho MQTT client for communication. We analyze the data collected by various sensors (current,temperature,vibration,etc). We used real-time-data collected by ourselves to analyze and make predictions on new values.The system continuously receives real-time data through MQTT messages, with the **on\_message** callback processing the data. A sliding window approach is implemented to create new observations for prediction, dynamically updating and evaluating the system's status.The system status is determined by comparing predicted values against historical maximum values for each sensor parameter. If any parameter exceeds predefined thresholds, the system status is set to "System fails." The current date and time are incorporated into the status message, which is then printed and published to an MQTT topic named "system\_status." By, using this system predictive maintenance is made

easier.

**5.MQTT Protocol detail:**

We used MQTT protocol for implementation of our system.

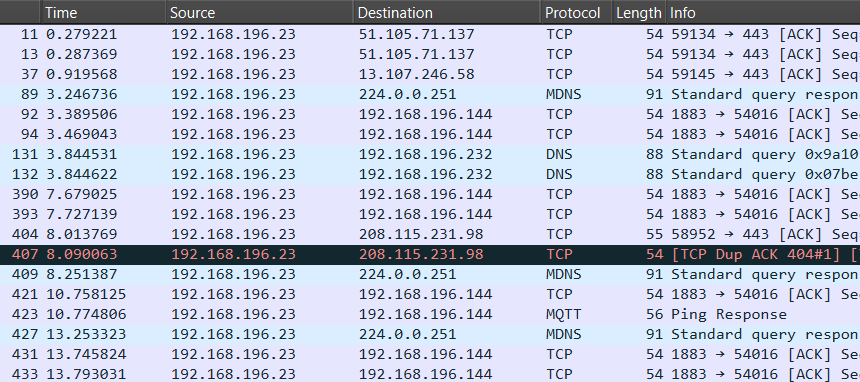
**Details of protocol:**

MQTT (Message Queuing Telemetry Transport) is a lightweight and open communication protocol designed for efficient and reliable messaging between devices, particularly those with limited resources. Operating on the publish/subscribe model, MQTT enables communication between a publisher, which sends messages to a broker, and one or more subscribers that receive messages from the broker. The central element in the MQTT architecture is the broker, which acts as an intermediary responsible for routing messages between publishers and subscribers. Devices that publish information, such as sensors or applications, send messages to specific "topics" on the broker. Subscribers express interest in specific topics, and the broker forwards relevant messages to those subscribers. MQTT's low overhead and simplicity make it well-suited for scenarios where bandwidth and power constraints are critical, such as in Internet of Things (IoT) deployments. Additionally, its ability to handle unreliable network connections, support for Quality of Service levels, and straightforward integration into various platforms contribute to its popularity in applications ranging from home automation to industrial IoT systems.

**6.Analysis of the sensor data using the SKlearn libraries:**

The initial steps include importing necessary libraries such as pandas for data manipulation, scikit-learn for machine learning tasks, and Paho MQTT for communication with the MQTT broker. The script reads sensor data from a CSV file and employs SKlearn's Linear Regression models to predict future values for multiple sensor parameters, such as current, voltage, temperature, humidity, and vibration. The analysis goes beyond mere prediction, with individual linear regression models trained for each pair of input (X) and output (Y) variables. Key metrics like R-squared and Mean Squared Error are utilized to evaluate the performance of these models, providing insights into the accuracy of the predictions. The script dynamically adapts to incoming real-time sensor data received via MQTT messages, showcasing a responsive and adaptable system. Moreover, the system implements a mechanism for assessing the system status based on predicted values. It checks whether certain sensor parameters exceed predefined thresholds, indicating potential issues or system failures. The real-time nature of the analysis is highlighted through the incorporation of date and time stamps in the system status messages, which are printed and published to an MQTT topic named "system\_status." Overall, the SKlearn libraries are effectively leveraged to deliver a robust and responsive system for sensor data analysis and predictive maintenance.

**7.Wireshark IoT Protocol stack capture between the Bluetooth / Wifi module/ Cloud platform and the sensor / network devices:**

****

**Protocols used are:**

* HTTP
* TCP
* DNS
* MQTT
* MDNS

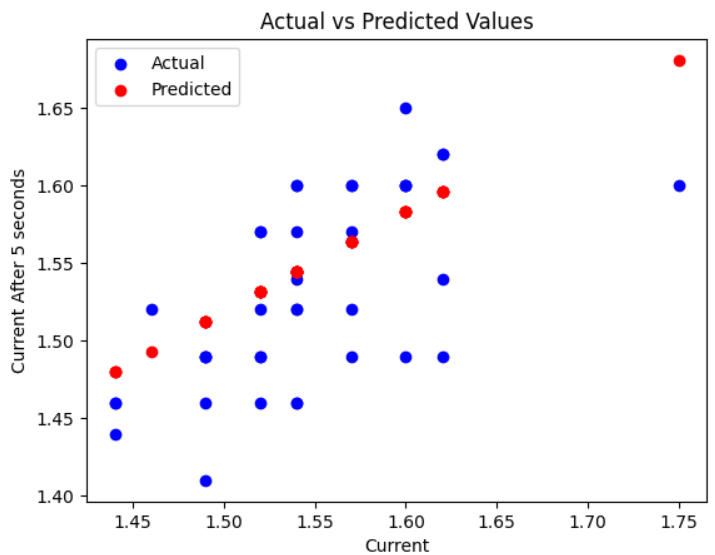
**8.Result Analaysis and Screenshots:**

We used Linear Regression to predict the values of current, voltage and tempearture that will be after 5 secs from at that point of time.The model did perform very well, values of evaluation metrics for current column:

**R-squared:** 0.30732789509735114

**Mean Squared Error:** 0.002580347497709372

**Graph analysis of test\_output vs predicted values of column named current:**

****

**From the above graph we can observe that model predicts the value closer to actual value.**

**9.Youtube video links referred:**

[**https://www.youtube.com/watch?v=5rHWeV0dwxo&t=339s**](https://www.youtube.com/watch?v=5rHWeV0dwxo&t=339s)

[**https://www.youtube.com/watch?v=N9J7Hd5IIfw&t=453s**](https://www.youtube.com/watch?v=N9J7Hd5IIfw&t=453s)

[**https://www.youtube.com/watch?v=RXRACfHTw6M&t=19s**](https://www.youtube.com/watch?v=RXRACfHTw6M&t=19s)

**10.References:**

[1] H. M. Hashemian and W. C. Bean, “State-of-theart predictive maintenance techniques,” IEEE Transactions on Instrumentation and measurement, vol. 60, no. 10, pp. 3480–3492, 2019.

[2] S.-j. Wu, N. Gebraeel, M. A. Lawley, and Y. Yih, “A neural network integrated decision support system for condition-based optimal predictive maintenance policy,” IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans, vol. 37, no. 2, pp. 226–236, 2017.

[3] E. Frontoni, R. Pollini, P. Russo, P. Zingaretti, and G. Cerri, “Hdomo: Smart sensor integration for an active and independent longevity of the elderly,” Sensors, vol. 17, no. 11, p. 2610, 2017.

[4] B. Lu, D. B. Durocher, and P. Stemper, “Predictive maintenance techniques,” IEEE Industry Applications Magazine, vol. 15, no. 6, 2019.

[5] B. Lu, T. G. Habetler, and R. G. Harley, “A survey of efficiencyestimation methods for inservice induction motors,” IEEE Transactions on Industry Applications, vol. 42, no. 4, pp. 924–933, 2020.

[6] W. T. Thomson and M. Fenger, “Current signature analysis to detect induction motor faults,” IEEE Industry Applications Magazine, vol. 7, no. 4, pp. 26–34, 2019.

[7] R. Yam, P. Tse, L. Li, and P. Tu, “Intelligent predictive decision support system for conditionbased maintenance,” The International Journal of Advanced Manufacturing Technology, vol. 17, no. 5, pp. 383–391, 2018.

[8] J.-H. Shin and H.-B. Jun, “On condition based maintenance policy,” Journal of Computational Design and Engineering, vol. 2, no. 2, pp. 119–127, 2021.

[9] M. Paolanti, C. Kaiser, R. Schallner, E. Frontoni, and P. Zingaretti, “Visual and textual sentiment analysis of brandrelated social media pictures using deep convolutional neural networks,” in International Conference on Image Analysis and Processing. Springer, 2020, pp. 402–413.

[10] M. Paolanti, E. Frontoni, A. Mancini, R. Pierdicca, and P. Zingaretti, “Automatic classification for anti mixup events in advanced manufacturing system,” in ASME 2015 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference. American Society of Mechanical Engineers, 2019, pp. V009T07A061– V009T07A061.

**11.Conclusion:**

Predictive maintenance requires continuous monitoring of equipment to detect and diagnose defects. Only when a defect is detected, the maintenance work is planned and executed. Today, predictive maintenance has reached a sophisticated level in industry. Till the early 1980s, justification spreadsheets were used in order to obtain approvals for condition-based maintenance programs. Luckily, this is no longer the case. The advantages of predictive maintenance are accepted in industry today, because the tangible benefits in terms of early warnings about mechanical and structural problems in machinery are clear. The method is now seen as an essential detection and diagnosis too that has a certain impact in reducing maintenance costs, operational vs. repair downtime and inventory hold-up.