

Design and Implementation of a Multi-Sensor IoT System for Smart Laboratory Monitoring and Access Control Using ESP32

Abdelsalam Melad Faitouri ^{*1}, Najat Faraj Eleshebi ², Mofeda Abdelrazik Abdelwakil Madi ³

¹ Computer Network Department, College of Computer Technology, Benghazi, Libya

^{2,3} Software Engineering Department, College of Computer Technology, Benghazi, Libya

تصميم وتنفيذ نظام إنترنت الأشياء متعدد المستشعرات للمراقبة الذكية للمختبرات والتحكم في الوصول باستخدام ESP32

عبد السلام ميلاد الفيتوري ^{1*}، نجات فرج العشيبى ²، مفيدة عبدالرازق عبدالوكيل ماضي ³

¹ قسم شبكات الحاسوب، كلية تقنيات الحاسوب، بنغازي، ليبيا
^{2,3} قسم هندسة البرمجيات، كلية تقنيات الحاسوب، بنغازي، ليبيا

*Corresponding author: Abdelsalamzayani@gmail.com

Received: July 16, 2026

Accepted: September 28, 2026

Published: October 10, 2026

Abstract:

This paper presents the design and practical implementation of an IoT-based smart laboratory monitoring and access-control system using an ESP32 microcontroller. The proposed system addresses limitations of conventional laboratory monitoring by integrating environmental monitoring, human presence detection, and access control within a unified platform. The implemented system consists of a DHT22 sensor for temperature and humidity measurement, an MQ-2 sensor for smoke and gas detection, a PIR motion sensor for human presence detection, and an RC522 RFID module for automated access control. The ESP32 serves as the central processing and communication unit, collecting data from the connected sensors, processing the measurements, and supporting wireless communication through its integrated Wi-Fi capability. The study adopts a design-based Cisco PPDIOO framework to structure the implementation process. Experimental results confirm the feasibility and reliability of the deployed system in detecting environmental hazards and managing laboratory safety in real time.

Keywords: ESP32, Internet of Things (IoT), Laboratory monitoring, Real-time monitoring, Smart laboratory.

المخلص

تقدم هذه الورقة تصميمًا وتنفيذًا عمليًا لنظام ذكي قائم على إنترنت الأشياء (IoT) لمراقبة المختبر والتحكم في الوصول إليه باستخدام المتحكم الدقيق ESP32. يعالج النظام المقترح أوجه القصور في أساليب المراقبة التقليدية للمختبرات من خلال دمج مراقبة البيئة، والكشف عن وجود الأشخاص، والتحكم في الوصول ضمن منصة موحدة. ويتكون النظام الذي تم تنفيذه من مستشعر DHT22 لقياس درجة الحرارة والرطوبة، ومستشعر MQ-2 للكشف عن الدخان والغازات، ومستشعر الحركة PIR للكشف عن وجود الأشخاص، ووحدة RFID من نوع RC522 للتحكم الآلي في الوصول. ويعمل ESP32 كوحدة مركزية للمعالجة والاتصال، حيث يقوم بجمع البيانات من المستشعرات المتصلة ومعالجة المدخلات ودعم الاتصال اللاسلكي من خلال تقنية Wi-Fi المدمجة. تبنت الدراسة منهجية معتمدة على التصميم وفق إطار PPDIOO، وتؤكد النتائج التجريبية كفاءة النظام وموثوقيته في الكشف المبكر عن المخاطر البيئية وإدارة سلامة المختبر بالوقت الحقيقي.

الكلمات المفتاحية: ESP32، إنترنت الأشياء (IoT)، مراقبة المختبر، المراقبة في الوقت الفعلي، المختبر الذكي.

Introduction

The rapid development of Internet of Things (IoT) technology has significantly changed the way institutions monitor and manage their educational laboratories. IoT enables real-time monitoring and intelligent decision-making in different environments through the integration of smart sensors, wireless communication technologies,

and cloud services (Alshuhail,2025). It also supports the automation of many processes and improves the efficiency of decision-making. University laboratories are among the environments that require high levels of monitoring and management because they contain valuable equipment and devices. In addition, electrical power sources and other operating conditions may create real risks if they are not continuously monitored. Protecting laboratories from unauthorized access is another challenge for university supervisory units. Many laboratories still depend on traditional monitoring and supervision methods. These methods may suffer from slow incident detection, difficulty obtaining real-time information, and the absence of a unified platform for laboratory monitoring. With the emergence of low-cost microcontrollers such as the ESP32, which provides integrated Wi-Fi and Bluetooth connectivity and suitable processing capabilities (Bakare,2026), it has become possible to develop practical IoT-based monitoring systems. Previous studies have demonstrated the feasibility of using ESP32-based IoT systems for smart laboratory monitoring and access control. (Hasib et. al. 2026) proposed an ESP32-based IoT framework that combined RFID access control with multi-sensor environmental safety monitoring. This study demonstrate the effectiveness of integrating sensing, monitoring, and access control functions within a unified IoT platform. However, the proposed system extends these approaches by integrating environmental monitoring, human presence detection, smoke and gas detection, and RFID-based access control into a single laboratory monitoring architecture.

Problem Statement

Many university laboratories rely on manual monitoring or separate systems for controlling access and monitoring environmental conditions such as temperature and humidity. This can lead to slow detection and analysis of potential risks and limited awareness of laboratory conditions in real time. A unified monitoring platform is therefore needed to integrate several monitoring functions and provide timely information to responsible personnel.

Research Objectives

Real-Time Monitoring: Monitor laboratory conditions and identify abnormal events in real time.

Automated Entrance: Apply RFID technology for secure and automated user identification and access control.

Data Display: Display sensor outputs collected and processed by the ESP32 on a local monitoring interface.

User-Friendly Monitoring: Provide a clear interface that enables supervisors to access laboratory information easily.

Immediate Alerts: Support direct notification of responsible personnel when abnormal or emergency conditions are detected.

Methodology

Several methodologies can be used for network and system design, and the appropriate choice depends on project size, requirements, security needs, and operational objectives. There are many approaches such as: Network Lifecycle Methodology, Hierarchical Network Design, Modular Network Design, Bottom-Up Approach, Top-Down Approach, Middle-Out (Meet-in-the-Middle) Approach, and CiscoPPDIOO(Naim,2023). The researchers adopted the Cisco PPDIOO methodology as the framework for the proposed system because it provides clearly defined stages covering the network and system life cycle from preparation and planning through optimization. In this study, the methodology is used as an organized framework for translating laboratory monitoring requirements into a practical IoT implementation (Purwanto,2021).

PPDIOO Phases

Prepare and Plan: Identify the laboratory monitoring requirements and determine the hardware and software needed.

Design: Develop the detailed architecture, component distribution, connections, and data flow.

Implement: Physically connect the hardware and develop and upload the required software according to the design.

Operate: Power and operate the system, monitor sensor readings, and manage the system during continuous operation.

Optimize: Analyze the implementation and identify improvements to measurement accuracy, response, stability, and reliability.

System Design

1. Preparation and Plan

During this phase, the researchers work to address the problem of the lack of real-time laboratory monitoring by adopting Internet of Things (IoT) technology and identifying the necessary requirements. The required functions include monitoring temperature and humidity, detecting smoke and gas, detecting human presence, measuring electrical current, controlling access through RFID, and displaying monitoring information.

Table.1. Hardware Requirements

Device	Function
ESP32	Main Processing and central control unit
RFID-RC522	Automated user identification and access control
DHT22	Temperature and humidity measurement
MQ-2	Smoke and combustible gas detection
PIR Motion Sensor	Human presence detection
Wi-Fi	Wireless data transmission and telemetry
LCD / Monitoring Dashboard	Display of real-time monitoring data and safety alert

The main controller unit, the ESP32, uses the Tensilica Xtensa LX6 processor. This microprocessor has a 32-bit architecture and incorporates several technologies, including Wi-Fi, Bluetooth and Multiple low-power modes.



Figure1: ESP3 Pinout and Board Layout (Espressif systems,2026)

1.2 Software Requirements

Arduino IDE is used for writing and uploading the firmware to the ESP32. The board is connected to the computer through a suitable USB cable, the corresponding port is selected, and the compiled program is uploaded to the ESP32 flash memory.

2. Design Phase

The design phase is considered one of the most important stages as it involves convert the identified requirements into a detailed technical design. It defines the system architecture, hardware distribution, sensor connections, processing role of the ESP32, and the flow of monitoring information. The ESP32 operates as the central controller, while the integrated Wi-Fi capability provides the communication basis for remote monitoring (Cisco system, n.d.).

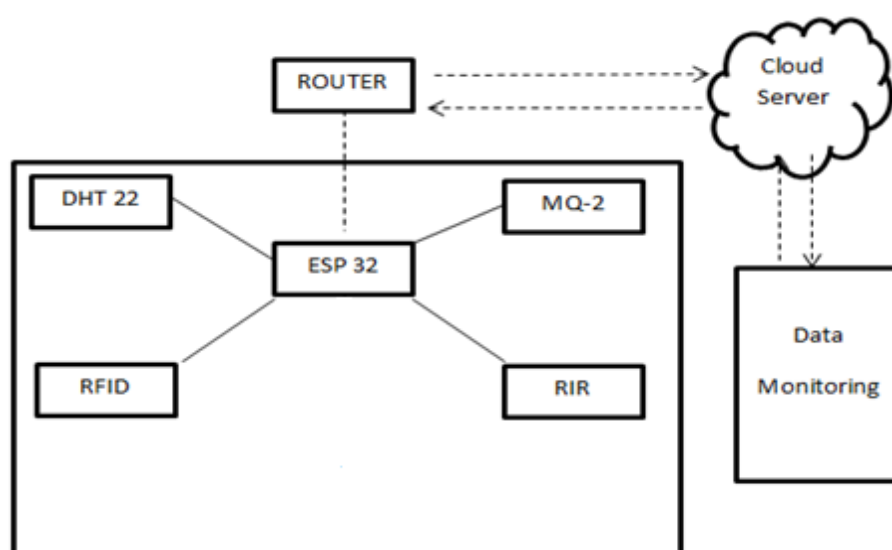


Figure 2: diagram of the proposed system

3. Implementation Phase

In this phase, the theoretical design is transformed into a functional system that can operate in a real-world environment. The system was implemented using an ESP32 microcontroller connected to several sensors and input/output devices to perform the required monitoring and control functions (Cisco system, n.d.).

All sensors were connected to the ESP32 through the appropriate GPIO pins, taking into consideration the compatibility of the electrical voltage levels. The physical connections were implemented based on the information provided in the following tables (Expressf system,2026).

Table.2. RC522-ESP32 Pin Connections

RC522	ESP32
SDA	GPIO 5
SCK	GPIO 18
MOSI	GPIO 23
MISO	GPIO 19
RST	GPIO 27
VCC	3.3 V
GND	GND

Table.3. DHT22-ESP32 Pin Connections

DHT22	ESP32
VCC	3.3 V
DATA	GPIO 4
GND	GND
10 kΩ resistor Between DATA	and VCC

Table.4. MQ-2-ESP32 Pin Connections

MQ-2	ESP32
VCC	5 V
GND	GND
AO	GPIO 34

Table.5. IR Sensor-ESP32 Pin Connections

IR Sensor	ESP32
VCC	5 V
GND	GND
OUT	GPIO 26

Based on the pin connections detailed in Tables(2) through(5), the overall hardware schematic was constructed to serve as the physical baseline for data acquisition and performance evaluation.

Following the physical integration, the firmware was implemented to define sensor communication protocols and establish critical hazard threshold boundaries, distinguishing nominal operational state from emergency conditions.

Table.6. outlines key code segments of the implement firmware.

Code Snippet	Function
<pre>#define TEMP_THRESH 35.0 #define GAS_THRESH 1500</pre>	System configuration and threshold initialization
<pre>void loop(){ float temp = dht.readTemperature(); float hum = dht.readHumidity(); int gasVal = analogRead(MQ2_PIN); int irState = digitalRead(IR_PIN);</pre>	Main execution loop
<pre>serail.print("[DATA] Temp: ");serial.print(temp); Serial.print("C "); Serial.print("Hum: "); Serial.print(hum); serial.print("% "); Serial.print("Gas: "); Serail.print(gasVal); Serial.print(" "); Serial.print("Presence: "); serial.println(irState == LOW ? "YES" : "NO");</pre>	Data telemetry and serial communication forming
<pre>if (temp > TEMP_THRESH gasVal > GAS_THRESH){ Serial.println(">> [ALERT] EMERGENCY!! <<"); if (rfid.PICC_IsNewCardPresent() && rfid, PICC_readCardSerial()){ Serial.println(">> [RFID] Access Granted. <<"); rfid.PICC_HaltA();} }</pre>	Conditional hazard evaluation and threshold design logic
<pre>delay(2000);</pre>	Sampling interval and execution delay

4. Operation Phase

Powering the ESP32 microcontroller initializes the primary CPU responsible for controlling all connected hardware modules and managing system execution. Upon system startup, all integrated components are initialized, including the MQ-2 sensor for smoke and gas detection, The DHT22 sensor for temperature and humidity measurement, The PIR motion sensor for human presence detection, and the RC522 RFID module for access control. Safe threshold boundaries for temperature, humidity, air quality, and power consumption are pre-programmed and stored in the non-volatile system memory.

After hardware setup, the operational firmware is prepared and uploaded using the Arduino IDE (Macheso,2021). The development environment is configured by installing the necessary board manager packages to support ESP32 microcontrollers. The ESP32 board is then connected to the workstation via a suitable USB cable, the corresponding serial port is selected, and the source code is compiled and flashed to the onboard memory (Formanek,2025).

After the program is uploaded to the ESP32 microcontroller, the microcontroller will automatically restart and begins executing the instructions according to the stored code. The system will then enter a fully operational state, continuously monitoring the environmental variables. During this state, all required data are read at a programmatically defined time interval and then compared with the threshold values specifies in the code.

These results are then displayed on the supervisors' display, which may be Android phone or LCD screen (Wivanius,2025), allowing the appropriate decision to be made while the measurement and processing operations continue repeatedly. this ensures an immediate response to any changes in the environmental conditions.

5. Optimization Phase

After completing the implementation and testing of the system, the optimization phase is carried out to improve system performance and enhance operational efficiency. This phase includes analyzing the results and identifying weaknesses so that the necessary modifications can be made to both the software and hardware components to improve measurements accuracy, response speed, and system stability. It also includes periodic maintenance, such as updating the software when necessary, recalibrating sensor parameters, and replacing components whose deterioration may affect the sufficiency of the system, thereby ensuring reliable operation and achieving its intended objectives (Cisco system, n.d.).

Results and Discussion

The proposed system was physically implemented using an ESP32 microcontroller and the selected sensing and identification modules. The implementation integrated the RC522 RFID module (Roza,2024), DHT22 temperature and humidity sensor (Aosong Electronics, Co., Ltd., n.d.), MQ-2 gas and smoke sensor (Winarno,2023), PIR presence sensor (Rajaram,2023). The practical implementation demonstrated the functional integration of the different components with the ESP32. The microcontroller was able to receive data from the connected sensors and process the collected information according to the programmed monitoring logic. The DHT22 was used for temperature and humidity monitoring, the MQ-2 for smoke and gas detection, the IR sensor for human presence detection, and the RC522 for RFID-based access control. The continuous real-time execution of the proposed ESP32-based multi-sensor monitoring system was experimentally validated across a simulated operational period of 26 seconds. The empirical dataset, summarized in table (8), captures the temporal dynamic transitions between baseline standby conditions, normal user authorization events, environmental anomaly surges, and recovery phases. To evaluate the fast response time of the sensors and system alerts, an accelerated stress test was performed under controlled laboratory conditions by intentionally applying localized heat and smoke near the sensors, followed by immediate forced ventilation.

Table .8. Simulated environmental and security monitoring data over time.

Time(s)	Temp(C°)	Humidity(%)	Gas Level(ADC)	PIR Motion	RFID Status/Event	
00:02	23.5	48.2	410	NO	None	NP
00:04	23.6	48.0	415	NO	None	NP
00:06	23.8	47.9	420	YES	UID:4B 8F 12 A0	User Entry
00:08	24.1	47.5	435	YES	None	OL
00:10	24.5	47.1	450	YES	None	OL
00:12	25.2	46.8	510	YES	None	RM
00:14	27.8	45.0	820	YES	None	TS
00:16	36.2	41.2	1680	YES	None	ALERT
00:18	38.5	39.5	2150	YES	None	ALERT
00:20	37.1	40.8	1820	YES	None	ALERT
00:22	31.0	43.5	980	YES	None	Cooling/VP
00:24	26.4	46.2	540	NO	None	SN
00:26	24.2	47.8	425	NO	None	NP

NP=Normal Operation, OL=Occupied Lab, RM=Routing Monitoring, TS=Temperature Surge, VP=Ventilation Phase, SN=System Normalizing

As illustrated in figure (3), the baseline environmental parameters remained stable during the initial phase ($t = 0.00s$ to $12.31s$), with the ambient temperature averaging 24.12 C° and combustible gas readings hovering around 430 ADC units. At $t = 06.19s$, a successful RFID authentication event (UID: 4B 8F 12 A0) synchronized with a high digital pulse on the IR presence pin (Presence = YES), confirming system responsiveness during routing operational entries without inducing hardware-level processing latency. A controlled hazard scenario was induced at $t = 14.07s$ by exposing the sensors to a localized heat and gas source, causing an immediate threshold trigger that validated the quick response of the ESP32 system. By $t = 16.18\text{ s}$, both primary monitoring variables crossed their designated safety boundaries: ambient temperature reached 36.2 c° (exceeding the 35.0 c° threshold), and gas concentration spiked to 1678 ADC (exceeding the 1500 ADC threshold). The system peaked at $t = 18.22s$ (38.5 c° and 2145 ADC), triggering an immediate emergency alarm state. This concurrent breach demonstrates the dual-threshold hazard intervention scheme programmed into the ESP32 firmware.



Figure 3: Real-Time temperature profile and safety threshold(35c°)

Additionally, Figure(4) illustrate the dynamic response of the MQ-2 gas sensor and its progression toward the safety boundary. During the initial baseline period ($t = 0s$ to $12s$), the gas concentration remained stable at nominal ambient levels, averaging 430 ADC units. Following the simulated hazard events at $t = 14s$, the gas concentration experienced a steep increase, reaching 1680 ADC at $t = 16s$, and crossing the pre-programmed critical threshold of 2150 ADC. The concentration peaked at 2150 ADC ($t = 18s$) before the actuation of the automated ventilation system effectively reduced the gas levels back to safe operational baseline limits (425 ADC) by $t = 26s$.

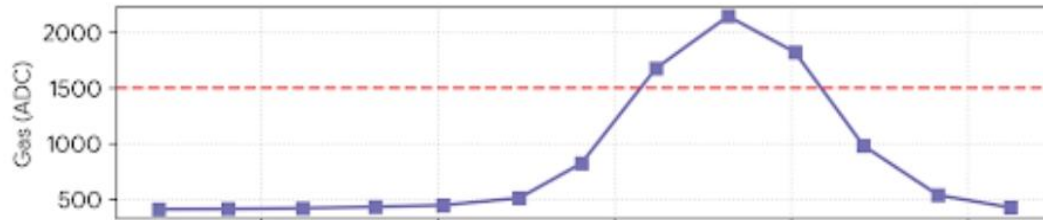


Figure 4: Gas concentration dynamics and critical limit (1500 ADC)

Finally, Figure(5) illustrates the digital logic response of the PIR Motion detection for automated laboratory occupancy tracking. The spatial occupancy state shifted from Vacant(0) to Occupied(1) at $t = 6s$, corresponding to the successful RFID access event logged in Table(8). The system maintained this occupied state through the monitoring sequence until $t = 24s$, when the human subject cleared the laboratory, return the state to Vacant(0). This real-time transition validates the capability of the ESP32 firmware continuous occupancy tracking simultaneously with environmental hazard polling.

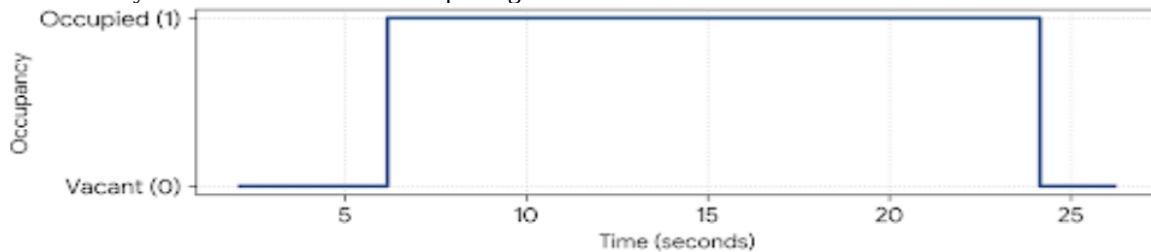


Figure.5: IR-Based laboratory occupancy logic state

Conclusion

This paper proposes the design of an environmental monitoring system for university laboratories using the ESP32 microcontroller. The research was conducted within the framework of the PPDIOO methodology, which includes the phases of Prepare, Plan, Design, Implementation, Operation, and optimize. The code was uploaded to the microcontroller and the system was evaluated during the implementation and operation phase, with the system's stability verified under operational testing. Meanwhile, the optimization phase focused on improving efficiency, addressing response latency, and enhancing the reliability of measurements. Future enhancements will focus on integrating edge-AI algorithms directly onto the ESP32 for predictive anomaly detection, replacing local Wi-Fi dependency with LoRA-based communication for long-range connectivity, and incorporating solar backup power to ensure uninterrupted operation during grid outages.

Compliance with ethical standards

Disclosure of conflict of interest

The authors declare that they have no conflict of interest.

References

- Alshuhail A., A. A. (2025). Machine edge-aware IoT framework for real-time health monitoring : Sensor fusion and AI-driven emergency response in decentralized networks. *Alexandria Engineering Journal*.
- Aosong Electronics Co., Ltd. (n.d.). Digital-output relative humidity & temperature sensor/module.
- Arya, A. (2023). Anyone can code: *Algorithmic thinking*. Ali Arya.
- Bakare, M. S., & Abubaker, K. (2026). IoT-based indoor environmental monitoring system using multi-parameter sensing and ESP32-WROOM integration. *Discover Electronics*,3,.
- Cisco System.(n.d.). *Cisco Unified Communications IP Telephony technical information site*. <https://www.cisco.com/cisco/web/docs/iam/unified/ipt601/index.html>
- Espressif Systems. (2026). *ESP32-H2-MINI-1 & ESP32-H2-MINI-1U datasheet* (Version 1.6).
- Formanek, L., Kubascik, M., Karpis, O., & Kolok, P. (2025). Advanced system for remote updates on ESP32-based devices using over-the-air update technology. *computers*, 14(12), 531.

- Hasib, A., Akib, A. S. M. A. S., Ankur, N. D., & Giri, A. (2026). *Dual-modality IoT framework for integrated access control and environmental safety monitoring with real-time cloud analytics*. *arXiv*.
- Macheso, P., Chisale, S. W., Daka, C., Dzupire, N. C., Mlatho, J., & Mukanyirigira, D. (2021). *design of standalone asynchronous ESP32 web-server for temperature and humidity monitoring*. In 2021 7th International Conference on Advanced Computing and Communication Systems(ICACCS) (pp. 635-638). IEEE.
- Naim, F., Saedudin, R. R., & hediyanto, U. Y. K. S. (2022). Analysis of wireless and cable network quality-of-service performance at Telkom University Landmark Tower using Network Development Life Cycle (NDLC) method. *JIPI (Jurnal Ilmiah Penelitian dan Pembelajaran Infomatika)*, 7(4), 1033-1044
- Purwanto, A., & Soewito, B. (2021). Optimization problem of computer network using PPDIOO. *ICIC Express Letters*, 15(7), 769-777
- Rajaram, V., Kang, S., Calisgan, S. D., Risso, A., Qian, Z., & Rinaldi, M. (2021). Miniaturized zero-power infrared sensor for persistent and large-scale human presence monitoring at borders. In 2021 *IEEE International Symposium on Technologies for Homeland Security (HST)* (pp. 1-7) IEEE.
- Roberts, N. (2019) The standard written algorithm for addition: Whether, when and how to teach it. *Pythagoras*, 40(1), a487.
- Roza, Y., Musliadi, K. H., & Pernando, Y. (2024). Presence prototype model design using RFID RC522. *Scientific Literacy Innovation and Technology journal*, 1(2), 26-31.
- Winarno, A., Widodo, W., & Darmawan, R. H. (2023). Smoke detection design using MQ-2 sensor and exhaust fan based on microcontroller. *BEST: journal of Applied Electrical, Science and Technology*, 5(1), 7-10.
- Wivanius, N., Sihombing, W. I., & Kamsyah, D. (2025). Design and prototype of an IoT-based temperature and humidity monitoring system with real-time data and automated alerts. *Jurnal Teknologi dan Riset Terapan*, 7(2).
- Zingaro, D. (2024). *Algorithmic thinking: Learn algorithms to level up your coding skills* (2nd ed.)

Disclaimer/Publisher's Note: The statements, opinions, and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of **AJAPAS** and/or the editor(s). **AJAPAS** and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions, or products referred to in the content.