Cloud Based Weather Station using IoT Devices

Palak Kapoor
Dept of CSE, IIIT Guwahati, Assam, India
kapoorpalak.1997@gmail.com

Ferdous Ahmed Barbhuiya
Dept of CSE, IIIT Guwahati, Assam, India
ferdous@iiitg.ac.in

***Abstract*—This paper proposes a smart system cloud based weather station. The system uses Raspberry Pi, for collecting and observing weather data. The storing and processing of the obtained weather data is done in cloud to predicting the effect of this weather change. The system is designed to effectively monitor the ambient weather conditions such as temperature, humidity, wind speed, pressure, and rainfall etc. The objective is to design a system which is low cost, requires less maintenance, and involved minimal manual intervention. The system is built using commodity hardware Raspberry Pi, various sensors and uses WiFi as a communication medium which makes the system consume very low power and low cost of building. Smaller Raspberry Pi Zero W boards are used to collect the sensor's data and send it to the base station Raspberry Pi 3 board. The Raspberry Pi 3 then further transmits the data over WiFi to the cloud database and this data is further used to train new Machine Learning model deployed in the cloud for prediction of the effect and to observe and study various weather patterns and trends. The users can access the weather data and insights remotely, and in real time through a web application that is built using the Django Framework, and is deployed in the cloud.**

***Index Terms*—Cloud based Weather Station, IoT based weather information**

I. Introduction

A weather station is a facility, which has the devices and instruments for measuring and observing various atmospheric parameters to provide necessary data for weather forecasts.

Changes in weather conditions have been observed since centuries. To determine the environmental changes, observing the weather parameter fluctuations is essential. Climate impact on human existence has always been enormous, motivating the growth of science fields on climate and weather observation. Simple and incorrect tools were initially used, that were inadequate to store measured parameters and easily read them. Nowadays, there are multiple automated weather forecasting systems and observatories around the globe that continually collect environmental parameters for some or the other application.

The measured parameter information is not helpful unless they are transferred to the users quickly and accurately. A very significant element of a contemporary weather forecasting is the processing and transmission of the measured data. The weather data can be transmitted by a variety of means: GSM/GPRS link, WiFi link, wired link, etc. Digital sensors and micro-controller technology are used by automatic wireless weather stations to take readings. Its quite simple to install automatic wireless weather stations and they save manpower as well.

The proposed system in this paper measures real-time weather parameters using low-cost high precision calibrated digital sensors for accurate measurement of data. This recorded data is then sent to the cloud platform for storage, processing, and prediction. This system is built in integration with Amazon IoT(Internet of Things) platform and uses various services like AWS IoT Core, AWS IoT Analytics, Amazon SNS(Simple Notification Service), Sagemaker etc. This makes this system remotely available, reliable, scalable, and an open source platform that can be integrated with any other technology. It also generates short term local alerts and notifications based on current weather conditions using Amazon SNS web service.

Weather forecasting, regardless of its implementation, must be reliable and precise. It must also provide easy access to all the parameters measured. Sensor quality and measurement precision may differ, so this system is modeled as a distributed environment consisting of multiple Pi Zero W boards and sensors connected to them enhancing the reliability and accuracy of the weather data. A single Pi 3 board acts as a central coordinator to which all Zero W boards send data and from where the data is filtered and sent to the cloud.

A. Literature Survey

The idea of using embedded systems for weather monitoring was proposed way back in the early 2000s. Since then, with the advancement in technology this idea has been carried forward.

Using Arduino and Raspberry Pi board, a weather monitoring system was proposed in [2] with an aim to develop a low-cost weather station that remotely stores climate information on a database server, enabling access from anywhere via any device. The suggested architecture regarded the system's scalability, enabling new sensors to be connected, thus promoting new information surveillance. The architecture was conceived to minimize the effect of these inclusions on the growth of the new versions of the station. The suggested architecture was organized in layers to decrease complexity and expenses and enable further changes Fig. 1. They considered using Arduino along with Raspberry Pi and the sensors to measure temperature, pressure, wind direction, pluviometer for rainfall and anemometer for wind speed. However, due to the fact that two sensors with time-dependent readings, such as the anemometer and the pluviometer, were not able to use two threads efficiently at the same time, some problems arose. Initially two Arduinos were used to fix this restriction, one for each of these sensors. This idea was further enhanced by using just Raspberry Pi 3 embedded with sensors to build a system

978-1-7281-1895-6/19/$31.00

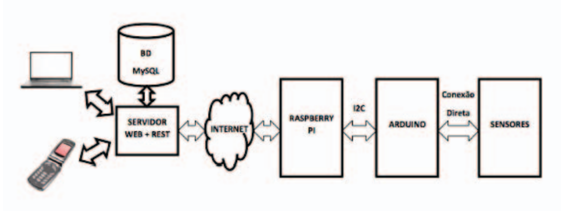


Fig. 1: System Architecture as proposed in [2]

that monitors weather parameters at a location and displays information as required by the user [1] Fig. 2. By linking this weather station to the internet, in anticipating and knowing the weather information at a specific location, the IoT can produced much more comprehensive results. The Thingspeak and mobile app database for storing and sharing weather data are key ways of connecting the weather station to the Internet of things. In comparison with traditional batch processing, IoT applications collect more data. The user can access this data through the internet anywhere in the world at any time.They were inspired by the already implemented automated weather stations in remote areas of Sri Lanka, but so far little attention has been paid to using these types of tools to reduce the cost effectively [8]. They also proposed several modifications enabled by the use of Raspberry Pi 3 in their model, such as the immediate alert message or the sending of e-mail to the mobile phone when the parameter changes are drastic.Also, since the applications are limitless, with the addition of related sensors to the system architecture, other weather parameters can also be easily monitored. With the inclusion of miniature components and increasing the scaling factor, this mini weather station can be made much more compact and reliable. Next, an advanced solution was proposed in [4] to monitor the

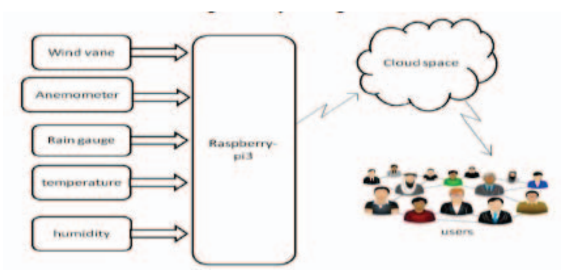


Fig. 2: System Architecture as proposed in [1]

weather at a specific location and make the information visible anywhere in the world using the Internet of Things (IoT). The system dealt with monitoring the environmental conditions like temperature, humidity with respect to its measured time with a micro-controller interfaced with sensors and GSM module to send the information to a remote server wirelessly and use graphical statistics to plot the sensor data A novel wireless prototype of a weather monitoring station that uploads weather data received from multiple sensors from a remote location, to a database in cloud that can be monitored from anywhere was proposed in [3]. Weather data were recorded, monitored and processed in order to forecast various weather events and predict possible disasters. It consist of three main modules - the wireless sensor module, the Arduino microcontroller with Ethernet/Wi-Fi network connectivity and an Android app for users with a feature of instant alert messages via e-mail, text, and tweet notification when data fluctuates drastically.Fig. 3. Development in embedded systems has proved to be a reliable

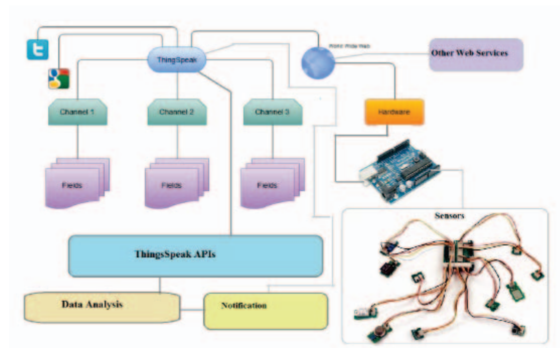


Fig. 3: IoT framework architecture with Thingspeak cloud as proposed in [3]

solution to control and monitor the environment weather monitoring system. [7] proposes a solution for agricultural monitoring in which the aim is to build a system that can be used universally to monitor parameters in a given environment at any scale. With the development of miniaturized sensor devices combined with wireless technologies, parameters such as temperature, humidity, amount of CO_2 in air and many more can be monitored remotely. Raspberry Pi is used as the main board and various sensors to collect all the real time data of the parameters. Users could access the data from anywhere through the Internet. Farmers face large amounts of financial losses due to incorrect weather forecasting, incorrect methods of irrigation, and mistaken amounts of insecticides pesticides and used for the crops. With this motivation they aimed to build a system that proves to be an important part of agricultural development Fig. 4.

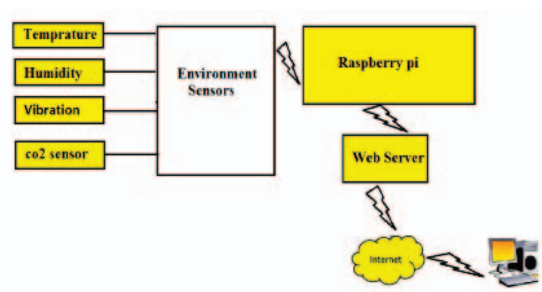


Fig. 4: System Architecture as proposed in [7]

B. Motivation

Today, some institutions and firms now have their own weather stations deployed to collect weather data, such as airports, agencies for the environment and farming.These data are used for their own purposes and shared with third parties in some cases. Many firms, such as Simepar Brazil's INMET (Instituto Nacional de Meteorologia), have weather stations to process data collected and predict climate change.The weather stations used in these cases are professional and their cost is commensurate with the precision and complexity of the sensors used.

For example, the WMR200 is manufactured by the Oregon Scientific company and its cost exceeds 1,000 Dollars, making the purchase impossible for small businesses.The high cost is due to the use of proprietary software and hardware components.The AccuRite 5 in 1 Pro Weather Station is also an outdoor weather station and is bulky and expensive. Hence, it's quite a task to install and maintain it.

Most of the professional weather stations are not customizable, which means they don't allow any sensors to be added or removed, sold as a closed product, and the generated data is usually stored locally in the station.This serves as another limitation for such huge products. Usually the computer interface for viewing or manipulating this data is static, thereby not allowing for customization. Weather stations, as they are now, are usually found in large urban centres, airports, or climate research institutions. The disadvantage is that these stations are many miles away from the place they are monitoring, and the data collected may be inconsistent with what is actually happening there, depending on the distance. Information such as temperature, rainfall index, and wind speed is highly unstable, and distance variation will distinguish the data from the real conditions of the location that we want to monitor. Also, these systems might fail in case of disasters which may result in loss of property.

A vast amount of research has been done in order to make weather stations up to date with state-of-the-art technologies. However, most of this research is largely theoretical and is yet to materialize into a practical product. Our objective is to make a practical working weather station that incorporates the latest advancements in all the relevant technologies.

Many crises can be averted, or their effects mitigated if real-time data to predict their severity is used. Further, such solutions are viable only if they are spatially, temporally and economically efficient. This can be achieved through distributed and cooperative paradigms.

Keeping all such problems in mind, a modular, cost-efficient, compact, smart, higher precision and remote access solution is desired and will be realized in this product.

II. Problem Statement

A. Block Diagram

The plan is to have a distributed sensing from the embedded sensors which records and sends data to Raspberry Pi Zero W boards which further send it to the Raspberry Pi 3 board that acts as a central unit in the model. Keeping in mind all the desirable features, such a model has been designed and multiple Raspberry Pi Zero W boards have been used to make our product modular, compact and less power consuming. Also, this helps in attaining accuracy in the data recorded as multiple sensors value in aggregation gives better insights and further leads to correct prediction of the effect. Further, the collected data is then transmitted to the Amazon IoT platform for further analysis and conclusions. This enables remote access of the data to the users. The representation is shown below: [Fig. 5]

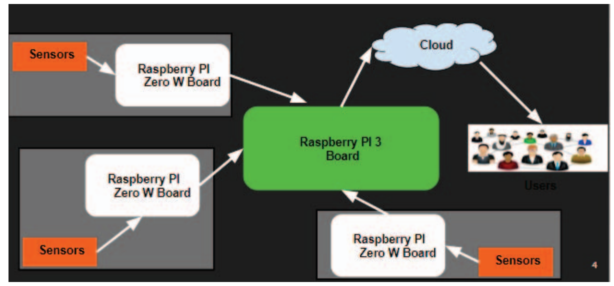


Fig. 5: Model: Block Diagram

B. Overview

The model consists of a Raspberry Pi 3 board, which is the central unit with multiple Raspberry Pi Zero W boards (the number of Zero boards depends on the area that needs to be monitored), communicating with it to send multiple sensors' data. The data is sent to Pi 3 using the Retained MQTT(Message Queuing Telemetry Transport) protocol. Further, data received by Pi 3 is processed, filtered and forwarded to the AWS IoT platform for analysis using a MQTT topic. An algorithm that enables the board to decide when to upload the data to the cloud, since negligible changes in the sensor data need not be uploaded, has been developed. This saves power consumption and also bandwidth, thus making the storage efficient. The Raspberry Pi 3 has also been explicitly programmed to check the proper functioning of the sensors and prevent false alarms. Therefore, for accuracy and preventing false alarms, multiple modules of Pi Zero W with sensors are used (multiple inputs give better results). Moreover, this makes the device modular as well as compact. The data on the cloud platform is stored and then brought for analysis using a trained model put up in the cloud for prediction. The analysis aspects are kept open for the user and can be customized, and suitable models can be used as per the requirement and the predictions that need to be done.

A web-based user interface is also built using Django in integration with Dynamo DB that enables real-time weather monitoring, and availability of data records and insights.

C. System Workflow

Basically, the entire project can be divided into four stages: 1) Collection of Data 2) Storing Data in Database 3) Analysis of the Collected Data 4) Prediction

1) **Collection of Weather Data**
   This is the first stage where all the available sensors are interfaced with the Pi Zero W boards such that the data from these sensors can be recorded. This involves the installation of various in-built packages and libraries

which are easily available. The sensors can also be calibrated depending on the performance and value of the parameters the sensors provide.

The values from these sensors are acquired and sent to Pi 3 for further processing. Processing includes setting up the time interval in which the data must be uploaded to the cloud, any anomaly in the data received, any changes in the weather data, checking the proper working of all the sensors, and preventing false alarms.

Multiple Pi Zero W boards send the recorded data to the base Raspberry Pi 3 device using the MQTT protocol. The MQTT protocol provides a lightweight method to use a publish / subscribe model to perform messaging. This makes it suitable for the Internet of Things messaging with low power sensors or mobile devices such as phones, embedded computers or micro-controllers. Mosquitto, an open source message broker that implements the MQTT protocol is installed on each device in the network which facilitates this data transfer. For better performance, MQTT Retained messages are used. Normally if a publisher publishes a message to a topic, and no one is subscribed to that topic the message is simply discarded by the broker. However, the publisher can tell the broker to keep the last message on that topic by setting the retained message flag. This enables the PI 3 device to get the last active state of the Pi Zero W devices in case of the occurrence of some disconnection or message drop.

2) **Storing Data in Database**

The next phase includes storing of the weather data in the database hosted on the cloud. Dynamo-DB is used to store this data.

Amazon DynamoDB is a fully managed NoSQL database service that provides fast and predictable performance with seamless scalability. DynamoDB lets us offload the administrative burdens of operating and scaling a distributed databases so that the problems of hardware provisioning, setup, configuration, replication, software patching or cluster scaling is addressed. The data is sent from Raspberry Pi 3 board to the AWS IoT platform using the MQTT topic to which the device subscribes and is then passed to Dynamo-DB using a Dynamo DB rule which is created. DynamoDB rules allow taking of information from an incoming MQTT message and writing it to a DynamoDB table. The data can be stored in tables and can be retrieved whenever required.

AWS SDKs(Software Development Kits) have been used to interact with Dynamo DB and access it using the API. For this, an AWS access key ID and secret access key must be obtained which is used to establish a connection with Dynamo DB. Further, this weather data is also used for giving insights and analysis.

3) **Data Analysis**

The next step is to obtain insights from the data collected. The data along with Dynamo DB is also sent to the AWS IoT Analytic platform so that ML tools and models can be used for analysis.

AWS IoT Analytics is a fully-managed service that makes it easy to run and operationalize sophisticated analytics on massive volumes of IoT data without having to worry about the cost and complexity typically required to build an IoT analytics platform. It is the easiest way to run analytics on IoT data and get insights to make better and more accurate decisions for IoT applications and machine learning use cases.

IoT data is highly unstructured which makes it difficult to analyze with traditional analytics and business intelligence tools that are designed to process structured data. IoT data comes from devices that often record processes that are fairly noisy. These device data can often have corrupted messages, significant gaps, and false readings that need to be cleaned-up before analyzing it.

Data set is prepared from this raw data after performing various filtering techniques and then this data set is ready to be used as training data for the ML model for analysis. This data set is also passed on to Amazon Quicksight for further insight and observations.

Amazon Quicksight is also used for getting insights about the data recorded and to see various trends in weather changes and observe patterns on daily, weekly, monthly and yearly basis for gaining knowledge to make various decisions depending on the application.

4) **Prediction of the Effect**

After obtaining the insights from the analysis phase, and setting up a trained model in the cloud, the incoming data is given as an input to the model and prediction of any weather change or emergency is made.

Also, warnings, notifications, e-mails, etc. are automatically sent in case of any significant and/or alarming changes in the weather conditions using Amazon SNS service.

Fig. 6 shows the system architecture and the data flow within the system is shown in Fig. 7.

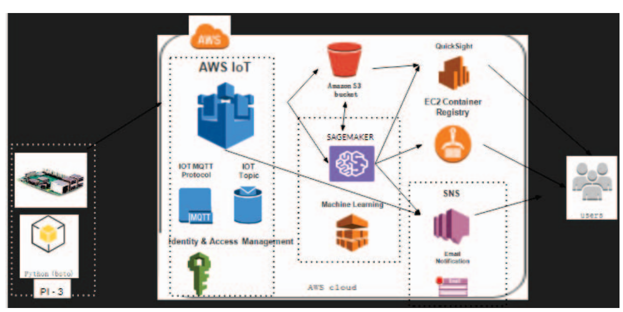


Fig. 6: System Architecture

D. Addressing Various Characteristics of a System

1) Accuracy

An algorithm has been made that parallelly reads input from all the sensors through Pi Zero W boards and maps it to a single real value based on the recorded parameters.

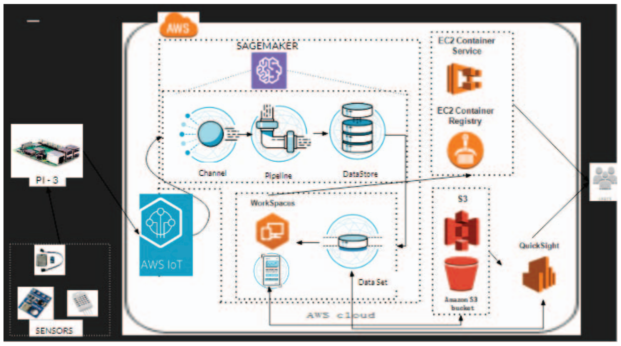


Fig. 7: system data flow

This single real value is a general representation of the recorded parameters. For working on a smaller scale, absolute mean is used to compute the representative. By increasing the number of sensors, the probability of getting an accurate representative increases.
In the case of any faulty/noisy data, proper error messages are used to check on the faulty sensor. Another way to deal with noisy parameter values is to assign various weights to the boards and then data for a particular parameter can be collectively computed.
Later on, a trained machine learning model may also be used to learn the best value for the weather parameters from all the different sensor readings and send it to the Pi 3 for storing it into the database.

2) Availability
The system provides real-time weather monitoring and analysis using the present prevailing weather parameter values of a particular area. The system doesn't generate much traffic while communication too. Since the data storage is cloud-based, availability is not an issue. At most two Pi 3 boards might be involved in a system (depending on the area covered) communicating with the Dynamo DB. This will not generate any overhead and will never cause the system to fail.
3) Scalability
Use of multiple modular systems consisting of multiple Pi Zero W boards integrated with sensors enables scaling of the system. The system can have multiple such modules depending upon the area that needs to be monitored.
Dynamo DB has an auto-scaling feature that considers the data being recorded.
4) Reliability
Using cloud-based infrastructure provides our system and database leading-edge security. Updation of the database occurs only when there are significant changes in the records. This provides the proposed system with high-reliability on minimum update overhead.
5) Portability The proposed system is automated, portable, easy-to-customize and easy-to-integrate with other available platforms. Automation of the system is done as follows: One Pi Zero W boots up, a script automatically senses the weather parameters and then transfers it to the base Pi 3 board for further processing.
6) Modularity
To ensure modularity of the system various modules consisting of Pi Zero W boards and sensors, each of which works independently of one another, communicating and sending data to the base Pi 3 have been used.

III. Implementation

A. Components

1) Raspberry PI 3 contains a 1.2 GHz ARM Cortex-A53 CPU and supports the ARM 64 architecture.
2) Raspberry PI Zero W
The Raspberry Pi Zero W extends the PI Zero family and comes with additional wireless LAN connectivity and Bluetooth.
3) Sensors
   a) DHT22 - Humidity and Temperature
   The DHT22 is a basic, digital low-cost humidity and temperature sensor. It works with a thermistor and a capacitive moisture sensor. The sensor measures humidity readings between 0 to 100 % humidity with an accuracy of between 2 - 5% and temperature readings of -40 °C - 80 °C with an accuracy of +0.5C to -0.5. On the data pin, it then sends out a digital signal.
   b) BMP180 - Atmospheric Pressure, Humidity, and Temperature
   This Bosch precision sensor for measuring barometric pressure and temperature is the best low-cost sensing solution. This board/chip uses I2C(Inter-integrated Circuit) 7-bit address 0x77.
   c) BME280
   Sensor BME280 is an environmental sensor measuring temperature, barometric pressure and humidity. This sensor is ideal for weather / environmental sensing of all kinds and can even be used in I2C and SPI (Serial Peripheral Interface).
   This Bosch precision sensor is the best low-cost sensing solution for measuring barometric pressure with +-1 hPa absolute accuracy, humidity with +-3% accuracy, and temperature with +-1.0°C accuracy. The BME280 is the BMP180/BMP085/BMP183 upgrade with 0.25 m low altitude noise and the same fast conversion time.
   d) Rain Sensor
   The sensor module Raindrops Detection is used to detect rain. It is also used to measure the intensity of rainfall.The module includes a rain board and a separate control board for convenience. It has an LED power indicator and a sensitivity that can be adjusted via a potentiometer.
   This is based on the LM393 op-amp.
4) Jumper Wires
5) Power Source

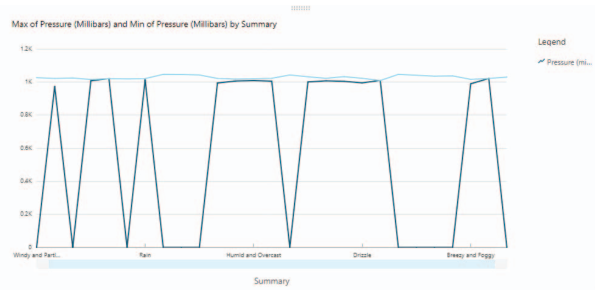


Fig. 9: Maximum and Minimum Pressure by summary

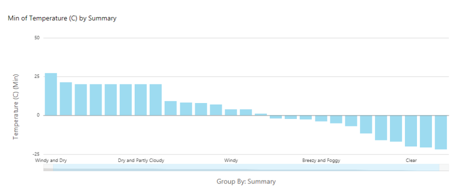
6) Bread Board (optional)

IV. Result

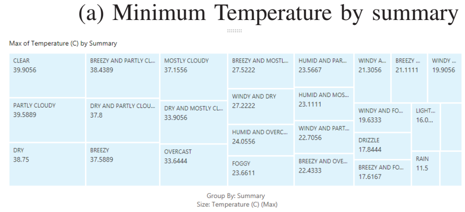
Results include various insights that can be obtained on a daily basis using Amazon Quicksight.[Fig:8] shows us the trend in temperature depending upon the summary. Similarly [Fig:9] shows the trend of pressure values. We can gain any such type of insights by interchanging the parameters and setting up the filter properly using the Quicksight interface.

Also, the prediction model deployed in the cloud gives us the prior information about any calamity or emergency that might occur from the real-time data being sent for testing.

The web application built also provides real-time weather data to the users.



(a) Minimum Temperature by summary



(b) Maximum Temperature by summary

Fig. 8: Temperature Insights from Amazon Quicksight

V. Conclusion

This work has achieved its proposed goal of developing a cost-effective, modular, smart weather station that remotely stores climate data in a database server that can be accessed from anywhere via any device.

The usage of Raspberry Pi Zero W made the product much more compact and thereby easing installation. Using multiple such modules helped in attaining accuracy and also solved the problem of false alarms. Using Raspberry Pi 3 as a central processing element and using all its features, with reduced physical size, low energy costs, and high processing power, this idea has been made successful by allowing the execution of developed applications through commercial language such as Python.

The AWS IoT Device SDK enables our device to easily connect, authenticate, and provide all other benefits of cloud storage such as security, remote access, etc. The platform provides a one-click integration with the device and also connects to the dynamo database for storage of data.

VI. Future Work

The future work includes collaboration with satellite images to make more accurate predictions using the topography as a parameter in our prediction model. Also the model can be extended to use it in monitoring dams/crop yields and other allied areas. Data Analytics and Recommender Systems based on weather data can be used for recommending various day-to-day human activities like travel plan, outdoor activities etc.

Acknowledgment

This paper is an outcome of the project ”Design and Innovation Centre” spoke at IIIT Guwahati, a part of ”National Initiative for Design Innovation” funded by Ministry of Human Resource Development, Govt. of India.

References

[1] K.N.V. Satyanarayana*, S.R.N. Reddy, P.V.Y.N Sai Teja, MD. Basit Habibuddin, “IOT Based Smart Weather Station Using Raspberry-PI3,”,Journal of Chemical and Pharmaceuticals Sciences,JCHPS Special Issue 10: December 2016

[2] R. C. Brito, F. Favarim, G. Calin and E. Todt, “Development of a low cost weather station using free hardware and software,”, 2017 Latin American Robotics Symposium (LARS) and 2017 Brazilian Symposium on Robotics (SBR), Curitiba, 2017, pp. 1-6. doi: 10.1109/SBR-LARS-R.2017.8215292

[3] S. Gangopadhyay and M. K. Mondal, “A wireless framework for environmental monitoring and instant response alert,”, 2016 International Conference on Microelectronics, Computing and Communications (MicroCom), Durgapur, 2016, pp. 1-6. doi: 10.1109/MicroCom.2016.7522535

[4] Chaw Myat Nwe, Zaw Min Min Htun , “A smart weather monitoring system using Internet of things,”,International Journal of Scientific Engineering and Research ,Volume 6 Issue 7, July 2018

[5] S. Halder and G. Sivakumar, “Embedded based remote monitoring station for live streaming of temperature and humidity,”, 2017 International Conference on Electrical, Electronics, Communication, Computer, and Optimization Techniques, Mysuru, 2017, pp. 284-287. doi: 10.1109/ICEECCOT.2017.8284683

[6] K. VivekBabu, K. Anudeep Reddy, CM. Vidhyapathi and B. Karthikeyan , “Weather Forecasting using Raspberry pi with Internet of things (IoT),”, 2006 ARPN Journal of Engineering and Applied Sciences

[7] Jadhav G, Jadhav K and Nadlamani K, “Environment Monitoring System using Raspberry-Pi,”, IRJET, 03 (04), 2016, 1168-1172.

[8] Lanre Joseph O and Umoru Sam A, “Design and Construction of a Low Cost Digital Weather Station,”, AUJT, 16 (2), 2012, 125-132.