IoT Implementation for Server Room Security Monitoring Using Telegram API

by Mochamad Fajar Wicaksono

Submission date: 19-Nov-2022 09:00AM (UTC+0700)
Submission ID: 1958309661
File name: JURNAL.pdf (2.15M)
Word count: 5617
Character count: 27374

International Journal on Advanced Science Engineering Information Technology

Vol.12 (2022) No. 5
ISSN: 2088-5334

IoT Implementation for Server Room Security Monitoring Using Telegram API

Mochamad Fajar Wicaksono [a,*], Myrna Dwi Rahmatya [b], Ilham [c]

[a] Computer Engineering Study Program, Universitas Komputer Indonesia, Bandung, Indonesia
[b] Informatics Management Study Program, Universitas Komputer Indonesia, Bandung, Indonesia
[c] Computer Systems Study Program, Universitas Komputer Indonesia, Bandung, Indonesia
Corresponding author: *mfajarw@email.unikom.ac.id

***Abstract*— This research aims to create a system that can monitor server room security by utilizing IoT technology, controllers, sensors, actuators, and Telegram API. The system made includes security for the indoor and outdoor parts. The method used in this study is experimental. The output obtained by the user is telegram text messages, photos, and video. The main controller on this system is the Raspberry Pi. This study showed that if the server room temperature is >230C, the system will send a message via Telegram. If the PIR sensor_1 or PIR sensor_2 detects an intruder or the measured distance is <300cm from the side of the door, the system will send text messages, photos, and videos via Telegram. That way, the user can determine the next steps related to the condition of the server room. However, if the RFID tag is recognized, the solenoid door lock, PIR, and ultrasonic will be deactivated. Other features, Users can ask the system to record a video of the room conditions and also messages related to room temperature through commands sent via Telegram messages. Testing is carried out using 3 scenarios: testing command, testing regarding valid access, and testing regarding invalid temperature and intruder detection. Overall test results show all these tests were successful 100% and followed the objectives and program scenarios. This tool has fulfilled the objective where this tool can monitor the condition and security of the server room by utilizing IoT technology.**

***Keywords*— IoT; server room; monitoring; Raspberry Pi; Telegram.**

Manuscript received 8 Dec. 2020; revised 12 Jun. 2021; accepted 10 Jul. 2021. Date of publication 31 Oct. 2022.

I. INTRODUCTION

The server room is used to store servers that contain applications and databases. Keeping the server room safe is very important because it is a data center that stores all information related to company operations. Today, security can mean either physical security, as in physical access control or logical security. Physical security is needed to avoid physical attacks on the server room. One form of physical attack on the server room is an abuse of privilege. This attack physically requires illegal access to the system, such as physical breaches and theft. Related to physical security, procedures or mechanisms are needed to prevent physical attacks. A well-built management strategy can counter-espionage and theft [1]. The prevention mechanism can also minimize risk, find the cause of the damage, and determine the appropriate steps to overcome it. Implementation of physical security and effective policies and procedures are as important as staff awareness [2].

Server rooms can be monitored remotely by implementing the Internet of Things (IoT). IoT enables the collection and sharing of data with other devices over the internet. With IoT, information is collected, stored, and can be used for data analysis processes [3]. A government decree states that temperature and humidity in the server room are some of the important things to monitor [4]. In Indonesia, there are already standards related to temperature and humidity set by the government in 2017. The standard temperature is in the range of 21^0C-23^0C, and humidity is 45% to 60%. High humidity can cause corrosion and short circuits. Apart from temperature and humidity, many other things related to server room security must be monitored, mainly related to defense mechanisms.

Much research is related to this server room for monitoring and security. Research in 2017 is also still focused on the problem of temperature and humidity, and no features related to warnings messages are given to users [5]. A study in 2017 has added fuzzy logic to design intelligent air conditioners, but it just focuses on temperature [6]. Furthermore, in the

same year, namely 2018, there was research on this server room. However, this research only focuses on the security of the server room door by using an application and a door lock [7]. In 2019, new research related to this server room could send notification messages via the Telegram application. However, these studies only focus on temperature and humidity [8].

Nearly all of the previous studies focused on one part of server room temperature and humidity and one that focused solely on server room entrances. So, in general, no one directly discusses monitoring the conditions and security of the server room.

In this research, a system was made to monitor the condition and security of the server room. The security scope of this research is the inside and the outside of the server room. The user's output is telegram text messages and video telegrams. The main controller on this system is the Raspberry Pi. Raspberry Pi will read input from ultrasonic sensors, PIR sensors, DHT11 sensors, and cameras. The PIR sensor is here used to detect human presence in the server room. In this study, two PIR sensors were used, placed across from each other inside the server room. DHT11 sensor is used to measure the temperature of the server room. If the temperature is outside the threshold, the user will get a message related to this, and the buzzer in the room will also be active. On the outside, placed the RFID reader and keypad. This is related to access rights to the server room. If the password is entered or the RFID tag is recognized, the Raspberry Pi will deactivate the solenoid door lock to open the door and turn off the ultrasonic sensor and all PIR sensors. Users can ask the device to record a video of the room conditions and also messages related to room temperature through commands sent via Telegram messages. If there is an incident, for example, the door is forcibly opened, the distance the ultrasonic sensor reads will get smaller, so the Raspberry Pi will activate the buzzer and send a telegram message to the user regarding the incident. Then Raspberry will wait for the next command. The next action is obtained from the telegram message sent by the user. This can be a request for a video recording related to the current condition or a photo taken.

II. Materials and Methods

As mentioned above, IoT technology can be implemented and utilized to monitor the condition of a room from afar, especially in this case, the server room. The research method used is the experimental method.

Fig. 1 Diagram process of the research methods

In the implementation process, a series of designs, simulations, and experiments were carried out directly following the theoretical studies that had previously been carried out to obtain the expected results following the test scenario to achieve the initial objectives of the research. The stages of the research carried out are shown in Fig. 1.

A. Study of Literature

The first step is to conduct a literature study related to server rooms, research related to server rooms, and determine the system requirements to be made related to hardware and software.

B. Design

At this stage, design related to hardware, software, and mechanical design is carried out based on theoretical studies that have been carried out in the previous stage. In this study, the Raspberry Pi 3B is used as the system's main brain, which will process all input and output. Raspberry Pi 3B is a mini PC and it has peripheral ports and GPIO pins [9]. It has SoC BCM2837 and 1GB RAM [10]. Raspberry Pi can be used for both monitoring and controlling [11]. Python language is used to create programs on the Raspberry Pi [12]. The input part of this system includes a DHT11 sensor, two PIR sensors, an ultrasonic sensor, an RFID reader, a keypad, and a camera module.

On the other hand, the output includes Buzzer, 16x2 LCD, Relay, and Solenoid Door Lock. This telegram messaging application functions as input and output on this system. A block diagram is generated in the hardware design section that shows the overall system architecture, as shown in Fig. 2.

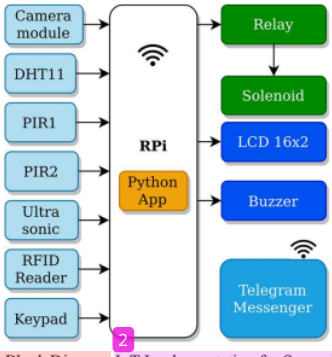

Fig. 2 System Block Diagram IoT Implementation for Server Room Security Monitoring Using Telegram API

DHT11 is a sensor module that can be used to measure temperature and humidity [13], [14]. The temperature measurement range on this sensor is 0^0C-50^0C. The humidity measurement range on this sensor is 20%-90% [14]. This sensor has a high level of accuracy and can be easily integrated into a system [15]. In this study, DHT11 is placed in the server room, and Raspberry Pi will read data from DHT11. PIR sensor is a type of motion sensor [16]. To detect human movement, we can use this sensor [17]. This study uses the PIR HC-SR501 sensor module. This sensor module has digital output [18].

In this study, two HC-SR501 sensors were used. The two sensors are placed opposite in the server room. In this case, it

is intended to detect intruders who enter the server room via the roof. Ultrasonic sensors are commonly used to measure distances. In this study, the HC-SR04 ultrasonic sensor was used. This ultrasonic sensor works by sending sound pulses with a high frequency [19]. The measured distance is obtained from the initial time of sending the ultrasonic signal until it is received at the echo pin, then multiplied by the speed of sound (340m/s) and finally divided by 2 [20]. The distance that can be measured with this sensor is 2cm to 4m [21]. In this study, the HC-SR04 ultrasonic sensor was placed on the wall in the room near the entrance. The sensor will send a legible distance to the Raspberry Pi. If there are thieves who try to enter by breaking the entrance, the incident will be immediately known because the distance read by the ultrasonic sensor is outside the range specified in the program.

Furthermore, the Raspberry Pi will activate the buzzer and send a message to the user regarding the incident. A special camera module for the Raspberry Pi is also used to take photos and videos. This study uses the Raspberry Pi Camera Module V2. This module uses Sony IMX219 with 8 Megapixel image sensor [22]. This module has 1us exposure time and provides 200FPS for video [23]. The solenoid door lock is used to lock server room doors. To activate and deactivate the solenoid, it can be done in two ways, using an RFID card or entering a password via keypad 4x4. The keypad is a 4x4 keypad in this system, while the RFID reader is MFRC-522. This RFID uses electromagnetic fields to work [24]. This MFRC-522 module works at a frequency of 13.56 MHz Tag reading distance between 3cm-5cm [25]. Communication with this module is done via the SPI pin. [26], [27].

In this system, the telegram application is used so that users can interact with the tool. Telegram is a free instant messaging application and can be used on various platforms [28]. In this application, users can send various types of data such as images, videos, and others. The advantage of this application is open-source API, and users can create bots for their needs. One of the features used by Telegram in this study is the Telegram Bot API. This bot is a third-party application that can run on the Telegram application [29].

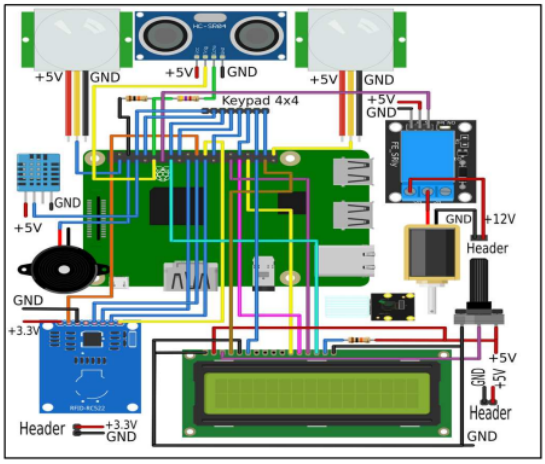

Fig. 3 The electronic circuit of IoT Implementation for Server Room Security Monitoring Using Telegram API

Based on the theoretical studies obtained and the block diagrams that have been made, the next step is to translate these shapes into a circuit design to realize this system as shown in. Fig. 3. After creating a series of systems, the next step is software design. An algorithm is created to create a Python program for this system at this stage. The following algorithm is an outline and illustrates how this system will work:

1. Begin
2. GPIO and variable initialization
3. Read incoming commands from Telegram
4. Read DHT11, PIR1 sensor, PIR2 sensor, Ultrasonic sensor
5. If command="/photo" then
   - capture photo and send it via Telegram
   - back to step 3
6. If command="/video" then
   - capture video and send it via Telegram
   - back to step 3
7. If temp > 23 then
   - activate a buzzer, send a warning message via Telegram, and go to step 3
8. If PIR1=1 then go to point 17
9. If PIR2=1 then go to point 17
10. If distance < 300 then go to point 17
11. Read RFID reader
12. If code number recognized, then go to step 13
13. Access granted procedure:
    - give False value to lock variable
    - deactivate solenoid door lock,
    - deactivate DHT11,
    - deactivate sensor PIR1,
    - deactivate sensor PIR2,
    - deactivate sensor ultrasonic.
    - Capture photo and video.
    - Send photo and video via Telegram.
    - Read keypad
    - If lock = False, then read keypad again
    - If lock = True, then
      - activate solenoid door lock,
      - activate DHT11,
      - activate sensor PIR1,
      - activate sensor PIR2,
      - activate sensor ultrasonic.
      - back to step 3
14. If code number not recognized, then read input keypad
15. If input password from keypad True, then go to step 13
16. If input password from keypad false 3 times, then go to step 3
17. Activate buzzer, send warning message via Telegram, capture photo, record video, send photo and video via Telegram. After that, back to step 3.

From the algorithm above, the program flow of this tool can be explained. When it is first turned on, the Raspberry Pi will first check whether there is an order from the user sent via Telegram. Furthermore, if an order is received, Raspberry will store it in a variable. The Raspberry Pi will read all output from the DHT11, PIR_1, PIR_2, and ultrasonic sensors. All readings are stored in variables first. Next, all variables will be compared. If there is a "photo" command, the Raspberry Pi

will take a photo at that time and send the results to the user via Telegram. If there is a "video" command, the Raspberry Pi will take the current video and send the results to the user via Telegram. If the measured temperature is more than 23^{0}C, the Raspberry Pi will send a warning message to the user via Telegram. If the PIR_1 sensor or PIR_2 sensor detects movement, the Raspberry Pi will send the user warning messages, photos, and videos via Telegram. If the distance read from the ultrasonic sensor is less than 300cm, the Raspberry Pi will send warning messages, photos, and videos to the user via Telegram. When all the parts above are safe, the Raspberry Pi will read data from the RFID reader regarding the door opening, which involves deactivating the solenoid door lock, PIR_1 sensor, PIR_2 sensor, and ultrasonic sensor. Raspberry will also take photos and videos at that time for documentation purposes related to people accessing the server room. If the user forgets to bring his RFID tag, the user can still enter the server room by entering the password via the 4x4 keypad. The opportunity to enter this password is 3 times. The next process that occurs when the password is recognized is the same as the process when the RFID tag is recognized. If the user leaves the server room and locks the door using the keypad, all previously deactivated things such as the solenoid door lock, PIR_1 sensor, PIR_2 sensor, and ultrasonic sensor will be activated again.

After getting materials and tools suitable for use according to the block diagram, a design is made regarding the placement of each component of this tool. The tool placement design is shown in Fig. 4.

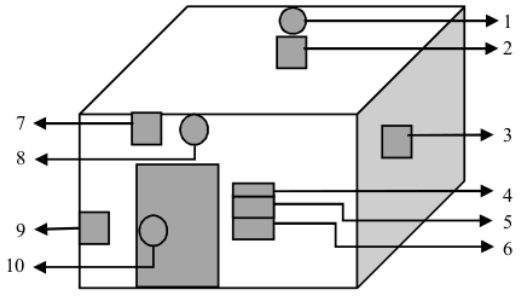

Fig. 4 Design of component placement in the server room

The following is a description of the names of all hardware components placed according to Fig. 4 in this study:

1. PIR1 sensor
2. Raspberry Pi and Pi Camera V2
3. DHT11 sensor
4. LCD 16x2
5. Keypad
6. RFID Reader
7. Buzzer
8. PIR2 sensor
9. Ultrasonic sensor
10. Solenoid Door Lock

C. Simulation and Verification

At this stage, hardware and software simulations are carried out based on schematics and programs that have been made in the previous stage. Simulation is in the form of debugging this system program and verifying the circuit according to the schematic.

D. Implementation

At this stage, all hardware and software are combined to synchronize the two things to suit the initial objectives of this research.

E. Testing

At this stage, testing is carried out on the system created by testing the tool's functionality based on commands sent from the telegram application and testing the system according to the scenario created.

III. Results and Discussion

The tests carried out in this study include 3 parts: testing the giving of commands to test the response of the tool, testing when access rights are valid, and testing when an intruder tries to enter the server room.

A. Testing Commands

After all the parts of this tool are installed in the server room, the test command sent via the telegram application is carried out. This serves to ensure that the Raspberry Pi can read commands sent by users and that the Raspberry Pi responds according to the commands given. The initial command to start the bot is "/start". When receiving this command, the Raspberry Pi must respond by providing some information regarding using the bot command using this telegram application. The results of this test are shown in Fig. 5.

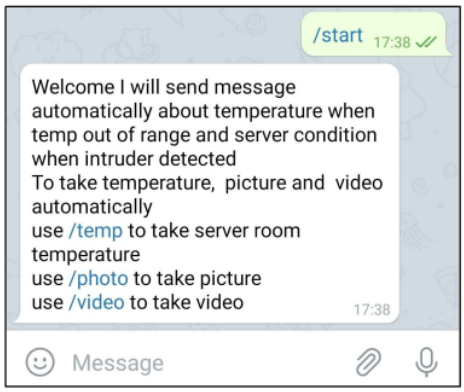

Fig. 5 Test "/start" command

The next test is testing every command that has been determined on the tool. The commands tested were the commands "/temp", "/photo" and "/video". Command "/temp" is used to get the measured temperature at that time. The command "/photo" is used to get a photo in the current server room. The command "/video" is used to get the video in the current server room. Each test on this command is performed 10 times. When the user sends the "/temp" command, the Raspberry Pi will send the current measured temperature and humidity values 10 times. The test results for the "/temp" command is shown in Table 1.

TABLE I
A COMMAND FOR ACTUAL TEMPERATURE AND HUMIDITY

Test No.	Command (Send)	Time (Send)	Response received	Time (Receive)
1.	/temp	17:38	√	17:38
2.	/temp	19:32	√	19:32
3.	/temp	20:11	√	20:11
4.	/temp	20:45	√	20:45
5.	/temp	21:10	√	21:10
6.	/temp	08:00	√	08:00
7.	/temp	08:30	√	08:30
8.	/temp	10:00	√	10:00
9.	/temp	12:15	√	12:15
10.	/temp	13:00	√	13:00

The time for testing the "/temp" command is varied, it is also intended to check that the tool is still running. From the test results in table 1, it can be seen that all the "/temp" commands sent have been responded to well by the tool with a 100% success percentage from 10 testing. The next test is manually testing the photo and video capture requests using the "/photo" and "/video" commands. The test results for this command are shown in Table 2.

TABLE II
THE COMMAND FOR TAKING PICTURE MANUALLY

Test No.	Command (Send)	Time (Send)	Photo received	Time (Receive)
1.	/photo	15:35	√	15:35
2.	/photo	16:30	√	16:30
3.	/photo	18:20	√	18:20
4.	/photo	19:00	√	19:00
5.	/photo	20:50	√	20:50
6.	/photo	21:15	√	21:15
7.	/photo	07:40	√	07:40
8.	/photo	09:28	√	09:28
9.	/photo	10:18	√	10:18
10.	/photo	10:50	√	10:50

When the test is carried out regarding the "/photo" command, it is carried out in various ways, it is also intended to check that the tool is still working. From the test results in table 2, it can be seen that all the "/photo" commands sent have been responded to well by the tool with a 100% success percentage from 10 testing. The next step is to test the video sending request command using the "/video" command regarding the current server room situation. The test results for the "/video" command that was performed are shown in Table 3.

TABLE III
THE COMMAND FOR TAKING VIDEO MANUALLY

Test No.	Command (Send)	Time (Send)	Video received	Time (Receive)
1.	/video	16:45	√	16:46
2.	/video	17:05	√	17:05
3.	/video	18:00	√	18:01
4.	/video	18:20	√	18:20
5.	/video	19:15	√	19:15
6.	/video	21:00	√	21:01
7.	/video	08:36	√	08:36
8.	/video	09:10	√	09:11
9.	/video	10:15	√	10:16
10.	/video	12:05	√	12:05

The recording duration in this test is 15 seconds. The time for testing the "/video" command is varied, it is also intended to check that the tool is still working. From the test results in table 3, it can be seen that all "/video" commands sent have been responded to well by the tool with a 100% success percentage from 10 testing. Testing of all commands that exist in this system has been successful 100% following every testing process carried out.

B. Testing Regarding Valid Access

The next test is a test related to valid access. This test involves 3 users who have been given an RFID tag card with the test code names T1, T2, and T3. The user will "tap" the card in this test to enter the server room. If the card is recognized, the door lock solenoid and all sensors are not active, and Raspberry sends photos and videos to the admin. This testing process was also carried out using an unrecognized RFID tag card with the card name "T4" (not registered in the system). The test results are shown in Table 4.

TABLE IV
RFID TAG TESTING

Desc	RFID tag Recognized	Solenoid door lock	PIR 1	PIR 2	Ultrasonic	Photo and Video
T1	√	x	x	x	x	Sent
T2	√	x	x	x	x	Sent
T3	√	x	x	x	x	Sent
T4	x	√	√	√	√	No
T4	x	√	√	√	√	No
T2	√	x	x	x	x	Sent
T3	√	x	x	x	x	Sent
T4	x	√	√	√	√	No
T4	x	√	√	√	√	No
T1	√	x	x	x	x	Sent

The test results are shown in table 3 explain that when the RFID card tag is recognized, the door lock solenoid will be deactivated so that the door can be opened. Furthermore, the PIR_1 sensor, PIR_2 sensor, and Ultrasonic sensor are also deactivated, and the Raspberry Pi sends photos and videos when this happens. When the RFID tag is not recognized, the opposite happens where all actuators and sensors remain active. Of the 10 times, this testing process is all successful with a success percentage of 100%.

The next testing process is testing access to the server room using a password input via the 4x4 keypad. This test is performed 10 times by entering the passwords of 3 users. Recognized passwords are given the test code names P1, P2, and P3. The unknown passcode is given the test code name P4. The test results related to these access rights are shown in Table 5.

The test results in table 5 show that when the input password is recognized, the door lock solenoid, PIR_1 sensor, PIR_2 sensor, and ultrasonic sensor will be deactivated. In other words, the user can enter the server room. At the same time, the Raspberry Pi will also take photos and videos and then send them to the admin.

When the wrong password is entered, the actuator and all sensors remain active, and the unauthorized person cannot enter the server room. Raspberry Pi will also not send photos and videos at that time because the server room is locked, and no one is detected in the server room.

TABLE V
KEYPAD PASSWORD TESTING

Desc	Password Recognized	Solenoid door lock	PIR 1	PIR 2	Ultrasonic	Photo and Video
P3	√	x	x	x	x	Sent
P4	x	√	√	√	√	No
P1	√	x	x	x	x	Sent
P2	√	x	x	x	x	Sent
P4	x	√	√	√	√	No
P1	√	x	x	x	x	Sent
P2	√	x	x	x	x	Sent
P4	x	√	√	√	√	No
P3	√	x	x	x	x	Sent
P1	√	x	x	x	x	Sent

C. Testing Regarding Invalid Temperature and Intruder Detection

The first test is a test related to temperatures that are outside the predetermined limits. Wherein in this system, the specified threshold is 21^0C-23^0C. The Raspberry Pi must send a message to the telegram bot when this happens. The test results in the form of a warning message received by the admin are shown in Fig. 6.

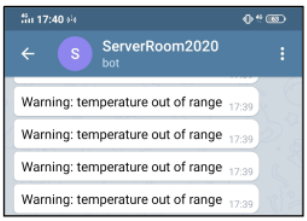

Fig. 6 Warning messages regarding temperature

The next test is testing when an intruder is detected entering from the front door. This means that an intruder breaks the front door to get into the server room. The distance read will automatically decrease. Furthermore, the Raspberry Pi will activate a warning buzzer and send a message to the telegram bot. Warning messages are sent in the form of text messages, photos, and videos. The message given by the system when this happens is shown in Fig.7.

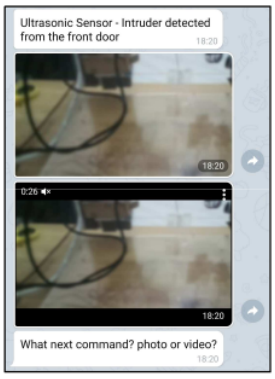

Fig. 7 Warning message regarding intruder from the front door

The next test is testing related to the PIR_1 sensor and PIR_2 sensor. This test aims to detect intruders who force their way through the roof or damage the back of the server room. When this happens, the intruder's movement will be detected by one or both of the PIR sensors. The message received by the user when an intruder is detected by the PIR_1 sensor or the PIR_2 sensor is shown in Fig. 8.

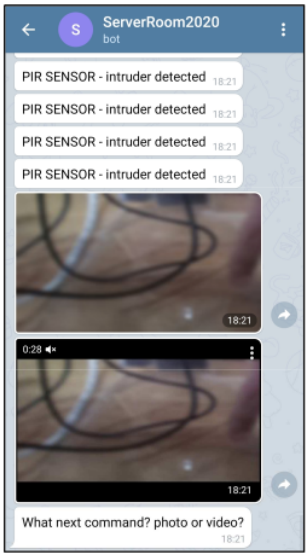

Fig. 8 Warning message regarding intruder detected with PIR sensor

Messages received by users are warning messages in the form of text, photos, and videos. After finishing sending all messages, the bot will wait for further instructions from the user. Commands can be in the form of commands to record and resend videos or take photos. All these tests were successful and followed the objectives and program scenarios. Based on all test results, this tool has fulfilled the objective where this tool can monitor the condition and security of the server room by utilizing IoT technology, while in other studies, it only focuses on temperature and humidity [5], [6], [8], and other studies only focus on door locks only [7].

IV. CONCLUSION

The server room can be monitored for its condition and security by implementing IoT and utilizing the telegram API. This tool has been running well, with a success percentage of 100% following the tests carried out. With this tool, users can also maintain access to the server room and will immediately receive a warning via Telegram bot when something goes wrong in the server room. It may be possible to add face detection features and the number of people in the server room for further research development.

REFERENCES

[1] J. P. A. Yaacoub, O. Salman, H. N. Noura, N. Kaaniche, A. Chehab, and M. Malli, "Cyber-physical systems security: Limitations, issues and future trends," *Microprocess. Microsyst.*, vol. 77, pp. 1–45, Sep. 2020, doi: 10.1016/j.micpro.2020.103201.

[2] I. Ghafir *et al.*, "Security threats to critical infrastructure: the human factor," *J. Supercomput.*, vol. 74, pp. 4986–5002, 2018, doi: 10.1007/s11227-018-2337-2.

[3] N. Scarpato, A. Pieroni, L. Di Nunzio, and F. Fallucchi, "E-health-IoT universe: A review," *Int. J. Adv. Sci. Eng. Inf. Technol.*, vol. 7, no. 6, pp. 2328–2336, 2017, doi: 10.18517/ijaseit.7.6.4467.

[4] Menteri Hukum Dan Hak Asasi Manusia, "Pedoman Penyelenggaraan Pusat Data Dan Ruang Server Di Lingkungan Kementerian Hukum Dan Hak Asasi Manusia Republik Indonesia," Indonesia, M.HH-01.TI.05.02, 2017. [Online]. Available: https://www.kemenkumham.go.id/attachments/article/1522/Kepmen Tentang Pedoman Penyelenggaraan Pusat Data dan Ruang Server - final.pdf.

[5] J. S. Jahlool, "Design and Simulation of Automatic Temperature Control and Alert System Based PIC16F887," *Int. J. Informatics Commun. Technol.*, vol. 6, no. 2, pp. 95–104, Aug. 2017, doi: 10.11591/ijict.v6i2.pp95-104.

[6] S. A. u. R. Omer and E. Muhammad, "Design of intelligent air conditioner controller using fuzzy logic," in *2017 International Conference on Innovations in Electrical Engineering and Computational Technologies (ICIEECT)*, 2017, pp. 1–5, doi: 10.1109/ICIEECT.2017.7916556.

[7] S. Niraj, K. Kamlesh, W. Pankaj, S. Danish, and G. Arvind, "Server Room Access Control Using Smartphone," *Int. J. Eng. Tech.*, vol. 4, no. 3, pp. 56–59, 2018, [Online]. Available: http://www.ijetjournal.org.

[8] M. Alvan, P. Utomo, A. Aziz, and B. Harjito, "Server Room Temperature & Humidity Monitoring Based on Internet of Thing (IoT)," in *Journal of Physics: Conference Series*, 2019, pp. 1–8, doi: 10.1088/1742-6596/1306/1/012030.

[9] J. Sobota, M. Goubej, J. Königsmarková, and M. Čech, "Raspberry Pi-based HIL simulators for control education," *IFAC-PapersOnLine*, vol. 52, no. 9, pp. 68–73, 2019, doi: https://doi.org/10.1016/j.ifacol.2019.08.126.

[10] V. Chauhan, M. Patel, S. Tanwar, S. Tyagi, and N. Kumar, "IoT Enabled real-Time urban transport management system," *Comput. Electr. Eng.*, vol. 86, pp. 1–18, 2020, doi: https://doi.org/10.1016/j.compeleceng.2020.106746.

[11] H. Hidayat, N. S. Hasibuan, and F. Wicaksono, "Design And Implementation Of Electronic Examination Device For Improving The Blind Students' Comfort," *J. Eng. Sci. Technol.*, vol. 16, no. 1, pp. 807–815, 2021.

[12] N. Bafdal and I. Ardiansah, "Application of Internet of Things in Smart Greenhouse Microclimate Management for Tomato Growth," *Int. J. Adv. Sci. Eng. Inf. Technol.*, vol. 11, no. 2, pp. 427–432, 2021, doi: 10.18517/ijaseit.11.2.13638.

[13] T. Zou *et al.*, "Raspberry promotes brown and beige adipocyte development in mice fed high-fat diet through activation of AMP-activated protein kinase (AMPK) α1," *J. Nutr. Biochem.*, vol. 55, pp. 157–164, 2018, doi: https://doi.org/10.1016/j.jnutbio.2018.02.005.

[14] M. T. A. Seman, M. N. Abdullah, and M. K. Ishak, "Monitoring Temperature, Humidity And Controlling System In Industrial Fixed Room Storage Based On IoT," *J. Eng. Sci. Technol.*, vol. 15, no. 6, pp. 3588–3600, 2020.

[15] I. Ardiansah, N. Bafdal, E. Suryadi, and A. Bono, "Greenhouse monitoring and automation using arduino: A review on precision farming and Internet of Things (IoT)," *Int. J. Adv. Sci. Eng. Inf. Technol.*, vol. 10, no. 2, pp. 703–709, 2020, doi: 10.18517/ijaseit.10.2.10249.

[16] M. Jacob and P. Mani, "A Reference Model For Testing Internet of Things Based Applications," *J. Eng. Sci. Technol.*, vol. 13, no. 8, pp. 2504–2519, 2018.

[17] S. Bin Saleh *et al.*, "Smart home security access system using field programmable gate arrays," *Indones. J. Electr. Eng. Comput. Sci.*, vol. 11, no. 1, pp. 152–160, 2018, doi: 10.11591/ijeecs.v11.i1.pp152-160.

[18] M. Murad, O. Bayat, and H. M. Marhoon, "Design and implementation of a smart home system with two levels of security based on IoT technology," *Indones. J. Electr. Eng. Comput. Sci.*, vol. 21, no. 1, pp. 546–557, 2021, doi: 10.11591/ijeecs.v21.i1.pp546-557.

[19] S. Nuanmeesri and L. Poomhiran, "Optimization Shortest One-Way Path for Energy Saving Auto Robot Collecting Floating Garbage using Fast Approximate Nearest Neighbor Search," *Int. J. Adv. Sci. Eng. Inf. Technol.*, vol. 11, no. 2, pp. 457–464, 2021, doi: 10.18517/ijaseit.11.2.13191.

[20] L. Benny and P. K. Soori, "Prototype of parking finder application for intelligent parking system," *Int. J. Adv. Sci. Eng. Inf. Technol.*, vol. 7, no. 4, pp. 1185–1190, 2017, doi: 10.18517/ijaseit.7.4.2326.

[21] O. Appiah, E. Quayson, and E. Opoku, "Ultrasonic sensor based traffic information acquisition system; a cheaper alternative for ITS application in developing countries," *Sci. African*, vol. 9, p. e00487, 2020, doi: https://doi.org/10.1016/j.sciaf.2020.e00487.

[22] X. Mouy *et al.*, "FishCam: A low-cost open source autonomous camera for aquatic research," *HardwareX*, vol. 8, p. e00110, 2020, doi: https://doi.org/10.1016/j.ohx.2020.e00110.

[23] K. Miikki *et al.*, "An open-source camera system for experimental measurements," *SoftwareX*, vol. 14, p. 100688, 2021, doi: https://doi.org/10.1016/j.softx.2021.100688.

[24] J. AnnRoseela and T. Godhavari, "Biometric and RFID based authentication system for exam paper leakages detection using IoT technology," *Indones. J. Electr. Eng. Comput. Sci.*, vol. 20, no. 3, pp. 1271–1277, 2020, doi: 10.11591/ijeecs.v20.i3.pp1271-1277.

[25] M. El Beqqal, M. Azizi, and J. L. Lanet, "Multimodal access control system combining RFID, fingerprint and facial recognition," *Indones. J. Electr. Eng. Comput. Sci.*, vol. 20, no. 1, pp. 405–413, 2020, doi: 10.11591/ijeecs.v20.i1.pp405-413.

[26] M. Vagaš, A. Galajdová, D. Šimšík, and D. Onofrejová, "Wireless data acquisition from automated workplaces based on RFID technology," *IFAC-PapersOnLine*, vol. 52, no. 27, pp. 299–304, 2019, doi: https://doi.org/10.1016/j.ifacol.2019.12.677.

[27] M. K. Kah Wen, N. binti Ahmad, and S. H. binti Ruslan, "Arduino based outing and attendance system for boarding school students," *Indones. J. Electr. Eng. Comput. Sci.*, vol. 20, no. 2, pp. 1053–1061, 2020, doi: 10.11591/ijeecs.v20.i2.pp1053-1061.

[28] N. Jannati, N. Nakhaee, V. Yazdi-Feyzabadi, and D. Tjondronegoro, "A cross-sectional online survey on patients' satisfaction using store-and-forward voice and text messaging teleconsultation service during the COVID-19 pandemic," *Int. J. Med. Inform.*, vol. 151, p. 104474, 2021, doi: https://doi.org/10.1016/j.ijmedinf.2021.104474.

[29] A. Dargahi Nobari, M. H. K. M. Sarraf, M. Neshati, and F. Erfanian Daneshvar, "Characteristics of viral messages on Telegram; The world's largest hybrid public and private messenger," *Expert Syst. Appl.*, vol. 168, p. 114303, 2021, doi: https://doi.org/10.1016/j.eswa.2020.114303.

IoT Implementation for Server Room Security Monitoring Using Telegram API

ORIGINALITY REPORT

7%	7%	2%	2%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	journal.uad.ac.id Internet Source	3%
2	insightsociety.org Internet Source	1%
3	www.journal-hyperion.ro Internet Source	1%
4	vdocuments.site Internet Source	1%
5	E W F Haikal. "Analysis of Movement Detection Applications in Pregnant Woman Using Body Mechanic and Sensors on Android Devices", IOP Conference Series: Materials Science and Engineering, 2019 Publication	1%

Exclude quotes	On	Exclude matches	< 1%
Exclude bibliography	On		