Journal of Artificial Intelligence and Engineering Applications

Website: https://ioinformatic.org/

15th October 2025. Vol. 5. No. 1; e-ISSN: 2808-4519

Development of a Dual Biometric Authentication System Based on IoT with Facial Recognition and Fingerprint for Safe Security

Rangga Sudrajad[1]*, Achmad Fauzi[2], Milli Alfhi Syari[3]

[1,2,3] *STMIK Kaputama Binjai*
ranggasudrajad11@gmail.com[1], fauzyrivai88@gmail.com[2], milli.alfhisyari@yahoo.co.id[3]*

Abstract

Protecting valuable assets requires a security system that adapts to modern challenges, where physical keys or numerical codes on conventional safes are vulnerable to loss, duplication, and breaking. This research develops an Internet of Things (IoT)-based dual biometric authentication system with face recognition using ESP32-CAM and AS608 fingerprint verification performed sequentially, granting access only if both authentication stages are successful. The NodeMCU ESP32 is used as the main controller, integrating the authentication process with the solenoid lock and buzzer, and utilizing Firebase Realtime Database for real-time monitoring of status and access history through an Android application. Unlike previous research, which only used face recognition with manual verification via Telegram, this system is fully automated with multi-layer authentication, making it more secure and efficient. The prototyping method is used to design, program, and test the system, with testing results showing 100% success in opening the safe only for registered users, making this system more reliable than single authentication.

Keywords: *Biometric authentication, ESP32-CAM, Fingerprint sensor, Internet of Things*

1. Introduction

Physical protection of valuable assets such as important documents, cash, and other valuable items is a fundamental need in an increasingly complex digital era. Conventional safes that use physical keys or numeric codes remain the primary choice for maintaining security; however, these methods have revealed various weaknesses. Physical keys can be lost or duplicated, while combination codes are prone to being forgotten or cracked.

The urgency of developing technology-based security systems is increasing, considering the rise in burglary cases and the demand for efficiency and practicality in access. Biometric-based security systems, such as facial recognition and fingerprints, are starting to be used in various devices because they offer a higher level of security and are difficult to forge. However, several previous studies have identified limitations in the biometric-based vault security systems that have been developed. For example, systems that rely solely on one biometric authentication method, such as facial recognition or fingerprints, tend to be more vulnerable. Facial recognition is still done manually through Telegram, which means users have to verify their faces directly through the messaging app, so the system is not yet fully automated. Additionally, a system that only uses one of the two authentications ("or") can create a security gap because if one method is hacked, access to the vault can still be gained [14].

Based on this gap, this research proposes the development of a vault security system with an IoT-based dual biometric authentication approach, which combines facial verification using ESP32-CAM and an AS608 fingerprint sensor in a sequential process. This means that facial verification must be successful first, and only then will the system allow fingerprint scanning. Only if both authentication stages are successful will the solenoid lock be activated to open the safe. Additionally, this system is supported by real-time communication through Firebase, allowing the owner to monitor the status and access history directly through the Android application. This approach not only enhances security through multi-layer authentication but also provides efficiency and convenience in its use. Considering the need for advanced security and the shortcomings of previous systems, this research becomes relevant and important to develop. Therefore, this research is titled: "Development of an IoT-Based Dual Biometric Authentication System with Facial Recognition and Fingerprint for Vault Security."

2. Theoretical Framework

a. Safebox

A safe is a box-shaped cabinet made of fire-resistant iron, specifically designed to store valuable items such as money, important documents, and jewelry. Its function is to protect valuables from the risk of fire as well as theft or forced entry. This safe usually has a

locking system that uses a combination of keys or digital keys, and is available in various sizes, ranging from small and portable to large and wall-mounted or even room-sized [15].

b. Internet of Things (IoT)

The Internet of Things (IoT) is essentially a network consisting of physical devices interconnected thru the internet, enabling communication and data exchange between these devices [4].

c. Face Detection

Face detection is a technology used to identify and localize human faces in images or videos, working by analyzing facial features such as eyes, nose, and mouth using algorithms like Viola-Jones, CNN, or HOG. In the context of security, this technology is often integrated with other systems, such as fingerprint recognition, to provide dual authentication. The main components of face detection include camera sensors, microcontrollers (like Arduino), and face recognition modules (like ESP32CAM), which work together to process facial images in real-time [12].

d. Microcontroller

A microcontroller is a small electronic circuit that serves as the main controller in a system. This device is equipped with a processing unit (CPU), memory, a timer, as well as serial and parallel communication interfaces, which enable it to manage the operation of various electronic components in an integrated manner [9].

e. NodeMcu ESP32

NodeMCU ESP32 serves as the main controller for the fingerprint-based safe security system, with the ability to send notifications via the Telegram application. Its advantages lie in its small size, low power consumption, and ability to connect with various sensors and external devices, such as fingerprint sensors, relays, and LCDs [6].

f. ESP32-CAM

The ESP32-CAM is a microcontroller module specifically designed for Internet of Things (IoT) projects that require camera features. This module is equipped with various facilities such as Bluetooth, WiFi, a microSD slot, and an OV2640 camera that can operate independently. Unlike the ESP32-Wroom module, the ESP32-CAM has a more limited number of I/O pins, requiring a USB TTL or computer USB port for programming.

g. AS608 Fingerprint Sensor

In this research, the AS608 sensor module is used, which has digital signal processing capabilities. This module performs several main functions, including image processing, data calculation, feature recognition, and finally matching with previously registered fingerprint data [8].

h. Solenoid Door Lock

A solenoid is a type of actuator that functions to produce linear motion. This device can operate electromagnetically (AC/DC) or hydraulically. Although they are of different types, all solenoids operate on a similar basic principle. When voltage is applied, the solenoid will produce a linear push [2].

i. Jumper Cable

A jumper cable is an electrical cable used to connect various components on a breadboard without the need for soldering. Typically, these cables are equipped with connectors or pins at both ends. The connector that functions to pierce is known as a male connector, while the connector that receives the piercing is called a female connector.

j. Relay

The relay plays a crucial role in electronic and electrical circuit systems, especially for activating high-current devices without needing to be directly connected to low-current control devices [7].

k. Buck Converter

The buck converter is capable of maintaining output voltage stability according to the given reference, making it highly ideal for use in microcontroller and sensor-based systems such as in this dual biometric authentication project [1].

l. Adapter

An adapter is an electronic circuit that functions to convert high AC (alternating current) voltage into lower DC (direct current) voltage [2].

m. Breadboard

Breadboard, one of the fundamental components in the world of electronics, functions as a prototyping platform that allows designers and technicians to build temporary electronic circuits without the need for soldering. This tool becomes a critical stage in the early development of electronic products thanks to its high flexibility and efficiency. With a breadboard, users can quickly test concepts, modify designs, and validate circuit functionality before moving on to project implementation [7].

n. Buzzer

A buzzer is generally used as an indicator or signal that a process has been completed, or to provide a warning if an error occurs in the device, similar to an alarm function [14].

o. Firebase

In the implementation of the safe security system, Firebase is used as a medium for online data storage and management, connecting various components such as facial recognition, fingerprint sensors for biometric authentication, and microcontrollers as the system's control center. Every activity, from identity verification, commands to open or lock the safe, to detecting unauthorized access attempts, can be recorded and monitored through Firebase [16].

p. Flowchart

A flowchart is a visual representation of an algorithm that uses graphic symbols to depict the sequence of steps in solving a problem. Each symbol in a flowchart has a specific meaning, such as process, decision, input/output, and connectors between steps. Flowcharts help facilitate problem analysis, identify errors, and serve as an effective communication medium among team members in program development [13].

3. Methods

At the design stage of this research, the prototyping method is employed to systematically and visually illustrate the sequence of research steps involved in developing an Internet of Things (IoT)-based secure system with dual biometric authentication, utilizing both facial recognition and fingerprint sensors. The visualization of the research method flow can be seen in the Research Methodology Flowchart displayed in Figure III.1.

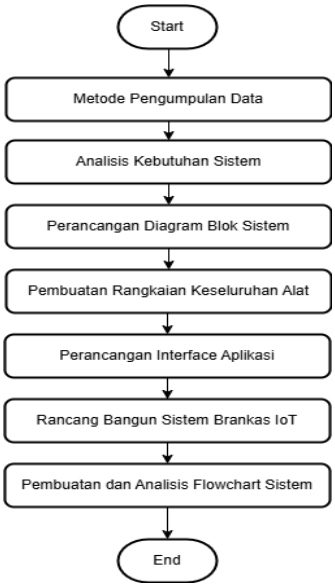


Figure III.1 Research Methodology Flowchart

Data collection methods are an important stage in research because they serve as the foundation for obtaining accurate, relevant, and accountable information to support analysis and conclusion formulation. In this study, data collection is carried out systematically and structured to support the design and evaluation process of an Internet of Things (IoT)-based safe system with biometric authentication in the form of facial and fingerprint recognition.

3.1 System Requirements Analysis

System requirement analysis for the design of an Internet of Things (IoT) based Safe Security System using Face and Fingerprint Detection includes two main components, namely software and hardware.

a. The software required for this research includes:
   1) Arduino IDE
   2) Firebase
   3) Fritzing

b. The hardware required for this research includes:
   1) NodeMCU ESP32
   2) ESP32-Cam
   3) Fingerprint Sensor
   4) Solenoid Door Lock
   5) Jumper Wires
   6) Adapter
   7) ESP32-CAM-MB
   8) Buck Converter
   9) Relay
   10) Breadboard
   11) Buzzer
   12) Smartphone
   13) Plywood and Chipboard
   14) Glue and Insulation

3.2 System Block Diagram

The system block diagram in this study is designed with the NodeMCU ESP32 as the main microcontroller that integrates and controls the operation of all components. The NodeMCU ESP32 receives power through a 12V adapter, which is stepped down using a buck converter to meet the device's requirements. The system employs two sequentially connected biometric authentication methods, namely facial recognition using the ESP32-CAM and fingerprint verification through the AS608 sensor. The system design block diagram can be seen in Figure III.2.

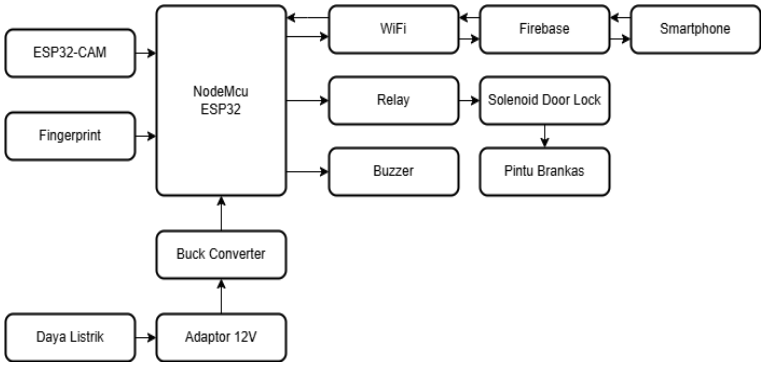


Figure III.2 System Block Diagram

3.3 Overall Equipment Arrangement

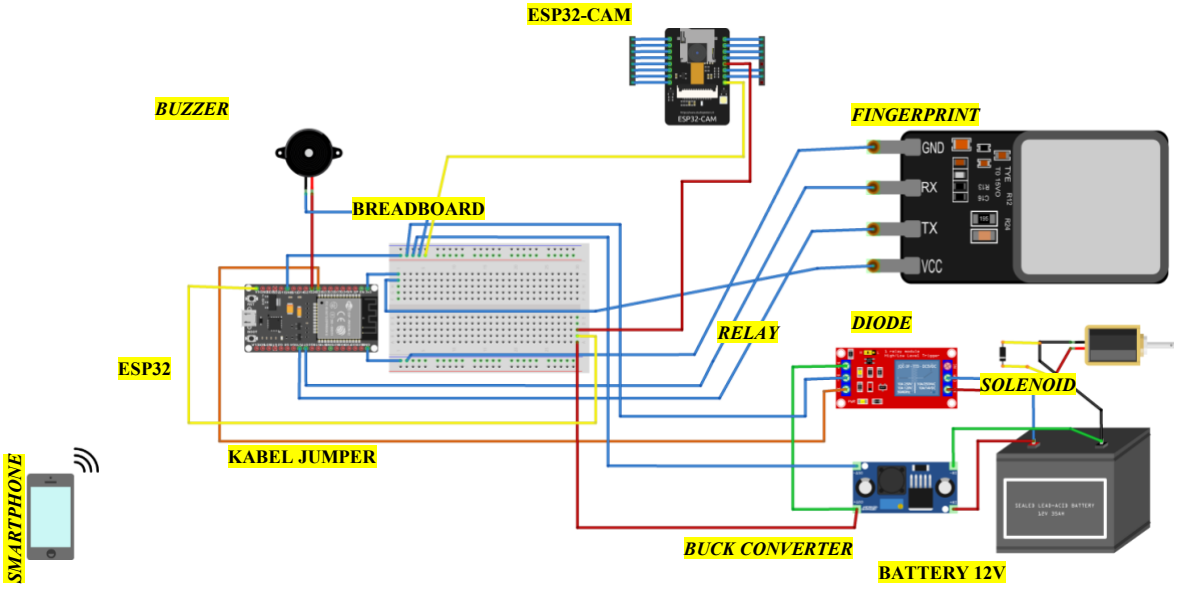


Figure III.3 Overall Circuit of the Device

3.4 Interface

The user interface (UI) of the Android application "My Brankas" is designed to be simple yet functional to facilitate users in operating the dual authentication-based vault security system. The main interface of the application consists of three main buttons: "Unlock," "Lock," and "View Access Log," each distinguished by different colors to indicate their functions (green for unlocking, red for locking, and blue for viewing the log). At the top of the buttons, there is a status notification area in the form of an information panel that displays messages such as "Waiting for verification from ESP32..." which is connected in real-time to Firebase as a cloud database. This design adopts a minimalist principle with a vertical layout and sufficient padding to provide space between elements, thereby enhancing user interaction comfort. Additionally, at the bottom of the screen, there is an access log panel that will display the history of successfully performed authentication activities, whether through facial recognition or fingerprints. This interface serves as the main bridge between the user and the vault control system, while also ensuring intuitive and responsive control over every action performed through the application. The appearance of the User Interface used can be seen in Figure III.4.

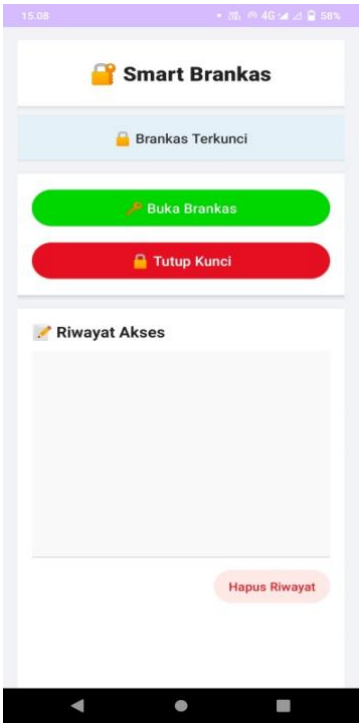


Figure III.4 User Interface Display

3.5 Development of a Dual Biometric Authentication System Based on IoT with Facial Recognition and Fingerprint for Safe Security

The process of creating this dual authentication system begins with the design of the circuit schematic and programming of the ESP32 and ESP32-CAM according to the specified authentication sequence. The system is developed so that the first stage, facial verification, is performed by the ESP32-CAM, and if successful, the system will activate the fingerprint sensor for the second stage of authentication. Next, if the fingerprint verification is valid, the ESP32 will activate the relay to unlock the solenoid. This system is connected to Firebase so that users can monitor the status of the safe in real-time through their smartphones. Testing is conducted to ensure that each stage of authentication runs smoothly and the device functions integratively.

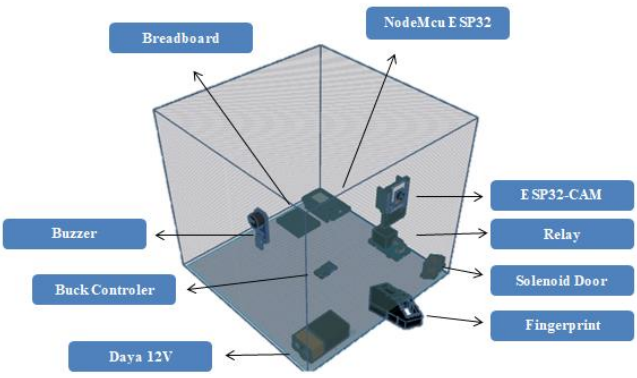


Figure III.5 Design and Construction of the Safe

3.6 Flowchart Network

The flowchart visually represents the workflow of the system, illustrating the sequence of processes and decisions in the developed Internet of Things (IoT)-based safe security system. In this system, two biometric devices, namely the ESP32-CAM for facial detection and a fingerprint sensor, are used sequentially as a dual authentication mechanism. The system's logic flow is designed so that the hardware and mobile application are interconnected through Firebase. This flowchart helps visualize the integration between components, starting from initialization, the identity verification process, granting access by the application, to the re-locking of the safe. The sequence flowchart can be seen in Figure III.6.

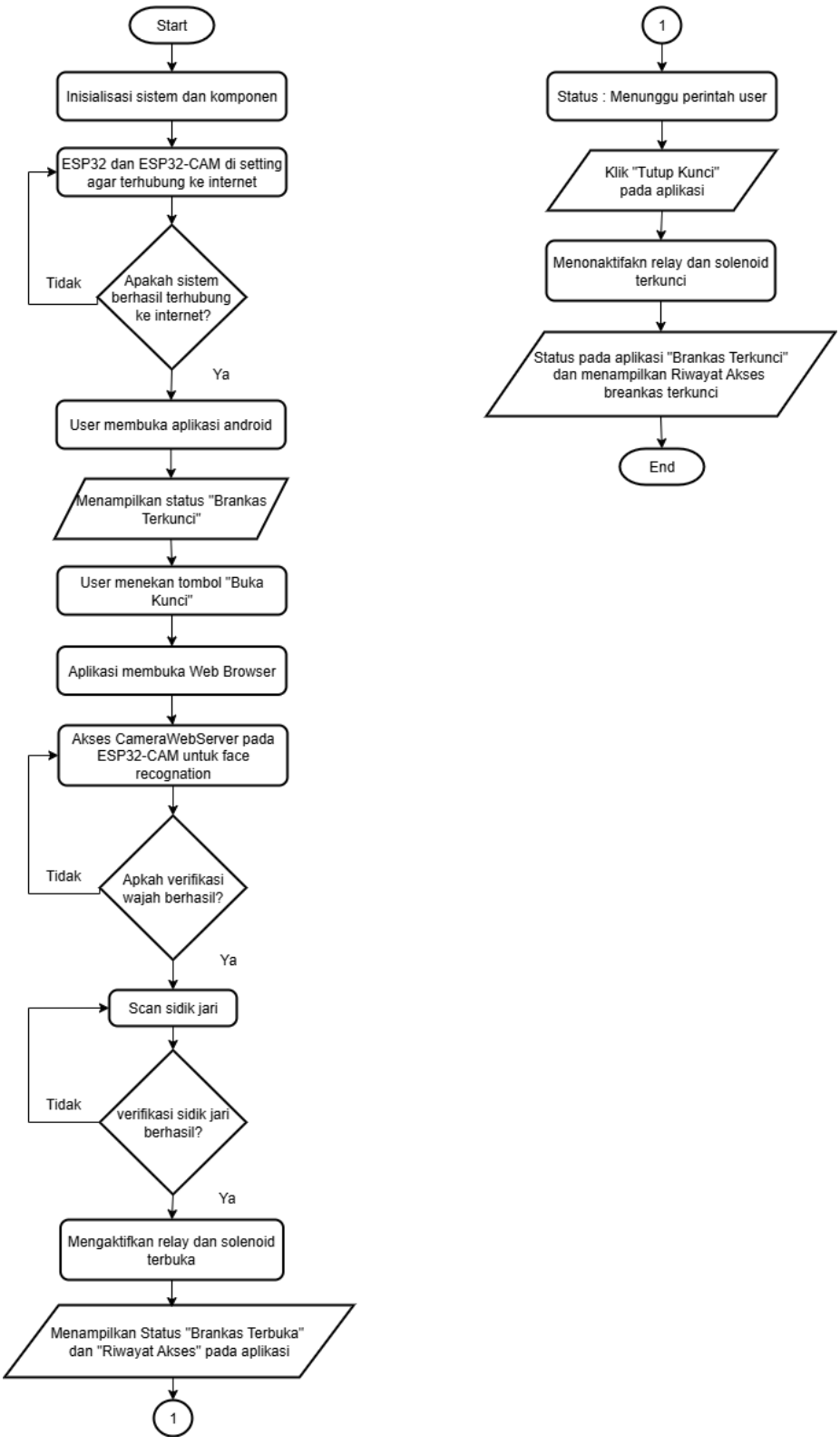


Figure III.6 Flowchart Network

4. Discussion Results

This testing was conducted by designing and programming the device using the Arduino IDE application to control the connected electronic components. By using this system, the owner can securely open the safe and receive status messages through an Android application developed with Android Studio and Firebase. Here are the tools and materials used in this safe security system, with the following details:

1. The adapter functions as the power source that connects and provides the necessary voltage to run the entire system, including the ESP32 and ESP32-CAM. This adapter ensures that all components have sufficient power to operate properly.
2. The ESP32 serves as the main microcontroller in this system. This ESP32 microcontroller is equipped with Wi-Fi and Bluetooth connectivity, enabling wireless communication. The ESP32 is responsible for managing the components within the system, such

as the fingerprint sensor, ESP32-CAM, relay, and buzzer. This microcontroller also receives and processes commands sent through the connected Android application.

3. The ESP32-CAM is an ESP32-based camera module used to capture the owner's face as part of the verification process. Its main function is to ensure that only registered owners can access the safe. This module must be connected to a network to function in supporting the facial verification process carried out through an Android application created with Android Studio and Firebase.
4. The fingerprint sensor is an electronic device used to read the fingerprints of the safe owner as part of the identity verification process. The captured fingerprint data is then sent to the ESP32 to be compared with the stored data. The result of this verification will determine whether the safe can be opened or not.
5. The solenoid door lock functions to lock and unlock the safe. This solenoid is controlled by a relay activated by the ESP32, based on successful facial and fingerprint verification. When the verification is successful, the relay will activate the solenoid to open the safe door.
6. The Android application is used as a control interface and to receive notifications related to the status of the safe. The owner can press the "Open Safe" button to open the safe through facial recognition and fingerprint verification.
7. The smartphone functions as a device that receives notifications from the Android application and controls the safe's security system through an application connected to ESP32 and Firebase. 8. The WiFi network serves as a communication medium between the ESP32 and ESP32-CAM and the Android application on the smartphone.

4.1 Software Testing

4.1.1 Testing with Arduino IDE

Testing using the Arduino IDE is conducted to ensure that the programs written for the ESP32 microcontroller, ESP32-CAM, and fingerprint sensor function as intended. The device testing carried out with the Arduino IDE is as follows:

a. Testing the ESP32 Microcontroller.

The initial step in the testing process is to open the Arduino IDE software. Once the application is open, the main interface will be displayed as shown in Figure 3.1. Before proceeding, ensure that the software configuration has been properly adjusted to support the ESP32 and ESP32-CAM. Next, select the appropriate board type and serial port corresponding to each microcontroller. After all the settings have been correctly configured, the programming and command installation process can proceed immediately.

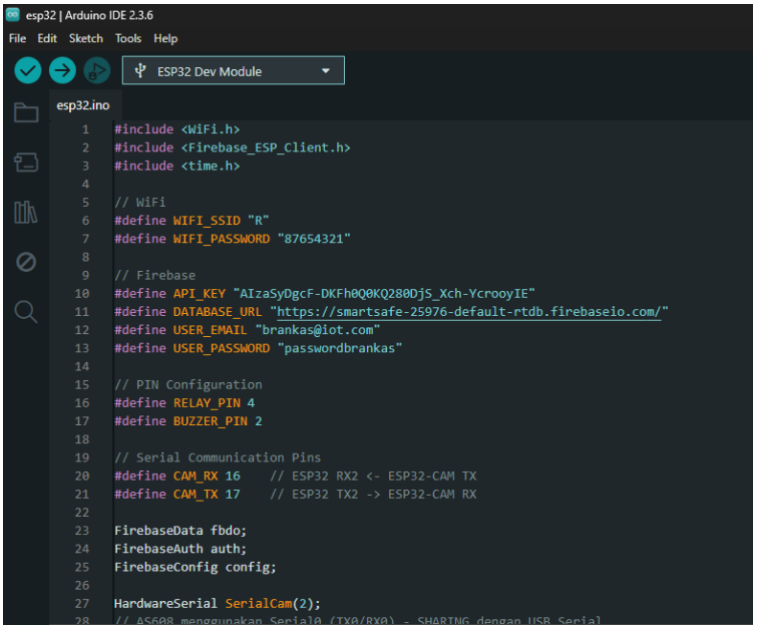


Figure IV.1 Display of the ESP32 Program

b. Face Detection Testing with ESP32-CAM

ESP32-CAM is used to test the IoT-based safe security system, which functions to utilize the camera in capturing facial images for the verification process. When the safe owner stands in front of the camera, the ESP32-CAM will automatically capture the facial image in front of it.

- The first step to start the testing is to open the Arduino IDE application. Once the Arduino IDE application is open, make sure the board and serial port are properly connected to "Board: AI Thinker ESP32-CAM" and "Port: COM5."
- Ensure that the Arduino IDE configuration is adjusted for the ESP32-CAM, and select the appropriate board, which is "AI Thinker ESP32-CAM," and the correct serial port. After that, the programming and command installation process can continue. Figure IV.2 below shows an example of the program used.

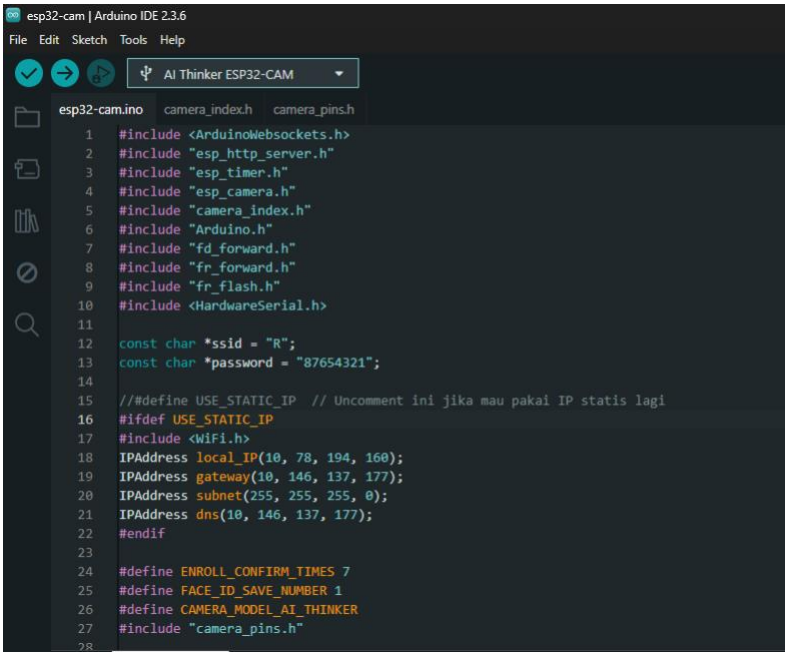


Figure VI.2 Program Display from ESP32-CAM

c. The next step is to access the IP displayed on the Serial Monitor. Access the IP using a Web Browser with the same network and connection as used by the ESP32-CAM to access the Camera Web Server from the ESP32-CAM. This process can be seen in the following Figure IV.3.

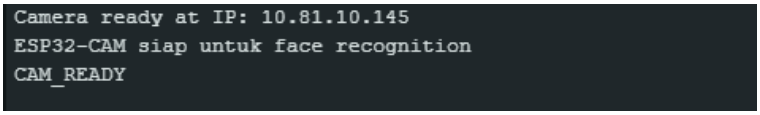


Figure IV.3 IP on ESP32-CAM

d. User face registration is done by entering the username, then pressing the "Add User" button. Next, the camera on the ESP32-CAM module will activate to detect the presence of a face and automatically scan and store facial data. The ESP32-CAM module will record seven facial samples from the user as the baseline data in the verification process. The interface display of the Camera Web Server during this process can be seen in Figure 3.4. The process of taking user face samples is shown in Table IV.1, while the results of verifying registered user faces are presented in Table IV.2.

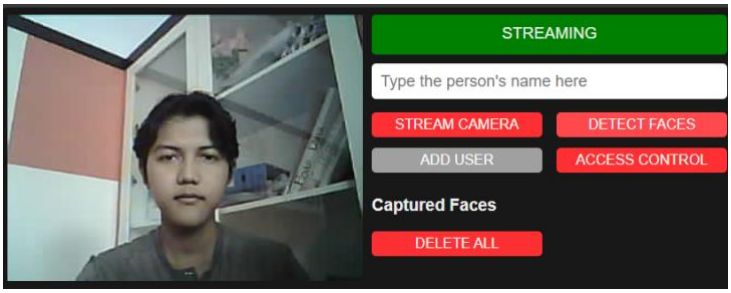
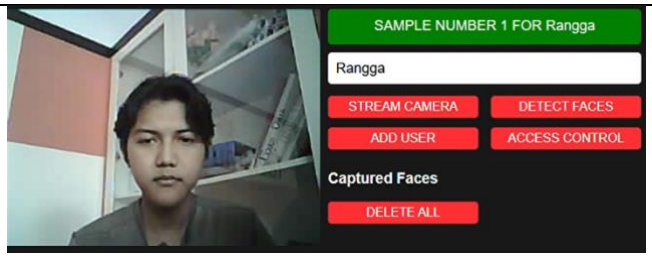


Figure IV.4 Display Results of Camera Web Server

No	Object	Sample Face Taken	Image Capture Results (Face Sample)
1	Face	Sample 1	

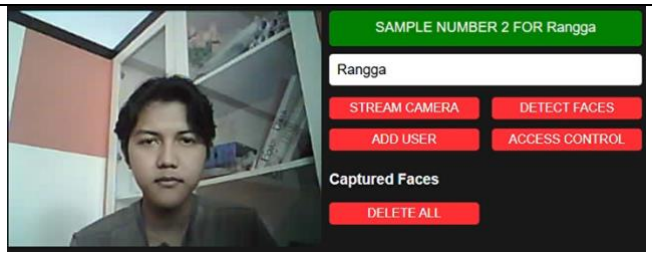
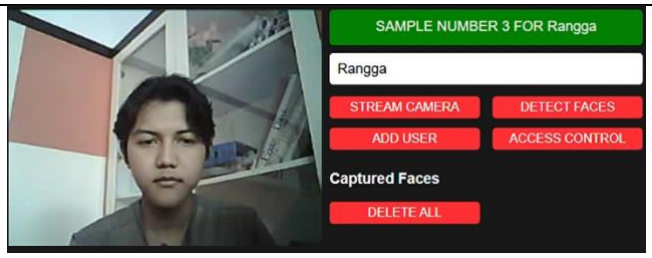
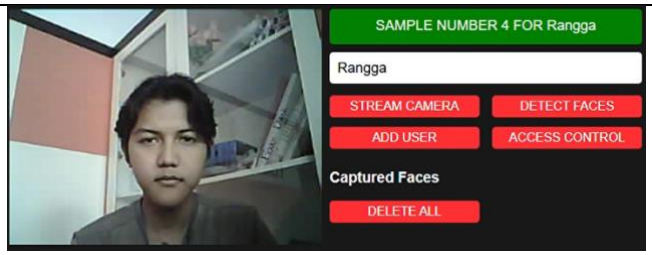
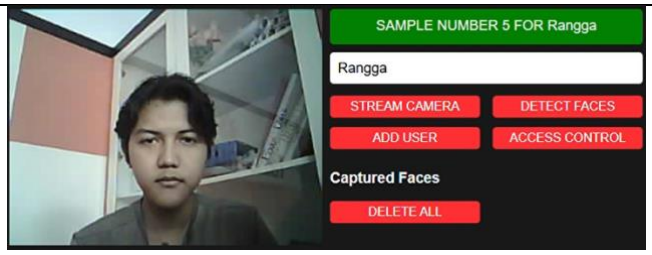
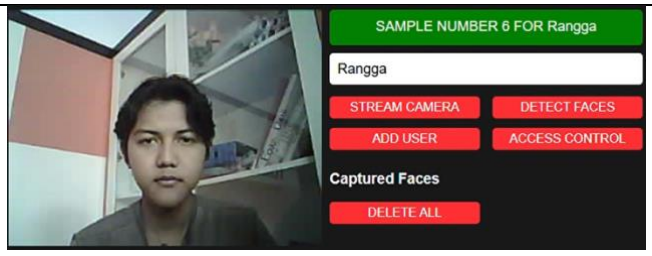
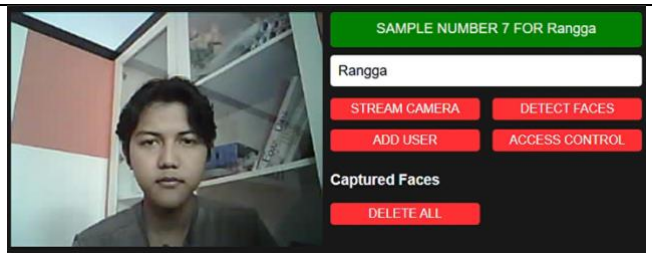
2	Face	Sample 2	
3	Face	Sample 3	
4	Face	Sample 4	
5	Face	Sample 5	
6	Face	Sample 6	
7	Face	Sample 7	

Table IV.1 Face Recording Testing on the ESP32-CAM Camera

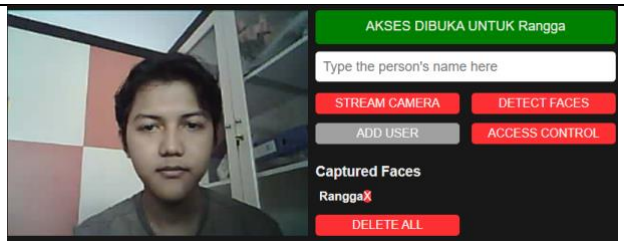
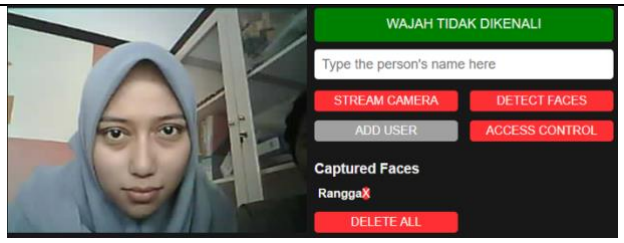
No	Object	Face Verification Results	Face Detection Status
1	Registered User Face		Detected
2	User's Face Not Registered		Not Detected

Table IV.2 Face Verification Testing on ESP32-CAM Camera

e. Fingerprint Sensor Testing

This Internet of Things (IoT)-based safe security system implements fingerprint authentication as an additional security layer. To open the safe, the owner must scan their fingerprint on a sensor integrated with the ESP32 module. The sensor will read the fingerprint data and send it to the ESP32 for verification against the fingerprint database stored in the system. If the scanned fingerprint matches the stored data, the system will confirm that the fingerprint belongs to the registered owner. The verification results will then be displayed in the status menu on the Android application. Fingerprint sensor testing is a crucial step in ensuring the system can accurately identify fingerprints and distinguish between registered and unregistered ones. The testing is presented in Table IV.3.

No	Testing	Finger	Remarks
1	Owner	Right Thumb	Detected
2	Owner	Right Index Finger	Not Detected
3	Owner	Left Thumb	Not Detected
4	Other People	Right Thumb	Not Detected
5	Other People	Right Index Finger	Not Detected

Table IV.3 Fingerprint Sensor Testing

If the scanned fingerprint matches the stored data, the relay will be activated to unlock the safe's solenoid lock. After this process is successful, the Android application will display the message "Safe Open" as an indicator that authentication has been completed. Figure IV.5 below shows the process of scanning the registered fingerprint, while Figure IV.6 displays the application status when the fingerprint verification process is successful.

Figure IV.5 Registered Fingerprint Scanning Process

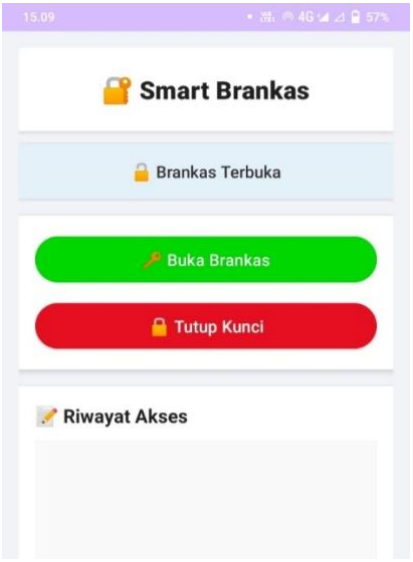


Figure IV.6 Application Display Result If the Fingerprint is Correct

Suppose the scanned fingerprint does not match the data stored in the system. In that case, the buzzer will sound as an indication that the authentication process has failed, and the status on the application will continue to display "Safe Locked." This mechanism is designed to ensure that only registered users can access the contents of the safe. Figure IV.7 below shows the process of scanning an unregistered fingerprint, while Figure IV.8 displays the view on the serial monitor when fingerprint verification fails.

Figure IV.7 Unregistered Fingerprint Scanning Process

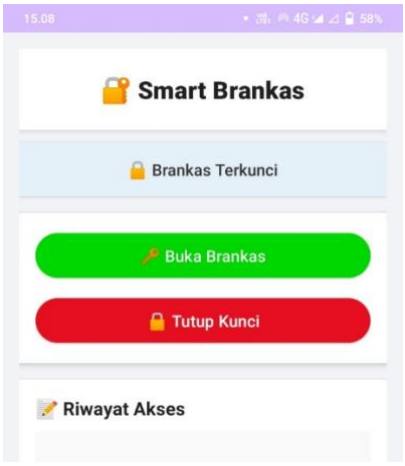


Figure IV.8 Application Display Result When Fingerprint Is Incorrect

4.1.2 Testing with Android Studio

This testing aims to ensure that the Android application developed using Android Studio runs smoothly and can communicate in real-time with hardware through Firebase. This application serves as a user interface to monitor and control the security status of the safe, including displaying authentication status and automatically opening the safe if verification is successful. The testing is divided into three main parts, namely:

a. Android Studio Program Code Display
The initial display in Android Studio shows the structure of the application project that has been created, including layout files (XML), the main program file (MainActivity.kt or .java), and Firebase configuration. At this stage, interface programming and logic functions are performed to connect the Firebase Realtime Database with the UI components in the application. The compilation results of the project in Android Studio show that there are no errors or bugs, so the application is ready to be run on both the emulator and directly on Android devices. Figure IV.9 shows the initial appearance of the project in Android Studio.

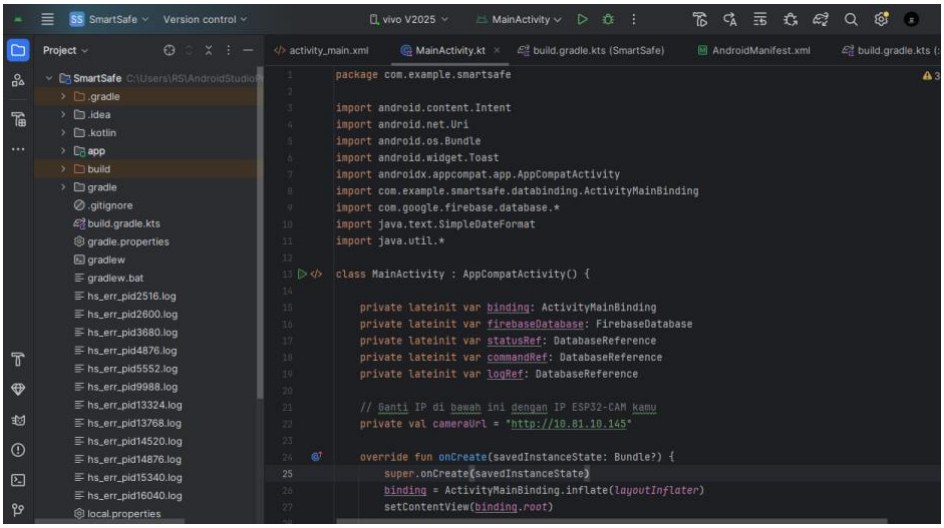


Figure IV.9 Display of Android Studio Program Code

b. The Interface of the Created Application
The application interface is designed to be simple, responsive, and easy for users to understand. Some of the main elements displayed include:
• The application title at the top is the system's identity.
• User authentication status (successful/failed).
• The "Open Vault" button directs users to the web browser and performs face recognition. • The "Close Lock" button to close the vault door again.
• Access history or other system information.
Figure IV.10 shows the interface of the created Android application.

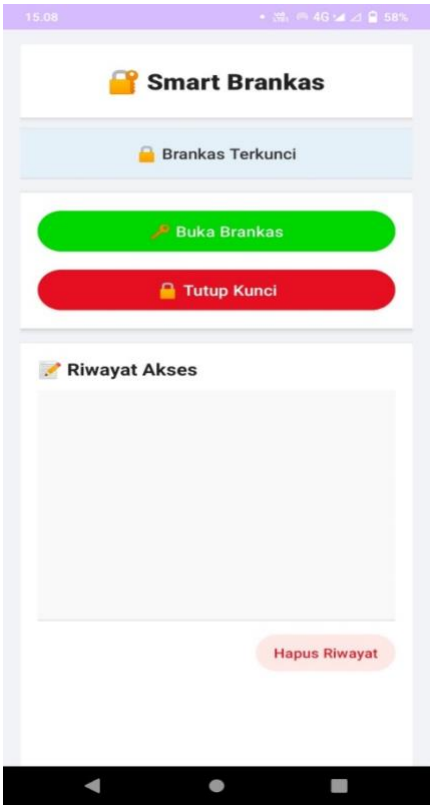


Figure IV.10 Android Application Interface Display

c. Vault Status Display and Access History

After the authentication process is successful (either through facial recognition or fingerprint), the status will be automatically updated on Firebase and displayed in the Android application. The message displayed on the screen will indicate "Vault Open," signifying that the system has granted access to the vault. Additionally, the application also displays a status log, which is the history of the safe's status automatically recorded in Firebase, including the time and type of authentication performed. After the access history is recorded, the status in the application will be updated to "Waiting for Command" for the next command, which is "Lock Door." This feature is useful for monitoring system usage activities periodically.

This test shows that the integration between the Android application, Firebase, and the vault security system is functioning well. Status changes on the hardware can be displayed in real-time on the application, and vice versa, commands from the application can affect the hardware status according to the system design. Figure IV.11 shows the status display of the vault that has been opened and the status waiting for commands, as well as the status log on the Android application.

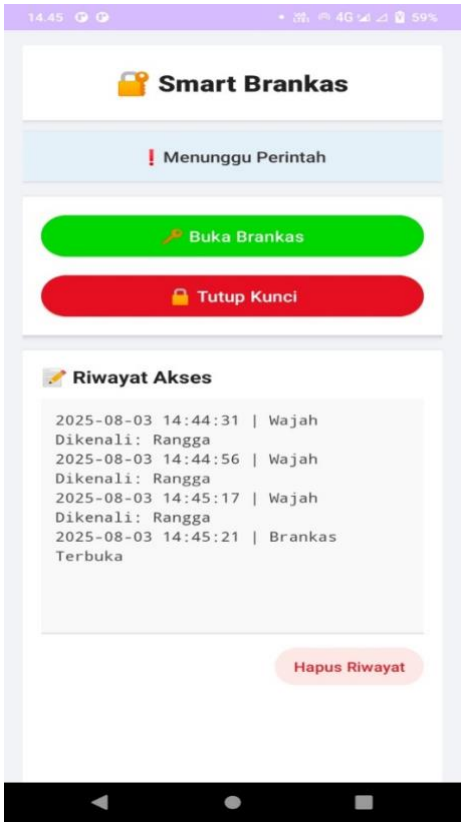


Figure IV.11 Status Display "Safe Open" and Status Log

4.2 Hardware Testing

After the entire circuit of the "Development of a Dual Biometric Authentication System Based on IoT with Facial and Fingerprint Recognition for Safe Security" has been successfully assembled, the next stage is to integrate all components into a single system. This integration process includes the combination of the ESP32 module, ESP32-CAM, fingerprint sensor, relay, solenoid door lock, buzzer, and other supporting components that have been adjusted to the system design that has been created. Figure IV.12 below shows the final assembly of the devices and components, illustrating the arrangement and connections between the components in the system comprehensively.

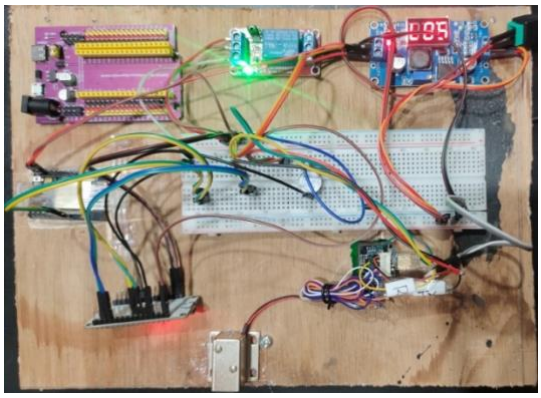


Figure IV.12 Integration Circuit of Modules and Components on the Device

4.2.1 Power Testing Using a Battery

Power testing aims to evaluate the system's performance under conditions of using an independent power source, namely, using a battery as the main supply. This test is important to ensure that all components in the dual authentication system can function optimally without relying on external power sources (such as adapters or USB ports).

The power source used in this test is a 18650 Li-ion battery with a capacity of 3.7V and a maximum current of 2200 mAh, connected in series to obtain a voltage suitable for the ESP32 module and other devices. To maintain output voltage stability and protect the devices from drastic voltage spikes or drops, this circuit is also equipped with a step-up/down converter (DC-DC converter), with an input power of 15V that will be reduced to 5V. During the test, the system was fully activated, including the activation of the camera on the ESP32-CAM, fingerprint sensor, and actuators such as relays and solenoid locks. The test results show that the system can operate normally.

With this testing, it can be concluded that the developed IoT-based safe security system is capable of operating independently using a battery for a certain period, thus supporting its application in locations without a permanent electricity supply, as long as power consumption management is carried out efficiently. Figure IV.13 below shows the results of the power test using the battery with the device and components operating normally.

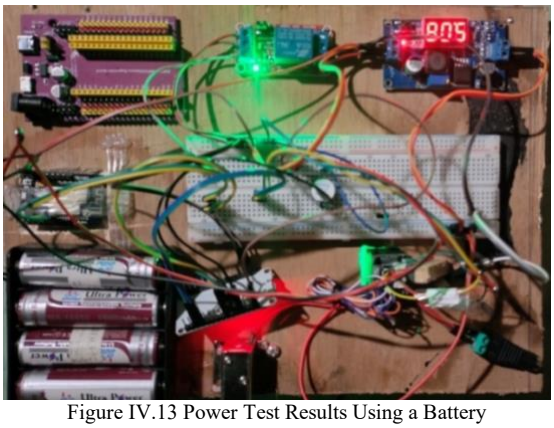


Figure IV.13 Power Test Results Using a Battery

4.2.2 Overall Device Testing Results

After all the hardware has been successfully programmed into the microcontroller and run using the downloader, the program will be automatically saved and ready to be run directly by the microcontroller. It can be seen in Figure IV.14 below.

Figure IV.14 Overall Device Testing Results

5. Conclusions

Based on the design, implementation, and testing results of the IoT-based dual biometric authentication system with facial recognition and fingerprint for safe deposit box security, it can be concluded that:

a. The system design was successfully implemented; the IoT-based dual biometric authentication system was successfully designed and developed by integrating the ESP32-CAM for facial recognition and the AS608 sensor for fingerprint verification. The NodeMCU ESP32 serves as the main microcontroller that manages the entire authentication process sequentially.
b. The system successfully implemented a layered authentication mechanism where facial verification must be successful before the system activates the fingerprint sensor. The solenoid lock will only open if both authentication processes are completed in sequence, increasing the security level compared to a single authentication system.
c. IoT Connectivity Works Well The system successfully connects to the Firebase Realtime Database, allowing users to monitor the status of the safe in real-time through the Android application. The application can display the status "Safe Open," "Safe Locked," and store access history with accurate timestamps.

6. Suggestions

For further development of this dual biometric authentication safe security system, it is recommended:

a. The system can be further improved by enhancing facial recognition accuracy by integrating an automatic lighting system to address issues with suboptimal lighting conditions.
b. It is expected that further research can develop the system to recognize multiple users (more than one user) with a larger database.

Reference

[1] A. Anggawan & M. Yuhendri, "Kendali Tegangan Output Buck Converter Menggunakan Arduino Berbasis Simulink Matlab", *JTEIN: Jurnal Teknik Elektro Indonesia*, *2*(1), 34–39, 2021, https://doi.org/10.24036/jtein.v2i1.110.

[2] A. Anifam, V. Kurnia Bhakti & W. Eko Nugroho, "Rancang Bangun Sistem Keamanan Brankas Menggunakan Sidik Jari (Finger print) Berbasis Arduinoi uno Dengan Notifikasi Telegram", *Journal of Telecommunication, Electronics, and Control Engineering (JTECE)*, 6, 2021.

[3] A. Maulana, A. Ullah, A. Faizal & H. Zarory, "Dual Sistem Keamanan Pada Pintu Dengan Pengenalan Wajah *Local Binary Pattern* Histogram (LBPH) Dan Sidik Jari serta Notifikasi Telegram", *Jurnal Al-Azhar Indonesia Seri Sains Dan Teknologi*, 10 (2), 153-161, 2025, http://dx.doi.org/10.36722/sst.v10i2.3696.

[4] A. Wibowo, "Internet of Things (IoT) dalam Ekonomi dan Bisnis Digital", Semarang: *Penerbit Yayasan Prima Agus Teknik*, 2023.

[5] A.A. Mahligai, N. Iksan, P. Gunoto, & I.Y. Panessai, "Perancangan Sistem Keamanan Brankas Dengan Verifikasi Password Dan Sidik Jari Berbasis Iot", *Sigma Teknika*, *5*(1), 100–107, 2022, https://doi.org/10.33373/sigmateknika.v5i1.4141.

[6] A.B. Sinabang, M. Martias & H. Adianto, "Alat Pengaman Brankas Berbasis Fingerprint Menggunakan Nodemcu Esp8266 Notifikasi Telegram", *Insantek*, *4*(1), 18–24, 2023, https://doi.org/10.31294/insantek.v4i1.2121.

[7] A.L.R. Dicky, T.W. Purboyo & R.E. Saputra, "Perancangan Sistem Keamanan Aplikasi Pada Lemari Brankas Dengan Menggunakan Modul Node MCU Yang Terkoneksi Dengan ESP8266", *E-Proceeding of Engineering*, *8*(6), 12110–12117, 2021.

[8] D. Rika Widianita, "Prototype Sistem Pengaman dan Pelacak Brankas Menggunakan *Fingerprint* Dan GPS Berbasis *Internet of Things*", *AT-TAWASSUTH: Jurnal Ekonomi Islam*, *VIII*(I), 1–19, 2023.

[9] E.P. Lumbanraja, S. Saniman & T. Tugiono, "Sistem Monitoring Keamanan Brankas Menggunakan Face Recognition Berbasis Mikrokontroler ESP32-CAM", *Jurnal Sistem Komputer Triguna Dharma (JURSIK TGD)*, *2*(3), 169–176, 2023, https://doi.org/10.53513/jursik.v2i3.6560.

[10] F. Nabila, "Sistem Keamanan Brankas Menggunakan Face Recognition Dan One-Time Password Berbasis Internet of Things", *Skripsi*, *4*, 2021.

[11] I. G. M. N. Desnanjaya, "Sistem Brankas Berbasis Internet Of Things Mengunakan Arduino Mega 2560", *Jurnal RESISTOR (Rekayasa Sistem Komputer)*, *5*(2), 131–137, 2022, https://doi.org/10.31598/jurnalresistor.v5i2.1169.

[12] J. Manurung, & B. Fernandes, "Alat Keamanan Brankas Perhiasan Dengan Face Recognition dan Fingerprint Berbasis Arduino Mega 2560 Terkendali Smartphone", *Jurnal Sains Informatika Terapan*, *2*(3), 90–95, 2023, https://doi.org/10.62357/jsit.v2i3.182.

[13] M. Romzi, "*Logika dan Algoritma* (Issue tahun 1736)", 2012.

[14] N. Pitaloka, A.M.H. Pardede & H. Khair, "*Design of a Safe Security System Based on Internet of Things Using Face and Fingerprint Detection*", *Journal of Artificial Intelligence and Engineering Applications (JAIEA),* 4(1), 348-357, 2024.

[15] O.R. Arsyad & K.P. Kartika, "Rancang Bangun Alat Pengaman Brankas Menggunakan Sensor Sidik Jari Berbasis Arduino", *JATI (Jurnal Mahasiswa Teknik Informatika)*, *5*(1), 1–6, 2021, https://doi.org/10.36040/jati.v5i1.3285.

[16] T. Aprilianto, S. Arifin, S. Jatmika, P. Studi, & S. Komputer, "*Aplikasi Sistem Keamanan Brankas Menggunakan E-KTP"*, *14*(1), 2022.