



An IoT Based Weather Station Monitoring System with a Telegram bot

Un sistema de monitoreo de una estación meteorológica basado en IoT con un bot de Telegram

Contreras-Salas Lintol¹, Ibarra-Cabrera Manuel J.²

0000-0001-9040-7602¹, 0000-0001-6711-4916²

¹ Universidad Nacional Micaela Bastidas de Apurímac, Perú – lcontreras@unamba.edu.pe

² Universidad Nacional Micaela Bastidas de Apurímac, Perú – mibarra@unamba.edu.pe

(Recepción: 05/03/2023 - Aceptación 21/06/2023)

Abstract. This research work presents the prototype of a low-cost weather station, with a system of queries made from the Telegram bot. The system includes planning, design, programming and testing for the weather station using Raspberry Pi and Arduino Uno boards and sensors for data reading: temperature, humidity, UV and rain gauge. The user can monitor this data, making queries through the social network Telegram or the website. Field tests were conducted on the rooftop of a house during the months of October, November and December 2022 and the results show that the system works properly.

Keywords: Automation, IoT, Weather Station, Raspberry Pi, Arduino

Resumen. Este trabajo de investigación presenta el prototipo de una estación meteorológica de bajo costo, con un sistema de consultas de realizadas desde el bot Telegram. El sistema incluye la planificación, el diseño, la programación y las pruebas para la estación meteorológica utilizando placas Raspberry Pi y Arduino Uno y sensores para la lectura de datos: temperatura, humedad, UV y caudal de agua. El usuario puede monitorear estos datos, realizando consultas a través de la red social Telegram o de la página web. Las pruebas de campo se realizaron en la azotea de una casa durante los meses de octubre, noviembre y diciembre del 2022 y los resultados muestran que el sistema funciona adecuadamente.

Palabras Clave: Automatización, IoT, Estación Meteorológica, Raspberry Pi, Arduino

1 Introducción

Las plataformas de mensajería, en particular aquellas con un enfoque móvil, se han vuelto más pervasivas en la sociedad. Estas plataformas de mensajería móvil pueden tener bases de usuarios relativamente grandes, y además de facilitar la comunicación, a menudo se utilizan para organizar movimientos sociales y como un lugar de encuentro para personas[1]. La red social Telegram, fue creada por un empresario ruso exiliado, Pavel Durov[2], se presenta como un garante de la privacidad en la mensajería que no pertenece a la corriente dominante ni a occidente. La aplicación Telegram tiene la ventaja de ser ligera, gratuita y multiplataforma. Además, la API de Telegram es completa e innovadora, y puede responder a los mensajes de forma automática.

Telegram tiene aplicaciones que se pueden hacer en diversas áreas, por ejemplo para dar soporte al intercambio de información en colegios [3] o universidades utilizado el método de comunicación Webhooks capaz de proporcionar latencia cero y manejar múltiples peticiones concurrentemente durante la comunicación dentro del bot de Telegram [4]; otro

campo de aplicación es en Internet de las cosas (IoT), con la integración entre smartphones y la plataforma Arduino utilizando Telegram Bots, permite a las personas crear prototipos de hardware y comunicarse con ellos utilizando la misma herramienta aplicada en la comunicación con otras personas [5]; también existen soluciones en el campo de la medicina utilizando Telegram para el caso de la pandemia del Covid-19 [6], [7].

El “bot” API de Telegram es una interfaz basada en el protocolo de comunicación HTTP (The Hypertext Transfer Protocol) creada para desarrolladores interesados en crear bots. Cada bot recibe un token de autenticación único al momento de crearse. El token tiene un aspecto similar a 654321:CBA-FED4321rsJlkl-xyz75X3kjkj32we33. Admite métodos HTTP GET y POST y permite pasar parámetros en las solicitudes de Bot API de 4 formas: a) URL query string, b) application/x-www-form-urlencoded, c) application/json (excepto para subir archivos) y d) multipart/form-data (se usa para subir archivos). El bot API de Telegram, es una interfaz basada en el protocolo seguro creada para desarrolladores interesados en construir bots para Telegram. La API ayuda a interactuar con los bots y está diseñado para ocultar toda la sección del protocolo de cifrado MTProto [8].

A nivel mundial, el monitoreo es indispensable para la gestión de riesgos, las decisiones correctas y la promoción de la salud. Se pueden implementar a través de las modernas tecnologías electrónicas y de monitoreo. Arduino es una placa de código abierto que combina un microcontrolador (tarjeta de circuito programable) con software o un entorno de desarrollo integrado (IDE) que opera en un ordenador personal y carga el código a la placa.[9]

La estación meteorológica es un dispositivo capaz de medir los parámetros meteorológicos basados en las condiciones atmosféricas, incluyendo temperatura, humedad relativa, dirección del viento, velocidad del viento, presión barométrica, precipitación (medidor de lluvia) y radiación solar y ultravioleta. La humedad y la temperatura representan la mayor parte de la composición del clima, y estos componentes pueden variar, especialmente en las regiones afectadas por la temperatura de la radiación solar y en los lugares a lo largo de la línea tropical [10].

Las estaciones meteorológicas profesionales incluyen el Davis Instruments 6162c Vantage Pro con un costo de 5700 soles en la tienda virtual de Mercado Libre en Perú (ver Fig. 1) y el Ambient Weather Ws-2000 Smart Weather Station con un costo de 2389 soles en la tienda virtual de Mercado Libre en Perú (Ver en Fig. 2). Estas estaciones meteorológicas se ven afectadas por algunos problemas, así, por ejemplo: utilizan un registrador de datos local (data logger) para guardar los datos leídos por los sensores y luego es necesario volver al lugar para descargar dichos datos cada 15 o 30 días; y otro aspecto a considerar es que el precio del producto supera los 600 dólares americanos.



Fig. 1. Vantage Pro2 Modelo 6152c [11]



Fig. 2. Ambient Weather Ws-2000 inteligente [12]

Este proyecto tuvo como objetivo crear un prototipo funcional de una estación meteorológica de bajo costo que permita monitorear las condiciones climáticas mediante el uso de la red social Telegram.

2 Trabajos Relacionados

En el trabajo de investigación realizado por Faid et al. [13] mencionan que, en comparación con la energía, el Internet de las Cosas (IoT) es una necesidad del siglo XXI. Como resultado de su uso extendido en campos importantes como agronegocios, ha cambiado radicalmente la vida de las personas en la última década. Mediante la mejora de la calidad de las cosechas, el fortalecimiento de la agricultura sostenible y, en última instancia, la generación de un desarrollo

constante, la explotación inteligente es una aplicación en el mundo real donde las tecnologías de Internet de las Cosas (IoT) como las estaciones meteorológicas pueden tener un efecto directo en los seres humanos. Si bien la mayoría de las soluciones meteorológicas no están adaptadas a las necesidades específicas de los pequeños agricultores en los países en desarrollo, tampoco están dentro de su rango de precios. Además, a menudo es difícil crear e implementar sistemas baratos pero robustos debido a los desafíos exteriores. La robustez del sistema, que se basa en aspectos como su eficiencia energética, portabilidad, interoperabilidad y facilidad de uso. El trabajo presenta una plataforma flexible de bajo costo basada en IoT impulsada por la Inteligencia Artificial (IA) para la vigilancia cognitiva en la agricultura inteligente. Con el fin de proporcionar a los usuarios datos ambientales en tiempo real y predicciones basadas en la IA, el sistema híbrido Multi-Agente y el sistema implementado, investigan constantemente varios parámetros agrícolas, como temperatura, humedad y presión. Varios nodos diversos estacionados en la base y en la región de detección abierta garantizan la fiabilidad de los datos recopilados. Después de la recogida, los datos se envían a un servidor local para el procesamiento preliminar y un servidor en la nube para el almacenamiento y redundancia. Las redes heterogéneas como MQTT, NRF24L01 y WiFi se utilizan para el marco de comunicación subyacente del sistema. También implementaron una interfaz gráfica de usuario (GUI) basada en la web que acoge a una amplia gama de usuarios. La plataforma en su conjunto está diseñada utilizando una metodología ágil, lo que la hace fácil de implementar, sencilla de actualizar y conveniente de administrar.

Asimismo, Mathew y Dida [14] realizaron una investigación en la cual analizan que, la industria agrícola se ve profundamente afectada por las fluctuaciones meteorológicas. En Tanzania, el estudio realizado en 2015 constató que la mayoría de las estaciones meteorológicas automáticas existentes no funcionan, y que las estaciones meteorológicas manuales encuestadas registran diariamente los datos meteorológicos en las fichas, pero estas se envían en un intervalo de tiempo de un mes a los órganos respectivos, lo que dificulta el concepto de alerta temprana. Por lo tanto, será posible que los sistemas de vigilancia y previsión de cosechas hagan un seguimiento del crecimiento de los cultivos y emitan alertas tempranas a los agricultores y al gobierno si tienen acceso a datos meteorológicos casi en tiempo real. Con la expansión del IoT, la creación de una red de sensores unificada para recopilar datos meteorológicos casi en tiempo real es ahora una posibilidad realista. La red integrada de sensores utilizada en esta investigación consta de sensores, una red de comunicaciones y una herramienta web que mejora la visualización de datos tanto en formato gráfico como tabular, y registra datos meteorológicos distantes como precipitaciones, humedad y temperatura. La tecnología LoRa se utilizó en el sistema de comunicación porque es superior a otras tecnologías de red de área amplia de baja potencia (LPWAN) como NB-IoT y Sigfox cuando se trata de este caso de uso particular. El sistema desarrollado utiliza el sistema operativo LoRa Gateway que proporciona capacidades para construir una red privada, y hacerla más rentable reduciendo el costo de operación para la suscripción de cuentas, en plataformas online como The Things Network (TTN), ThingPark, ThingSpeak, y Lorient para evitar las limitaciones de cuentas gratuitas. Los sensores Deep L pueden hablar con la pasarela desde hasta 25 kilómetros de distancia, por lo que el sistema desarrollado puede funcionar incluso en lugares con un servicio de Internet irregular.

3 Metodología

Es una investigación de tipo cuantitativo y el alcance es descriptivo. Primero se realizó un análisis de los atributos que se quiere medir y de acuerdo a eso encontrar los sensores adecuados, posteriormente se realizó pruebas de la medida y precisión de cada uno de estos sensores contrastándolo con equipos industriales; y finalmente se integró cada uno de los sensores a los microcontroladores; y se hizo el diseño y la implementación de una carcasa protectora usando una impresora 3D. Las pruebas de funcionalidad se hicieron inicialmente en la azotea de una casa y posteriormente en el Santuario Nacional del Ampay en Abancay.

Para el proceso de implementación del software se utilizó las fases [15] de la metodología ágil “programación extrema”. En la fase de *Planificación* se elaboraron 15 Historias de Usuario (HU). La Tabla 1 muestra un ejemplo de HU.

Tabla 1. Historia de Usuario

<i>Historia de Usuario</i>	
Numero: 1	Usuario: Cliente
Nombre de la Historia: Registro de temperatura y humedad leída por el sensor	
Prioridad en Negocio: Alta	Riesgo en desarrollo: Alta
Puntos estimados: 21	Iteración Asignada: 1
Descripción:	

El sistema registra la temperatura y humedad leída por el sensor, consignando la siguiente información:

- Sensor
- Fecha de Lectura
- Hora de Lectura
- Temperatura y humedad leída

Observaciones:

El registro de la temperatura y humedad se hace de manera automática.

En la fase de *Diseño* se realizó un boceto simple de las interfaces de usuario que debe tener la aplicación web y la aplicación móvil con Telegram. La Fig. 3 muestra el diseño de la interfaz en Telegram.



Fig. 3. Diseño de la interfaz móvil en Telegram

En la fase de *Codificación* se realizó la programación en pares cumpliendo algunos estándares de programación. En la Fig. 4 se muestra la parte más importante del código fuente de la programación del bot en Telegram con los comandos más utilizados.

```
time.sleep(120)
bot = telebot.TeleBot("5537330469:AAEa_kLtkGnqmxAUhA3e09FsRbqzGUhevMM")
chat_id= str(-704652762)
#comando /ayuda
@bot.message_handler(commands=['ayuda'])
def ayuda(mensaje):
    texto = mensaje.text
    bot.send_message(chat_id,'...BIENVENIDO AL BOT ....')
    bot.send_message(chat_id, text='/temp -comando para ver temperatura y humedad ')
    bot.send_message(chat_id, text='/uv -comando para ver radiación ultravioleta ')
    bot.send_message(chat_id, text='/pluv -comando para ver la lluvia ')
    bot.send_message(chat_id, text='ingrese nuestra pagina web http://ejemplo.ace.com.pe')
#comando /temp
@bot.message_handler(commands=['temp'])
def ayuda(mensaje):
    import Adafruit_DHT
    import time
    sensor = Adafruit_DHT.DHT22 #Cambia por DHT22 y si usas dicho sensor
    pin = 4 #Pin en la raspberry donde conectamos el sensor
    humedad, temperatura = Adafruit_DHT.read_retry(sensor, pin)
    bot.send_message(chat_id, text=("humedad - "+str(humedad)))
    bot.send_message(chat_id, text="temperatura - "+str(temperatura))
#comando /uv
@bot.message_handler(commands=['uv'])
def ayuda(mensaje):
```

```
ser = serial.Serial("/dev/ttyACM0", 9600)
global value
value = (ser.readline().strip())
value = float(value)
bot.send_message(chat_id, text="UV - "+str(value))
ser.close()

#comando /pluv
@bot.message_handler(commands=['pluv'])
def ayuda(mensaje):

#consulta a los datos
import mysql.connector
miConexion = mysql.connector.connect( host="estacion.educatic.org", user= 'educaics_usr_est',
passwd='F5z!xZ5jhSyg', db="educaics_db_estacion" )
cur = miConexion.cursor()
cur.execute( "SELECT metros FROM datos ORDER BY id DESC LIMIT 1" )
for a in cur.fetchall() :
    bot.send_message(chat_id, (str(a)))
miConexion.close()

bot.infinity_polling()
```

Fig. 4. Programación de Bot de Telegram.

Finalmente, las *Pruebas* del prototipo se realizaron en la azotea de una casa en la ciudad de Abancay. Se puso a prueba el funcionamiento del prototipo durante los meses de octubre, noviembre y diciembre del 2022, y en ese periodo el usuario pudo hacer consultas mediante la red social Telegram desde un dispositivo móvil (Fig. 5).



Fig. 5. Pruebas del prototipo con las consultas mediante Telegram

4 Resultados y discusiones

Los resultados de la implementación de este sistema muestran que un usuario puede hacer una consulta mediante comandos, así por ejemplo “/temp”, “/uv”, “/pluv”, esta petición es solicitada mediante el Bot de Telegram, y luego el sistema devuelve el valor leído por el sensor. Las pruebas de concepto realizadas en la azotea de una casa muestran que el sistema funciona adecuadamente y envía los datos al cliente en tiempo real.

Estos resultados de la implementación de este prototipo, están alineados con lo descrito por Mathew y Dida [14], puesto que efectivamente si no se tiene los datos en tiempo real de las variables leídas por una estación meteorológica, entonces prácticamente no se puede aplicar el concepto de “alerta temprana”. Por eso, es importante que los datos recolectados estén disponibles de manera inmediata, porque otras estaciones meteorológicas almacenan los datos de manera local y se tiene que ir a recoger cada 15 o cada 30 días y luego juntar con los datos leídos anteriormente.

5 Conclusiones

Se implementó un sistema para el funcionamiento de una estación meteorológica que permite monitorear las condiciones ambientales de la temperatura, humedad, radiación UV y caudal de agua. El monitoreo se puede hacer mediante una consulta desde un dispositivo móvil mediante el BOT y la API de Telegram. Como trabajo futuro se propone realizar la medición de más variables ambientales, por ejemplo, intensidad de luz, presión barométrica, entre otras; asimismo, se propone integrar una API de Facebook para usuarios que estén más familiarizados con esta red social.

6 Referencias

- [1] J. Baumgartner, S. Zannettou, M. Squire, and J. Blackburn, "The Pushshift Telegram Dataset," *Proc. Int. AAAI Conf. Web Soc. Media*, vol. 14, no. 1, pp. 840–847, May 2020, doi: 10.1609/icwsm.v14i1.7348.
- [2] N. Marechal, "From Russia with crypto: A political history of Telegram," 2018.
- [3] G. C. Lenardo, Y. Irawan, and others, "Pemanfaatan Bot Telegram sebagai Media Informasi Akademik di STMIK Hang Tuah Pekanbaru," *JTIM J. Teknol. Inf. dan Multimed.*, vol. 1, no. 4, pp. 351–357, 2020.
- [4] H. Setiaji and I. V. Paputungan, "Design of telegram bots for campus information sharing," in *IOP Conference Series: Materials Science and Engineering*, 2018, vol. 325, no. 1, p. 12005.
- [5] J. C. de Oliveira, D. H. Santos, and M. P. Neto, "Chatting with arduino platform through telegram bot," in *2016 IEEE International Symposium on Consumer Electronics (ISCE)*, 2016, pp. 131–132.
- [6] M. Franchini *et al.*, "Shifting the paradigm: The Dress-COV telegram bot as a tool for participatory medicine," *Int. J. Environ. Res. Public Health*, vol. 17, no. 23, p. 8786, 2020.
- [7] H. M. Lim *et al.*, "An automated patient self-monitoring system to reduce health care system burden during the COVID-19 pandemic in Malaysia: Development and implementation study," *JMIR Med. informatics*, vol. 9, no. 2, p. e23427, 2021.
- [8] M. H. Pérez and L. A. L. Céspedes, "Bot de Telegram para consultar información de la Universidad de las Ciencias Informáticas," *Ser. científica la Univ. las Ciencias Informáticas*, vol. 14, no. 7, pp. 82–95, 2021.
- [9] Z. K. Hussein, H. J. Hadi, M. R. Abdul-Mutaleb, and Y. S. Mezaal, "Low cost smart weather station using Arduino and ZigBee," *TELKOMNIKA (Telecommunication Comput. Electron. Control.)*, vol. 18, no. 1, pp. 282–288, 2020.
- [10] A. C. Gheorghe and C. I. Stoica, "Wireless weather station using arduino mega and arduino nano," *Sci. Bull. Electr. Eng. Fac.*, vol. 21, no. 1, pp. 35–38, 2021.
- [11] Mercado Libre, "Vantage Pro2," 2023. https://articulo.mercadolibre.com.pe/MPE-443838937-estacion-meteorologica-vantage-pro2-modelo-6152c-nuevo-stock-_JM#position=1&search_layout=stack&type=item&tracking_id=613530da-b971-4276-9975-2de57c9f6016
- [12] Mercado Libre, "Ambient Weather Ws-2000," 2023. https://articulo.mercadolibre.com.pe/MPE-622768274-ambient-weather-ws-2000-estacion-meteorologica-inteligente-c-_JM#position=1&search_layout=stack&type=item&tracking_id=3e27773d-7127-4c31-be09-bab592da0b62
- [13] A. Faid, M. Sadik, and E. Sabir, "An agile AI and IoT-augmented smart farming: a cost-effective cognitive weather station," *Agriculture*, vol. 12, no. 1, p. 35, 2022.
- [14] G. Mathew and M. Dida, "Agro-Meteorological data Collection using a LoRaWAN-Based IoT Sensor Network," 2022.
- [15] A. Rosado-Gómez, A. Quintero-Duarte, and C. D. Meneses-Guevara, "Desarrollo ágil de software aplicando programación extrema," *Rev. Ingenio*, vol. 5, no. 1, pp. 24–29, 2012.