XCZTJP1F

PONTIFICIA
UNIVERSIDAD
CATÓLICA
DEL PERÚ

# CONSTANCIA No. 2017-412-B-0000622-01

El Decano de la Facultad de Ciencias e Ingeniería de la Pontificia Universidad Católica del Perú y el Presidente del Comité Organizador dejan constancia que:

**EDWIN SALCEDO**

ha participado como Ponente en el III Congreso Internacional de Ingeniería Informática: Retos y Perspectivas del Mundo Digital, realizado en Lima, del 9 al 11 de agosto del 2017.

Conferencia titulada:

*Sistema automático de monitoreo y predicción de lluvias usando Internet of things y Machine Learning*

Se expide la presente constancia a solicitud del interesado para los fines y usos a que hubiere lugar.

Lima, 25 de agosto del 2017

MIGUEL HERMOGENES MEJIA PUENTE
DECANO

JORGE ALBERTO SOLIS TOVAR
PRESIDENTE DEL COMITÉ
ORGANIZADOR

# Automatic system for rainfall monitoring and prediction with IoT and Machine Learning

Edwin Salcedo Aliaga
Bolivian Catholic University "San Pablo"
La Paz, Bolivia
er.salcedo@acad.ucb.edu.bo

***Abstract* — Environment information such as precipitation, temperature, solar radiation, soil and ambient humidity is fundamental to identify the climate change patterns across a country. For example, the monthly and daily records of these meteorological variables are important to locate regions that present water deficit (droughts) or excess (floods). By, monitoring, analysing, and evaluating this data, it is not only possible to take effective actions to prevent any hard climate variation but also to improve the planning of superficial and underground hydrological resources in a region. The present paper describes the development process of a low-cost IoT system for automatic recording, monitoring, prediction of rainfall in urban and rural regions in Bolivia, using Arduino, sensors, GSM/GPRS communication, and Machine Learning. Additionally, the project considers the integration and processing of public data sources such as the System of Meteorological Information SISMET, which increased the accuracy of the project's visualizations and predictions across Bolivia.**

***Key words* — *Pluviometry, GSM/GPRS, Machine Learning, Internet of Things.***

## I. INTRODUCTION

During the last years, Bolivia has maintained a flourishing agricultural and livestock sector in its western and eastern region. Although agriculture continues to play an important role in the Bolivian economy, the agricultural sector and the government sector have failed to adopt advanced technologies to monitor weather conditions and to prevent multiple climate catastrophes in the last years, and thus have lagged behind many other developing countries.

Climate information such as precipitation, radiation, and humidity are critical to the success of agricultural activities in rural areas. Unfortunately, current methods to keep track of these data primarily use observation and manual recording of data generated by installed weather stations, which is highly error prone. In addition, another great problem is the high cost of rain gauge equipment and weather stations, which greatly limits their acquisition and implementation in the Bolivian territory. Nowadays, there are less than 150 rain gauges, weather and hydrological stations installed throughout the country, of which 60% are in the main cities: La Paz, Cochabamba, and Santa Cruz, leaving large sections of the country without monitoring [1].

The automation of these instruments can increase not only the reliability, but also improve the timely availability of the data, and therefore, contribute greatly to the improvement of agriculture and livestock, as well as to early prevention of droughts and floods.

This project consists of the development of a system of low-cost weather stations, each one composed of a rain gauge mechanism, sensors of temperature, soil humidity, humidity of the environment, and solar radiation, and a wireless communication component. On top of that, the system, illustrated in *Figure 1,* is oriented to capture data from all weather stations and their sensors in a central web system. Each device has a GSM/GPRS module that can send information about precipitation and sensors via SMS (Short Message Service) messages to the web system on a scheduled basis (every 24 hours). The web application can store information from a large number of remote weather stations in real time.

The data received in the SMS is extracted, sorted, and registered in the central database. This information can be accessed through a web interface, and can be viewed based on location and/ or time. In turn, this information can be used to predict droughts or floods in the different regions of the Bolivian territory. Furthermore, the project inherits the great coverage and reliability of the GSM/GPRS network infrastructure.

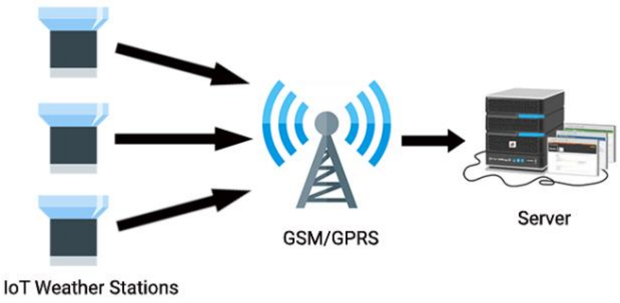


**Figure 1:** System description

## II. SYSTEM DESIGN AND IMPLEMENTATION

The system design and its implementation comprises the following parts, illustrated in *Figure 2*.

- Weather station
- Communication system
- Web system

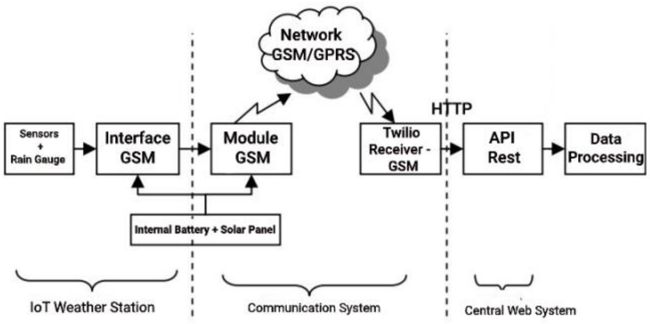


**Figure 2:** System implementation

### A. *Weather station*

The IoT weather stations are equipped with a rain gauge mechanism and other sensors (temperature, soil humidity, ambient humidity, and solar radiation) interconnected with a GSM/GPRS module as well as to a power supply mechanism composed of a 1.5 mA battery. 7 volts and a solar panel. The rain gauge is based on a precision 0.2mm rocker mechanism. this mechanism is detailed in *Figure 3*.

In operation, rainwater falls from an upper funnel onto the tilting buckets. The water fills the bucket in front of the funnel, it can hold up to 0.2mm of rainwater and will tilt when it reaches its limit, thus dropping its contents. Immediately, the other bucket is placed in front of the funnel opening, to repeat the water collection process again.

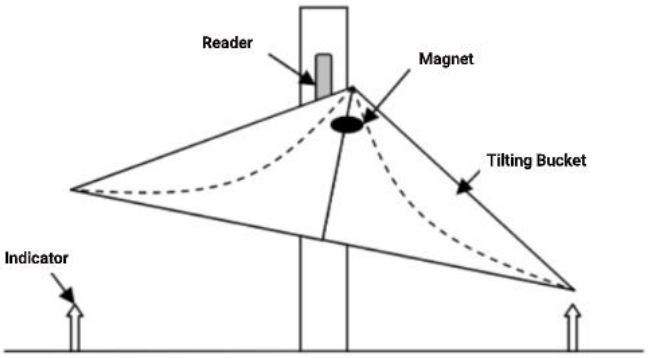


**Figure 3:** Rain gauge mechanism

**Figure 4:** Design of the Weather Station in Blender

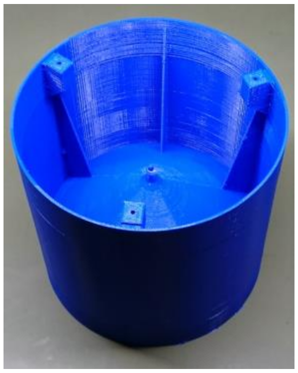


**Figure 5:** 3D printing of the weather station

Each time the cuvette is tilted, a magnetic pulse will be generated at the reader level. These pulses are counted in the Arduino Mega microcontroller to measure the precipitation then.

It should be noted that the components of both the rain gauge and the weather station were designed and printed in 3D. This process is shown in *Figures 4 and 5.*

While one part of the Arduino Mega microcontroller circuit is dedicated to counting the rain gauge pulses, another part is dedicated to recording the results of the sensors. The sensors used are described in more detail below:

- DHT11: Temperature and humidity sensor
- HL-69: Soil moisture sensor
- ML8511: Solar radiation sensor

The micro controller communicates with the GSM module using a USART (Uniform Synchronous Asynchronous Receiver Transmitter) port. The USART receiver pin is enabled to generate interrupts. The transmitted and received data are stored in the buffer and this is sent to the GSM module, which in turn proceeds to send SMS messages with the information formatted as a reading every 15 minutes.

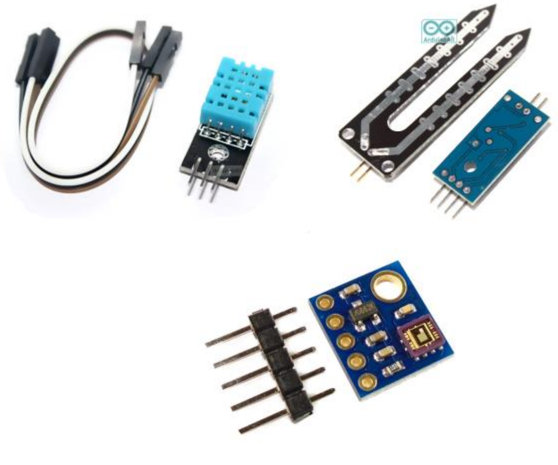


**Figure 6:** Used sensors

The remote weather station is powered by a 7V battery while a solar panel is connected to it. A lead-acid battery is used, as it is rechargeable. This energy is sent to a rectifier bridge circuit and then regulated to a voltage of 5V so that it can power the micro controller and the GSM module. In this way, the circuit has an uninterrupted supply either from the battery or the solar panel.

### B. *Communication System*

The precipitation and sensor information is embedded in an SMS (Short Message Service) message, and is sent to the central web system through a GSM/GPRS module. The communication system consists of three sections represented in *Figure 7*.

- Module GSM/GPRS
- Internet WAN
- SMS Twilio receiver

**Figure 7:** Communication system

The remote station communicates with the central station by SMS. For the transmission of these messages, an existing wireless GSM/GPRS network is used, which is already available in the entire country. These messages are initially sent to the Twilio SMS receiver, it has the ability to receive SMS and consequently send HTTP requests (GET, POST, PUT, DELETE) to the Central Web System. Then, for each scheduled sending of a reading, the Twilio receiver sends a POST request to a URL path, for example: http://domain/createnewreading.

There is also the option of requesting reads from the same central web server. The RF module in the remote station will generate an indication when a new message has arrived. This message will be received from the USART as a switch. The microcontroller will read the switch, recognize it as a new message and prepare the transmission of precipitation and sensor information.

The microcontroller has to convert the information from the rain gauge to text mode, this information is stored in binary format. So for every bit "1" or "0" the microprocessor will convert this to text format. Then this message is encapsulated in the command itself to send an SMS to the central web system.

### C. *Central Web System*

In the central server, a web application developed with the MEAN.JS stack (MongoDB, Express, Angular, Node.JS), is implemented for the management of users, weather stations, and monitoring networks. The structure of the database follows the following entity-relationship scheme, however, the implementation was carried out in a documentary format for the MongoDB database.

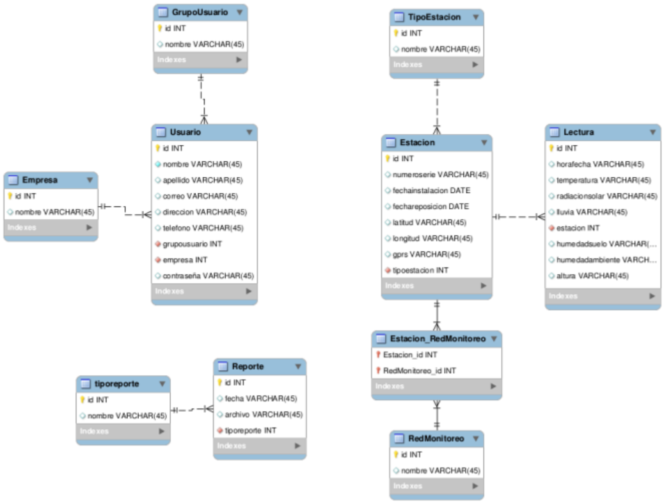


**Figure 8:** Database diagram

The *Figure 9* and *Figure 10* are screenshots of the main interfaces of the system.

### D. *Data management and rainfall prediction with Machine Learning*

**Data extraction** The weather stations record the data from the sensors locally and have several configuration options for automatic sending to the Central Web System. A MongoDB database stores this data for further processing and cleaning with libraries of the Python programming language. A main characteristic of this research project is the use of a second source of information from the National Service of Meteorology and Hydrology of Bolivia (SENAMHI), which provides data on precipitation and temperature in 18 stations installed in the 9 departments of the country. . However, the use of this data is not efficient because its use requires an email request or use of an application in Flash format. This was the main reason for the development of a data extraction component, also known as Scraper, to download the data using the Selenium library, Pandas, and BeautifulSoup. Thus, a .csv file is obtained with all the information published in SENAMHI in HTML format.

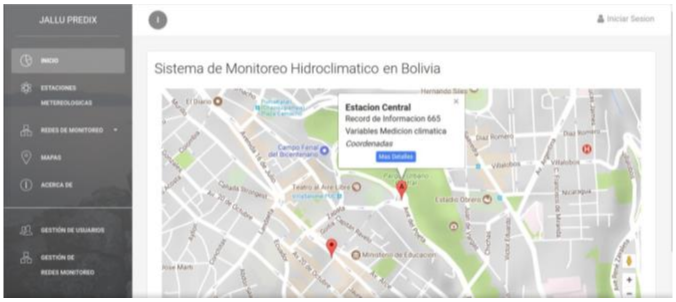
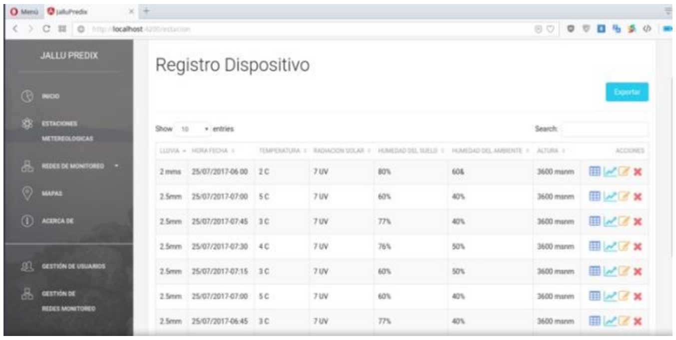


**Figure 9:** View of weather stations' location



Registro Dispositivo

| LLUVIA | HORA FECHA | TEMPERATURA | RADIACION SOLAR | HUMEDAD DEL SUELO | HUMEDAD DEL AMBIENTE | ALTURA | ACCIONES |
|---|---|---|---|---|---|---|---|
| 2 mms | 25/07/2017-06:00 | 2 C | 7 UV | 80% | 60& | 3600 msnm | |
| 2.5mm | 25/07/2017-07:00 | 5 C | 7 UV | 60% | 40% | 3600 msnm | |
| 2.5mm | 25/07/2017-07:45 | 3 C | 7 UV | 77% | 40% | 3600 msnm | |
| 2.5mm | 25/07/2017-07:30 | 4 C | 7 UV | 76% | 50% | 3600 msnm | |
| 2.5mm | 25/07/2017-07:15 | 3 C | 7 UV | 60% | 50% | 3600 msnm | |
| 2.5mm | 25/07/2017-07:00 | 5 C | 7 UV | 60% | 40% | 3600 msnm | |
| 2.5mm | 25/07/2017-06:45 | 3 C | 7 UV | 77% | 40% | 3600 msnm | |

**Figure 10:** Creation, update and removal of IoT

**Data cleaning** The data available on the website (SENAMHI) are daily data from the period 2003 to 2016, therefore they provide a wide range of historical data for each Bolivian government station. The next step was to clean and process them for their correct adjustment to the Machine Learning model. On the other hand, the data collected by the IoT stations, installed for this project, were also processed with the same pre-processing in such a way that both data sets had the same characteristics. For the project, it was possible to collect daily precipitation data with the IoT meteorological stations for the months of February to April 2017.

**Figure 11:** Visualisation of historical data + predictions for the following days

**<u>Choice of algorithm</u>** For this project, different Machine Learning models already used in a similar context were evaluated, but the Auto Regressive Integrated Moving Average (ARIMA) model was the one that best fit the data [6]. This model has the ability to work with historical data to predict future values. Once the most accurate model had been chosen, the SENAMHI data was separated for a training stage and an evaluation stage for each meteorological station. Subsequently, these predictions were recorded in the MongoDB database to be displayed in the web interface, to be able to access the data and the prediction in a friendlier environment. The procedure for the data from the IoT meteorological stations followed the same method, however, they required extrapolating data from the neighboring stations, mainly the meteorological stations located in the city of La Paz.

<u>Prediction results.</u> The results are sown in *Figure 12* are some of the results when calculating the precision of the model with respect to the data, for the validation stage.

| | |
|---|---|
| Overall accuracy | 0.787319 |
| Average accuracy | 0.858213 |
| Micro-averaged precision | 0.787319 |
| Macro-averaged precision | 0.797411 |
| Micro-averaged recall | 0.787319 |
| Macro-averaged recall | 0.365375 |

***Figura 12*:** Predictive model evaluation metrics

ARIMA fits the data and there is an acceptable level of precision, even more work is required in the data processing and the ML model.

## III. CONCLUSIONS

An automated rainfall monitoring and prediction system is a more reliable and accurate method compared to existing manual methods. This is particularly useful in remote areas and towns that lack easy access to roads. In areas where electrical power is not available, activation of the solar panel is essential, thus allowing remote weather stations to operate for long periods of time.

The main drawback of the current proposal is its high dependence on the GSM / GPRS wireless infrastructure. However, this has a significant advantage in that the precipitation and sensor data inherit the reliability and coverage of this network, which greatly simplifies the design of the solution. Because this network is constantly expanding, it is to be expected that it will reach more ground throughout the country.

The cost of the proposed IoT metrology station can range from $ 280 to $ 320, but its design is for research purposes. The next stage will be to use industrial electronic components that allow to build more resistant and better quality devices. On the other hand, the project rescues the manufacture of its external components through 3D printing.

The remote meteorological station developed can easily be extended to measure other types of variables such as wind, evaporation, atmospheric pressure, incorporating the appropriate sensors. This information would be easily integrated into the SMS messages, transferred by the GSM/GPRS network, as well as in the central web system.

Data processing still requires extensive work to consider all variables (temperature, soil and air humidity, radiation, precipitation) when making predictions. Besides that, the data from different stations should also be related based on how close they are. This would allow to create a model to generate forecasts in regions that do not have a weather station installed.

Finally, the information collected can be used for other benefits, not only the agricultural sector, but also society in general, applications such as the prediction and precaution of natural disasters.

## IV. REFERENCES

[1] Mendoza, O. (2013), Bolivia has 119 real time weather stations for the ealy alert and prevention of disasters, PIEB. https://www.pieb.com.bo/sipieb_notas.php?idn=8241

[2] Guide of metereological instruments and observation methods (WMO-No.8, 7th editon), WMO, Genova, 2008.

[3] Hofer, M. (2003). Practical Manual for Weather stations. 4° Edición. Panamericana, Editorial Misa.

[4] D. Estrin, R. Govindan, J. Heidemann, and S. Kumar, BNext century challenges: Scalable coordination in sensor networks,[ in Proc. 5th Annu. ACM/IEEE Int. Conf. Mobile Comput. Netw., 1999, pp. 263–270.

[5] A. Wood, G. Virone, T. Doan, Q. Cao, L. Selavo, Y. Wu, L. Fang, Z. He, S. Lin, J. Stankovic, “ALARM-NET: Wireless Sensor Networks for Assisted-Living and Residential Monitoring”, Technical Report CS- 2006-13, Wireless Sensor Network Research Group, Department of Computer Science, University of Virginia.

[6] V. M. Zavala, E. M. Constantinescu, T. Krause, and M. Anitescu, “On-Line Economic Optimization of Energy Systems Using Weather Forecast Information,” J. Process Control, vol. 19, pp. 1725–1736, January 2009

[7] D. Eni, F. K. Adeyeye, “Seasonal ARIMA Modeling and Forecasting of Rainfall in Warri Town, Nigeria,” Journal of Geoscience and Environment Protection, vol. 3, pp. 91-98, December 2014