

## IoT Based Fire Detection System for Peat Soil

Wong Shang Le<sup>1</sup>, Shamian Zainal<sup>1\*</sup>

<sup>1</sup>Department of Electronic Engineering Technology, Faculty of Engineering Technology, Universiti Tun Hussein Onn Malaysia, 84600 Pagoh, Johor, MALAYSIA

\*Corresponding Author Designation

DOI: <https://doi.org/10.30880/peat.2022.03.01.041>

Received 17 January 2022; Accepted 11 April 2022; Available online 25 June 2022

**Abstract:** In modern times, the development of IoT technology is faster and more advanced. Fire detection systems also inverted with IoT technology to replace traditional Fire detection systems. Users can get real-time information through the system. So, users can get notification from the system at any time or anywhere. In this project, the microcontroller used is a Raspberry Pi 4. The installation of this fire detection system is implemented to get the information's of the surrounding peat soil. The purpose of this microcontroller is to read data from sensors and communicate with websites and users. The fire detection sensor module, temperature and humidity sensor, and smoke detection sensor are the main sensors in this system. Users can easily view data parameters based on cloud services. User can also find out the conditions around the system in real time such as temperature, humidity and carbon monoxide concentration. The data will always be updated and displayed in the chart to give users can see the data easily. To improve the fire detection process, the Blynk application was implemented to improve the fire detection system. By using this application, users can get notification from this system in order to know the surrounding environment of the system.

**Keywords:** Cloud Storage, Fire Detection System, Raspberry Pi, Blynk App

### 1. Introduction

An IoT based fire detection system is designed to detect any sign of flame or smoke immediately and send the information through online with IoT technology to maximize the efficiency of the user to getting know about the condition of the fire and minimize the time to extinguish the fire. This IoT based fire detection system is make used of the embedded system like Arduino, NodeMCU and Raspberry Pi and built with any other sensors that regarding to the fire detection. It also aids to minimize the time of the user to getting know about the current condition with IoT technology. This IoT based fire detection system implementation has been used in many fields like IoT based forest fire detection system, IoT based fire alarming for workhouse and company [1]. Peat is a partially decomposed organic matter formed primarily from plant material that has collected in the surface organic layer of a soil due to waterlogging, oxygen shortage, excessive acidity, and nutrient inadequacy. Peat soil will normally be found at a warm moist climatic conditions. Peat fires harm the economy by irreversible losses of forest

---

\*Corresponding author: [shamian@uthm.edu.my](mailto:shamian@uthm.edu.my)

2022 UTHM Publisher. All right reserved.

[penerbit.uthm.edu.my/periodicals/index.php/peat](http://penerbit.uthm.edu.my/periodicals/index.php/peat)

land, extracted peat reserves, and peat deposit degradation, as well as the expenditures involved with fire detection and suppression [2].

### 1.1 Raspberry Pi 4

Raspberry Pi is a microcomputer that uses Linux-based operating system. This embedded system is equipped with ARM-architecture CPU, having a built-in RAM card and promote with two USB ports and an Ethernet controller. There are many types of programming languages such as python, java and C++ can be used. It can also install software that can be used. It can serve webpage by installing Apache HTTP Webservice on it [3].

### 1.2 Fire detection system

The fire detection system focus on detect surrounding condition automatically with IoT technology. An IoT platform use to store the parameters such as temperature, humidity and carbon monoxide level in real-time. The feature of this ideas is to provide a cloud platform that can stored the real-time data and a web page which administrators can access and observe the data collected from the sensors [4]. A flame detection sensor module uses to detect the existence of a flame. Flame sensor module is implemented to detect the fire by IR sensor [5]. DHT 22 use to detect surrounding temperature and humidity [6]. MQ-2 gas sensor use to detect the concentration of carbon monoxide with analogue output and MCP 3008 can convert the analogue data from sensor to digital pin for Raspberry Pi [7].

## 2. Materials and Methods

The methods and materials will be arrange for this development of IoT based fire detection system for peat soil. This paper will be focused on the planning and development of the IoT based fire detection system for peat soil.

### 2.1 Materials

The material of the hardware in this project was using: (1) Raspberry Pi 4, (2) MQ-2 gas sensors, (3) DHT 22 sensor, (4) Flame sensor module, (5) MCP 3008. The software will be using: (1) Thonny IDE, (2) ThingSpeak, and (3) Blynk,

For the setup for this system a schematic diagram is shown in Figure 1. This figure shows the connection between the microcontroller, sensors, and other components. A DC power supply with 5 V is connected to the Raspberry Pi. A flame detection sensor module is connected at the GPIO 16 pin. Buzzer is connected to the GPIO 22 and ground. DHT 22 sensor is connected at the GPIO 17. MCP 3008 is used to transfer from analogue input to digital input, this component is used with smoke detection sensor, MQ 2. There are 4 pins from the MCP 3008 are connected to the Raspberry Pi 4. Those pins are define as SPI pins which will connect to the GPIO 8, GPIO 9, GPIO 10 and GPIO 11. From MCP 3008, the pin number of 16 will be connected to the 5V and pin number 9 will connected to the ground. Then, the Channel 0 input from the MCP 3008 is connected with the analogue output from the MQ 2 smoke detection sensor. The digital output from the MQ 2 smoke detection sensor will be connected to the GPIO 26 from Raspberry Pi. MCP 3008 is connect to 5 V power supply and the ground channel is connect to the ground from Raspberry Pi 4. The clock signal CLK, digital input  $D_{in}$ , digital output  $D_{out}$ , and shutdown pin SHDN in MCP 3008 is connect with GPIO 11, GPIO 10, GPIO 9 and GPIO 8 respectively.

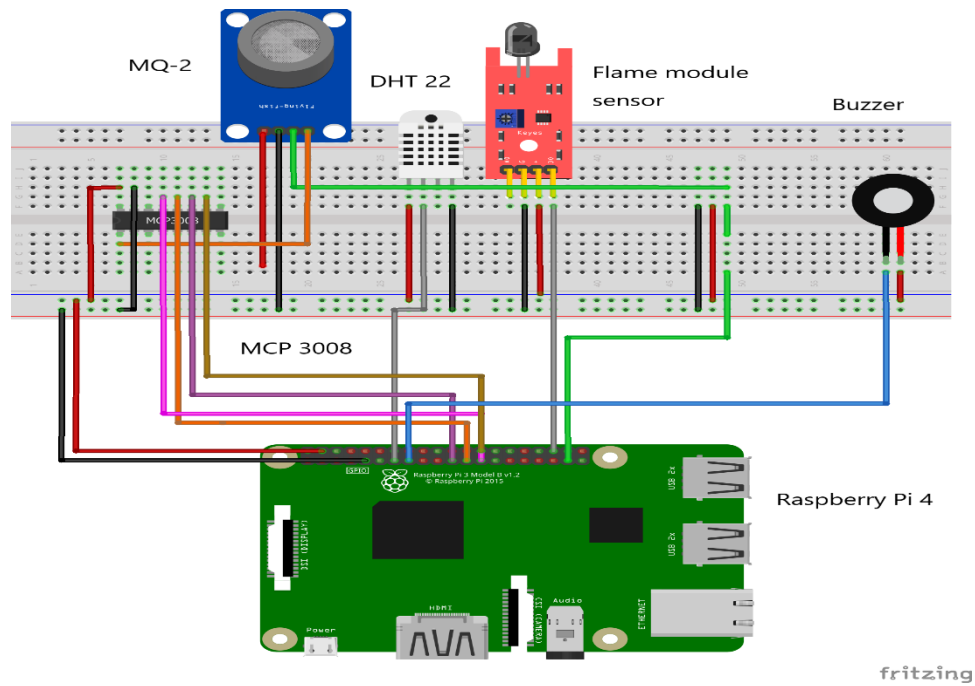


Figure 1: Schematic Diagram

### 2.2 Block diagram

Figure 2 shows the fire detection system consist of MQ 2, DHT 22, flame module sensor and a buzzer for normal operation. The block diagram show that there is a connection between sensors and Raspberry Pi. Other than that, a cloud storage ThingSpeak and blynk app are also connect with Raspberry Pi to store the real time data and notify the user.

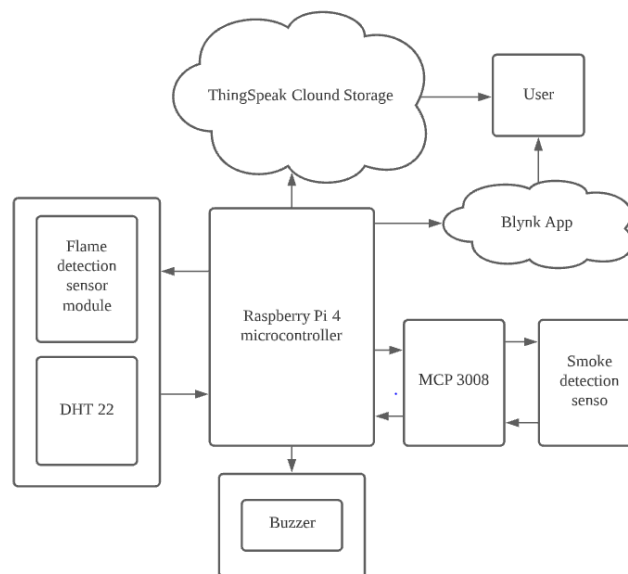


Figure 2: Block Diagram

### 2.3 Thonny IDE

Thonny is a free Python Integrated Development Environment (IDE) with Python programming language. It can provide service to the device that run with Window, macOS and Linux. In this project,

Raspberry Pi 4 is used, and it runs the Raspbian OS by the Linux Kernel. Therefore, Raspberry Pi 4 can use Thonny IDE to write the program code and upload to the microcontroller and run it directly.

Thonny IDE is user-friendly toward the beginners because it is able to show syntax error for the users. This feature allows the users to find out their error and modify it to the correct code. In this project, Thonny IDE has several modes for stepping through the code, including step-by-step expression evaluation, comprehensive call stack visualisation, and a mode for explaining the principles of references. A set of Python program code was done by using the Thonny IDE to run sensors and transfer the data to the IoT platform [8].

#### 2.4 ThingSpeak

ThingSpeak is an application of IoT platform, which allows the users to connect the platform and save real-time data in the cloud to develop any IoT application. ThingSpeak offers some features such as real-time data collection, data processing, visualizations, apps, and plugins. MATLAB support helps you act on data. Sensor data can be easily integrated and sent from Arduino or Raspberry Pi or any other IoT gateway.

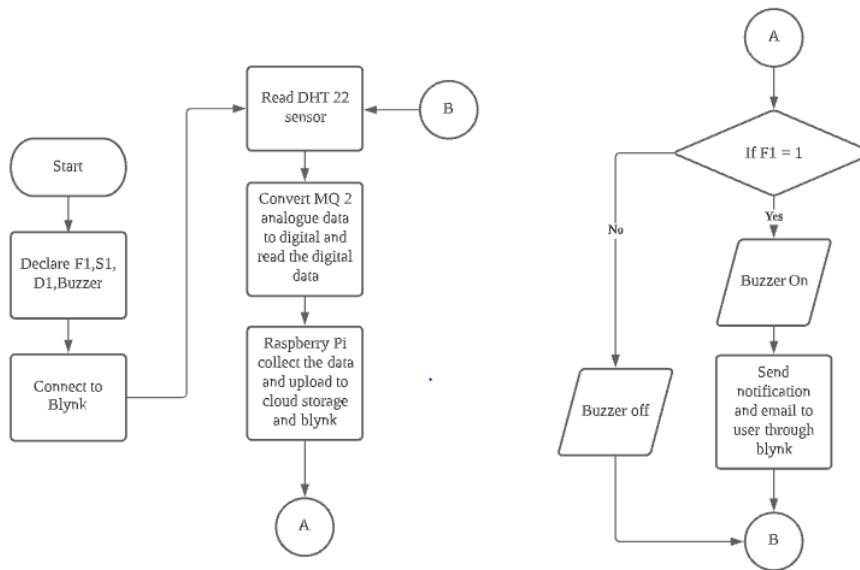
ThingSpeak is a free IoT platform which can provide a graphical interface with data display. The graph will show up the data collected from the sensor such as temperature or humidity data and available at any place if there is a network provided to connect with ThingSpeak. It is suitable to store and monitor the real-time sensor data from Raspberry Pi 4 in the cloud storage [9].

#### 2.5 Blynk

Blynk is a hardware-agnostic IoT (Internet of Things) platform supported by both iOS and Android applications based which help to control Arduino, Raspberry Pi, ESP 8266 and others to send the data over the Wi-Fi, ethernet, cellular like GSM, 2G, 3G, 4G/LTE as well as the coming 5G. Blynk is a digital type dashboard where the user can customize a graphic interface for their project by drag and drop the needed widgets. By using this application, the smart fire detection system relates to it, and the user can get the notification from the system. The Blynk system will get the hardware of Raspberry Pi 4 online as well as connect it to Blynk Cloud. Blynk application will be used as a remote for the user [10].

#### 2.6 Overall Programming Flowchart

For the overall programming in Figure 3, the program starts with declaring all the variables. Then, the Raspberry Pi 4 will then be connected to the Blynk with the provided authorize token.



**Figure 3: Flowchart**

After that, the program will proceed with reading the surrounding temperature and humidity data from DHT 22 sensor. The output from the MQ 2 smoke detection sensor will be converted from analogue data to digital data by using Analogue Digital Converter. Next, Raspberry Pi 4 will read the digital data. The next part is the notification system, a main loop is use for looping the program. When the flame detection sensor module is being triggered the buzzer will on followed with the notification to user through blynk and an email will also send through from blynk to user with certain email account. If the flame detection sensor module is not triggered, the buzzer will off and proceed to looping. Lastly the programming will start the delay 30s and then it will be looping back to the reading sensors data for detection and notification system.

### 3. Results and Discussion

This chapter is about the data uploaded and store to the ThingSpeak platform and the Blynk interface with the hardware.

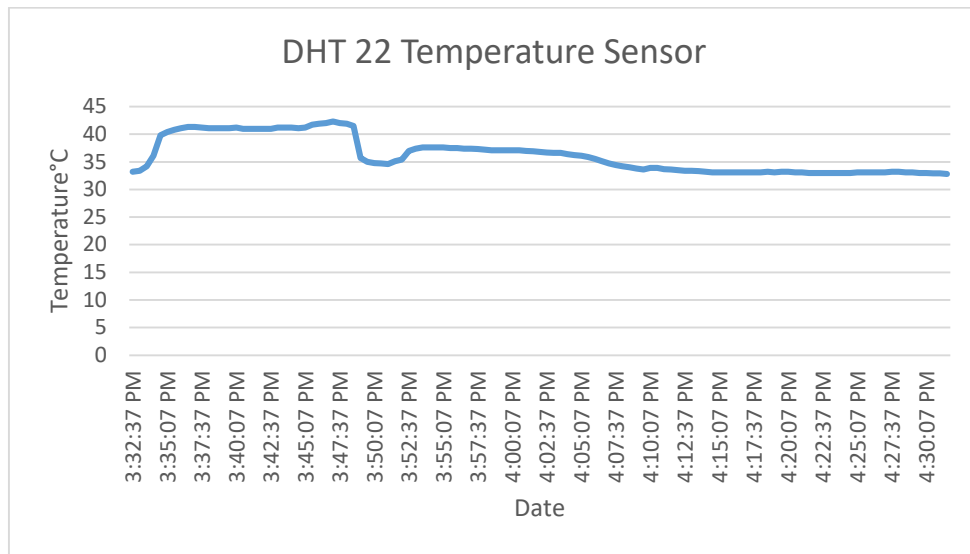
#### 3.1 Results

The findings of results will be displayed about the ThinSpeak will be used as an IoT platform to store the variety of the surrounding condition by showing those data with graph from different sensors for fire detection system. Besides that, the Blynk interface with the hardware is being include with the interaction between the app with the hardware and the communication in between hardware, Blynk, and email.

#### 3.2 DHT 22 temperature sensors

The DHT 22 sensor detect the surrounding temperature data and the Raspberry Pi 4 read DHT 22 data and upload the data to the customized IoT platform which is ThingSpeak. From the graph, The overall process for DHT 22 take an hour to detect the surrounding temperature. From the graph, the data show that the highest surrounding temperature of the peat soil is 41.9 °C while the lowest surrounding temperature of the peat soil is 32.8 °C. In between 3.36 pm to 3.48 pm, the DHT 22 probably has no barrier or cloud to block the sunlight and make the temperature start to climb and reach the highest point. In between 3.54 pm to 4.30 pm, the sunlight probably block by some obstacle or cloud and make the surrounding temperature start to decrease and reach a lowest surrounding temperature point at 4.30 p.m. Lastly, the graph also show that there is a 30 seconds interval for the Raspberry Pi 4

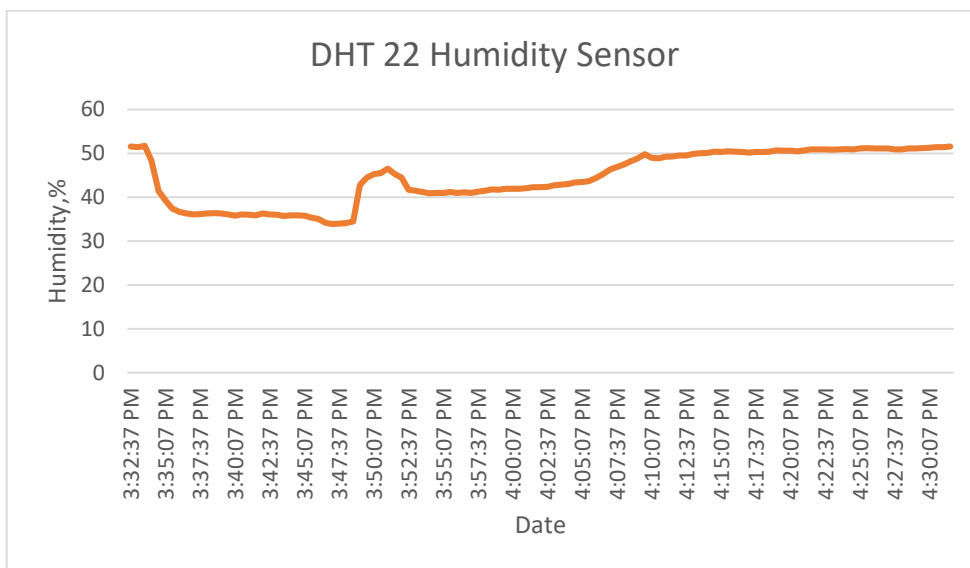
to upload the data to the IoT platform. This show that the cloud storage for the surrounding temperature data is successfully store in the ThingSpeak which allow the user can observe the previous data by accessing ThingSpeak.



**Figure 4: Graph of DHT 22 Temperature sensors**

### 3.3 DHT 22 humidity sensors

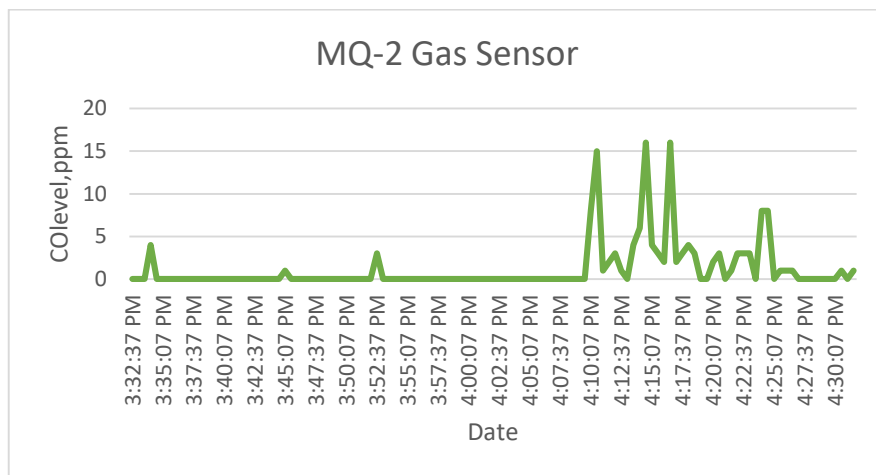
From the graph, the data that being displayed in this graph is surrounding humidity percentage. The data start displays the humidity percentage at 51.6 0% and then drop to 34.00 % in 5 minutes later. The graph also show that the highest surrounding humidity of the peat soil is 51.60 % while the lowest surrounding temperature of the peat soil is 34%. At 3.47p.m the DHT 22 detected the surrounding humidity start return to about 45.00 % for 5 minutes and the start to drop off to around 40.00 %. Then, the graph show that the data have a linear grow until 51.60 % at 4.07p.m. The highest humidity percentage shown in between 4.10 pm to 4.30 pm since in that time the surrounding temperature is quite low. The lowest humidity percentage shown in between 3.36 pm to 3.48 pm, since the surrounding temperature is quite high at that time. Those results show that there is a relationship between the surrounding temperature and surrounding humidity. Therefore, the reasons that affect the surrounding temperature can also influence the surrounding humidity.



**Figure 5: Graph of DHT 22 Humidity sensors**

### 3.4 MQ-2 Gas Sensor

The MQ-2 gas sensor can also detect the surrounding CO level which is concentration of Carbon Monoxide. The data that being displayed in this graph is surrounding concentration of Carbon Monoxide. From the graph, there are only between 0 – 5 of CO level at the first 30 minutes, Then, the data start to rise to highest surrounding concentration of Carbon Monoxide of the peat soil which is 16 and have a change in between 0 – 16 of CO level in another 30 minutes later. This concentration of Carbon Monoxide can be conclude as low since it less than 50 while the lowest surrounding temperature of the peat soil is 0 [11]. The highest concentration of Carbon Monoxide shown in 4.15 pm while the lowest concentration of Carbon Monoxide shown at many time. This results show that the surrounding concentration of Carbon Monoxide of peat soil is normal and also no smoke or flame occurred.



**Figure 6: Graph of MQ-2 gas sensor**

### 3.4 Blynk app interface with hardware

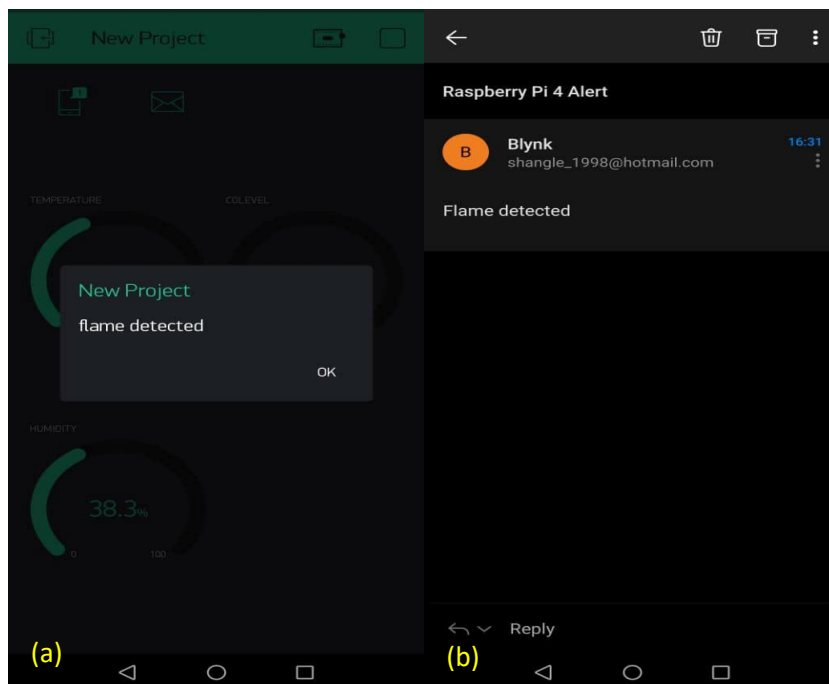
The app interface show that there is a connection between the Raspberry Pi 4 and blynk. In the blynk app there are 5 widgets are added from the widget box. The 1<sup>st</sup> widget with a phone icon is the notification widget. The blynk will send notification when there is an event occur at the python coding from the Raspberry Pi 4. Then, the next widget is an email widget this show that it will send an email through the blynk to the user with fixed email account. In the email widget it can adjust user email that should blynk send to. Lastly, there are 3 same widgets which are gauge from widget box. The three gauge being used to display the current temperature, humidity and carbon monoxide level respectively. It allow the user to check the surrounding condition from the blynk app. Those widgets are set with fixed virtual pin to receive the data from the Raspberry Pi 4.



**Figure 7: Blynk App**

### 3.5 Blynk app with notification system

It shows the interaction between Raspberry Pi 4 with Blynk App. The Raspberry Pi 4 is being coded as when the flame module sensor detects fire, the Raspberry Pi 4 will read the output pin from the flame module sensor and send the notification through the Blynk app. The Blynk will print a 'flame detected' message to the user as scripted in the programming code. At the same time, an email will also be sent to the user through the Blynk. When an event is triggered, the Blynk will send an email to the user with 'Raspberry Pi 4 Alert Flame detected'. The phrase is being scripted in the programming code.



**Figure 8: (a) Blynk notification, (b) E-mail notification**



#### 4. Conclusion

All over this paper, the fire detection system able to read surrounding parameters with DHT 22 sensors and MQ-2 gas sensor. The system can detect the surrounding fire occurrence with flame module sensor. The system able to communicate with Blynk app and notify the user via e-mail or the app. User can also access the real-time data in Blynk app. The system will also upload the data every 30 seconds to ThingSpeak, user able to access the previous surrounding condition. Raspberry Pi 4 function as a microcontroller to read data and give order to Blynk app in this fire detection system

#### Acknowledgement

The author would like to thank the Faculty of Electrical and Electronic Engineering, Universiti Tun Hussein Onn Malaysia for its support.

#### References

- [1] M. Trinath Basu, R. Karthik, J. Mahitha, and V. Lokesh Reddy, "IoT based forest fire detection system," *Int. J. Eng. Technol.*, vol. 7, no. 2, pp. 124–126, 2018, doi: 10.14419/ijet.v7i2.7.10277.
- [2] A. Filkov, V. Leroy-Cancellieri, D. Cancellieri, D. Gladky, and A. Simeoni, *Modeling Peat-Fire Hazards: From Drying to Smoldering*, vol. 4. Elsevier B.V., 2015.
- [3] C. Hariveena, K. Anitha, and P. Ramesh, "IoT-based Fire Detection and Prevention System," *IOP Conf. Ser. Mater. Sci. Eng.*, vol. 981, no. 4, 2020, doi: 10.1088/1757-899X/981/4/042080.
- [4] M. Mahbub, M. M. Hossain, and M. S. A. Gazi, "Cloud-Enabled IoT-based embedded system and software for intelligent indoor lighting, ventilation, early stage fire detection and prevention," *Comput. Networks*, vol. 184, p. 107673, 2021, doi: 10.1016/j.comnet.2020.107673.
- [5] I. A. Taha and H. M. Marhoon, "Implementation of controlled robot for fire detection and extinguish to closed areas based on Arduino," *Telkomnika (Telecommunication Comput. Electron. Control.*, vol. 16, no. 2, pp. 654–664, 2018, doi: 10.12928/TELKOMNIKA.v16i2.8197.
- [6] N. Abdul Khaleq, O. Khalaf, and D. Addulsahib, "IOT fire detection system using sensor with Arduino," *Rev. Aus*, vol. 26, no. 1, pp. 74–78, 2019, doi: 10.4206/aus.2019.n26-7/.
- [7] R. Hayat Khan, Z. Ayub Bhuiyan, S. Sharar Rahman, and S. Khondaker, "A Smart and Cost-Effective Fire Detection System for Developing Country: An IoT based Approach," *Int. J. Inf. Eng. Electron. Bus.*, vol. 11, no. 3, pp. 16–24, 2019, doi: 10.5815/ijieeb.2019.03.03.
- [8] A. Imteaj, T. Rahman, M. K. Hossain, M. S. Alam, and S. A. Rahat, "An IoT based Fire Alarming and Authentication System for Workhouse using Raspberry Pi 3," *ECCE 2017 - Int. Conf. Electr. Comput. Commun. Eng.*, no. February 2010, pp. 899–904, 2017, doi: 10.1109/ECACE.2017.7913031.
- [9] M. D. Setiawan, P. Studi, T. Informatika, F. Ilmu, and U. Katholik, "LOCATION BASED FIRE DETECTION , WITH NEAREST," vol. 2, no. 2, pp. 77–88, 2019.
- [10] "1. What is Blynk? - Tech Explorations." <https://techexplorations.com/guides/blynk/1-what-is-blynk/> (accessed Jan. 12, 2022).
- [11] "Carbon monoxide Levels that Sound the Alarm | Kidde." [https://www.kidde.com/home-safety/en/us/support/help-center/browse-articles/articles/what\\_are\\_the\\_carbon\\_monoxide\\_levels\\_that\\_will\\_sound\\_the\\_alarm\\_.html](https://www.kidde.com/home-safety/en/us/support/help-center/browse-articles/articles/what_are_the_carbon_monoxide_levels_that_will_sound_the_alarm_.html) (accessed Jan. 13, 2022).