Smart Pedestrian 2-Way Crossing System for the Visually Impaired Using Esp32

Herjed R. Flores, Andre Axel M. Grageda, Mark Ian Guillen, Charles Miranda, Gerrold M. Sillacay, Minerva C. Zoleta, Albert einstein C. Zoleta

Department of Computer Engineering, Eulogio "Amang" Rodriguez Institute of Science and Technology, Manila, Philippines

DOI: https://doi.org/10.47772/IJRISS.2026.100600025

Received: 20 May 2026; Accepted: 25 May 2026; Published: 16 June 2026

ABSTRACT

Traditional pedestrian infrastructure in urban environments heavily relies on visual signals, posing severe safety risks and limiting independent mobility for visually impaired individuals. To address this challenge, this study presents the design and development of a *Smart Pedestrian 2-Way Crossing System for the Visually Impaired Using ESP32*. Developed through an experimental research design, the prototype integrates an ESP32 microcontroller with a Radio Frequency Identification (RFID) system (MFRC522 reader) alongside automated audio-visual feedback modules. Visually impaired pedestrians can tap an authorized RFID card or keychain to trigger traffic light transitions that stop vehicle traffic, while simultaneously initiating clear audio guidance to signify when it is safe to cross.

Experimental testing under controlled conditions demonstrated a high system reliability, achieving a 90% success rate in baseline RFID credential detection. Performance benchmarks showed rapid processing capabilities, with the RFID card yielding a response time of 800 ms compared to 950 ms for the keychain alternative. Wireless testing indicated that the ESP32 Wi-Fi module maintains stable data communication with the traffic light infrastructure within an optimal range of 10 to 15 meters. Furthermore, simulation data regarding the walking speeds of visually impaired individuals suggested that a safe crossing configuration requires allocating a minimum duration of 20 seconds for standard lanes spanning 8 to 10 meters. Ultimately, the findings confirm that the proposed system serves as a viable, scalable, and inclusive solution for transforming pedestrian infrastructure and fostering greater independence for visually impaired citizens in Philippine cities.

Keywords: Assistive Technology, ESP32, Embedded Systems, Pedestrian Safety, Radio Frequency Identification (RFID), Visually Impaired Mobility.

INTRODUCTION

Background of the Study

In the Philippines, visually impaired individuals face significant challenges in navigating pedestrian crossings safely. Traffic intersections pose considerable risks due to the absence of adequate assistive infrastructure. According to the National Council on Disability Affairs (NCDA), there are approximately 1.9 million registered persons with disabilities in the country as of January 2025. This issue necessitates the development of an smart pedestrian crossing system that enhances the mobility and safety of visually impaired individuals in urban environments. Technological advancements, particularly in the field of embedded systems, have paved the way for smart solutions in assistive technology. One such innovation is the integration of Radio Frequency Identification (RFID) technology and microcontroller-based systems, such as ESP32, into pedestrian crossing mechanisms. These systems offer an automated approach to traffic light control, allowing visually impaired individuals to activate stoplights through RFIDenabled devices such as cards or keychains. While specific studies

ISSN No. 2454-6186 | DOI: 10.47772/IJRISS | Volume X Issue VI June 2026

on RFID-assisted pedestrian systems in the Philippines are limited, global research indicates the potential benefits of such technologies. For instance, a study by Smith et al. (2020) demonstrated that RFID-assisted pedestrian systems significantly reduce road-crossing difficulties for visually impaired individuals. A local study by Bermudo et al. (2020) from De La Salle University discusses the development of an assistive device utilizing ultrasonic sensors to aid visually impaired individuals in obstacle detection. Furthermore, Philippine government agencies and organizations have taken steps to support visually impaired individuals. The Department of Social Welfare and Development (DSWD) has provided assistive devices to persons with disabilities to enhance their mobility and independence. The NCDA has also approved guidelines for the provision of assistive devices and technology services in the Philippines.

Statement of the Problem

The main problem addressed in this study is the lack of accessible pedestrian crossing solutions for visually impaired individuals in the Philippines. Traditional traffic light systems primarily rely on visual signals, making them ineffective for visually impaired users. This study aims to answer the following research questions:

1. How effective is the RFID-based stoplight system in facilitating safe pedestrian crossing for visually impaired individuals in the Philippines?

2. What is the response time of the RFID system in activating traffic lights in a Philippine urban setting?

3. How reliable is the system in detecting RFID credentials with high accuracy under Philippine environmental conditions?

4. How fast a visually impaired individual to cross a pedestrian lane?

General Objective

The primary objective of this study is to design and implement a Smart Pedestrian Crossing System for the Visually Impaired Using ESP32 to enhance pedestrian safety and mobility in the Philippines.

Specific Objectives

1. To develop a prototype that integrates RFID technology with ESP32-based stoplight control.

2. To assess the accuracy and reliability of RFID card/keychain detection in triggering pedestrian stoplights within Philippine urban environments.

3. To evaluate the response time of the system in activating stoplight sequences.

4. To know how long a visually impaired individual to cross a pedestrian lane.

5. To analyze user feedback and identify potential improvements for future implementations in Philippine cities.

Scope and Delimitation

This study focuses on developing and testing a prototype of Smart Pedestrian Crossing System for the Visually Impaired Using ESP32. The system allows visually impaired individuals to activate traffic stoplights by tapping an RFID card or keychain on an RFID reader. The study is limited to:

- The use of ESP32 as the primary microcontroller.

- RFID-based authentication to trigger stoplight transitions.

- Audio feedback for user guidance.
- Experimental testing in a controlled environment, rather than real-world deployment.

This study does not cover the full-scale implementation of the system in public traffic intersections across the Philippines, nor does it integrate additional accessibility features such as GPS tracking or smartphone applications. Future research may expand on these aspects to further enhance system usability and effectiveness.

Significance of the Study

The implementation of this system contributes to the safety and independence of visually impaired individuals in Philippine urban environments. It offers an innovative solution for reducing pedestrian risks at intersections, aligning with national efforts to improve accessibility and inclusivity. Additionally, this study serves as a foundation for future developments in assistive traffic technology, paving the way for smarter and more inclusive urban infrastructure in the Philippines.

METHODOLOGY

Research Design

This design uses an experimental research design to create an effective RFID-enabled pedestrian crossing system for visually impaired individuals. This project focuses on the integration of hardware components such as ESP32, MFRC522 (RFID reader), RFID card, RFID keychain, Speaker, SD Card, MP3-TP-16P, and TPA3110. The project helps visually impaired individuals cross the pedestrian safely by tapping the RFID card, which activates the stoplight and turns it red. The project's effectiveness depends on a series of tests to meet its objectives.

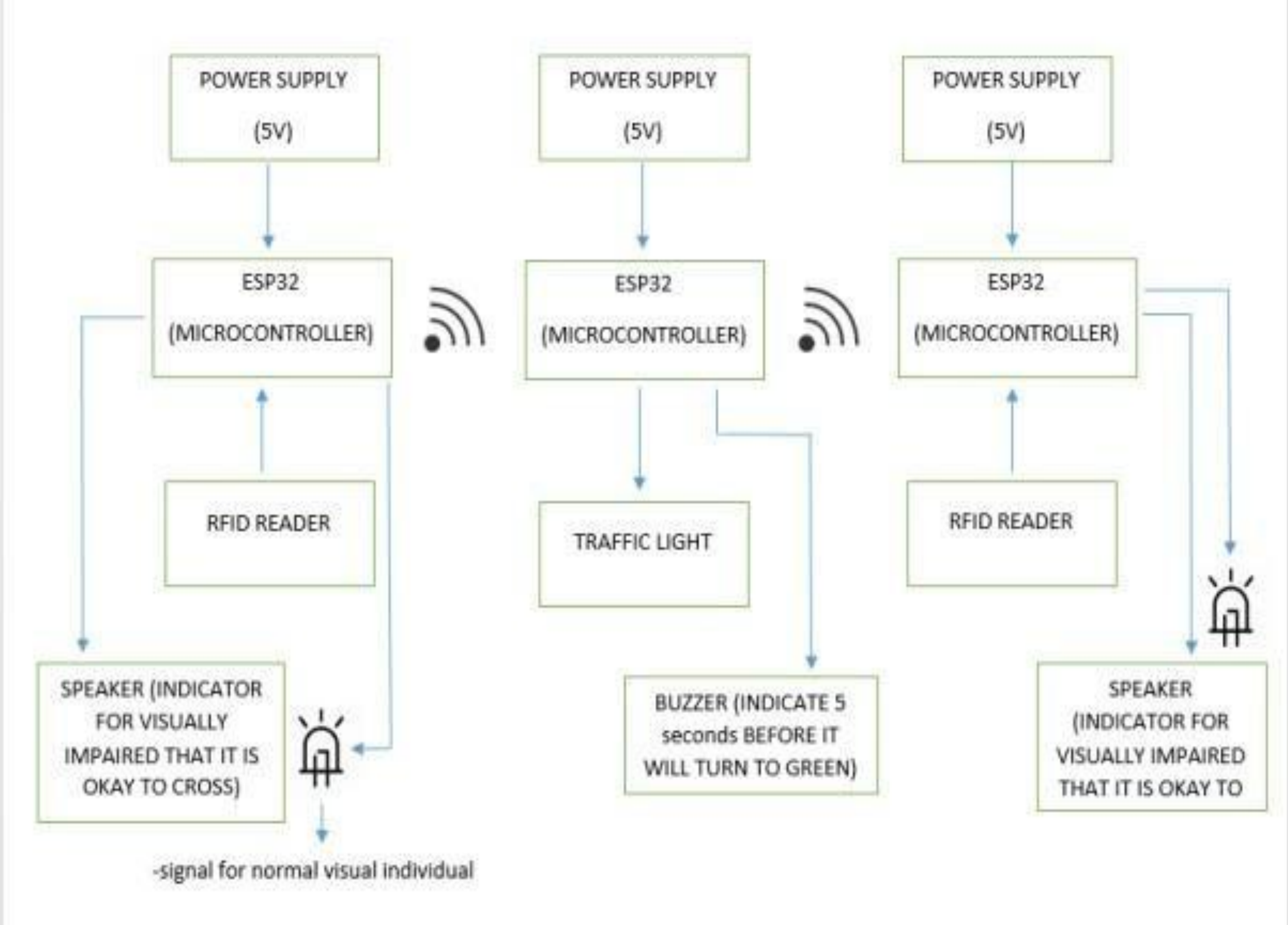
Materials and Equipment

- MFRC522 - RFID Reader - Reads unique identification data from RFID cards or keychains and sends it to the control system to trigger pedestrian crossing mechanisms.
- RFID Card (Main RFID) - Serves as the primary credential for visually impaired individuals to activate the RFID-enabled crossing system by tapping it on the reader.
- RFID Keychain (Alternative) - An alternative to the RFID card, offering a portable, keychain-style option for activating the system.
- ESP32 - Serves as the system's processing unit, receiving signals from the RFID reader and controlling the speaker and stoplight sequences.
- SD Card – Stores pre-recorded audio files for playback in the pedestrian crossing system. Speaker – Outputs audio alerts to assist visually impaired pedestrians.
- MP3-TP-16P – Plays stored MP3/WAV files when triggered by the system.
- TPA3110 – Amplifies the audio signals to ensure clear and loud sound output.
- Buzzer – Trigger when the red light comes to end.
- LED – Serves as a signal for normal sighted individual.

ISSN No. 2454-6186 | DOI: 10.47772/IJRISS | Volume X Issue VI June 2026

System design & Development

Figure 1. Block Diagram of the Proposed Improvement



1. Power Supply – Provides electrical power to the RFID reader, microcontroller (ESP32), buzzer, and stoplight to ensure the system functions properly.

2. RFID Reader – Detects the RFID card when tapped by a visually impaired individual and sends a signal to the microcontroller to initiate the stoplight sequence.

3. ESP32 - Acts as the brain of the system, processing signals from the RFID reader and executing the stoplight control logic. It controls the buzzer and stoplight transitions.

4. Speaker – Emits a sound for safety indication that the visually impaired can cross safely.

5. Traffic Light – Controls the pedestrian crossing signal with the following sequence: o Turns red after the 5-second indicator sound. o Stays in the red phase for pedestrian crossing. o Introduces a 5-second delay before turning green again to resume vehicle traffic.

6. LED- Gives signal to the individuals who have a normal eye sight. Figure 2. Circuit Diagram: Right Side of RFID system

ISSN No. 2454-6186 | DOI: 10.47772/IJRISS | Volume X Issue VI June 2026

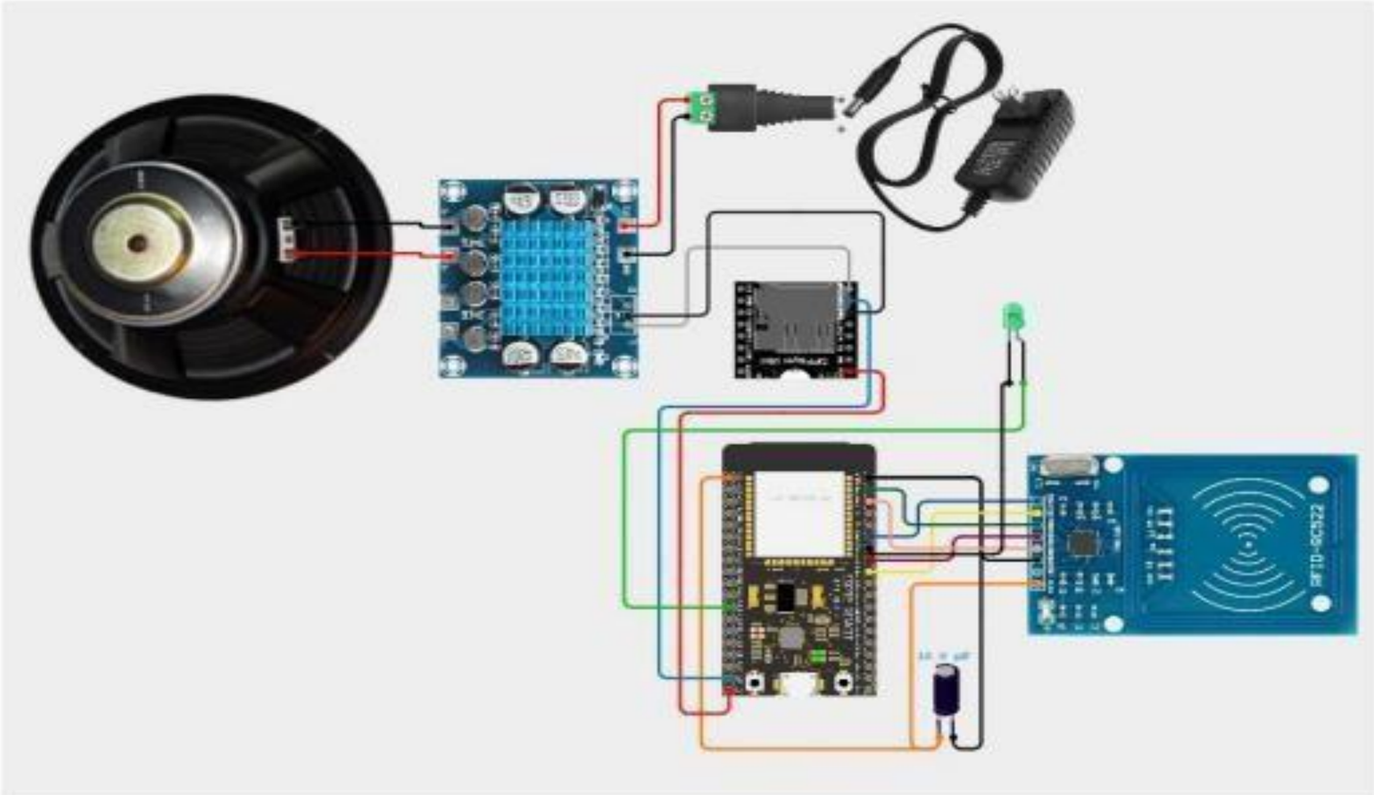


Figure 3. Circuit Diagram: Left Side of RFID system

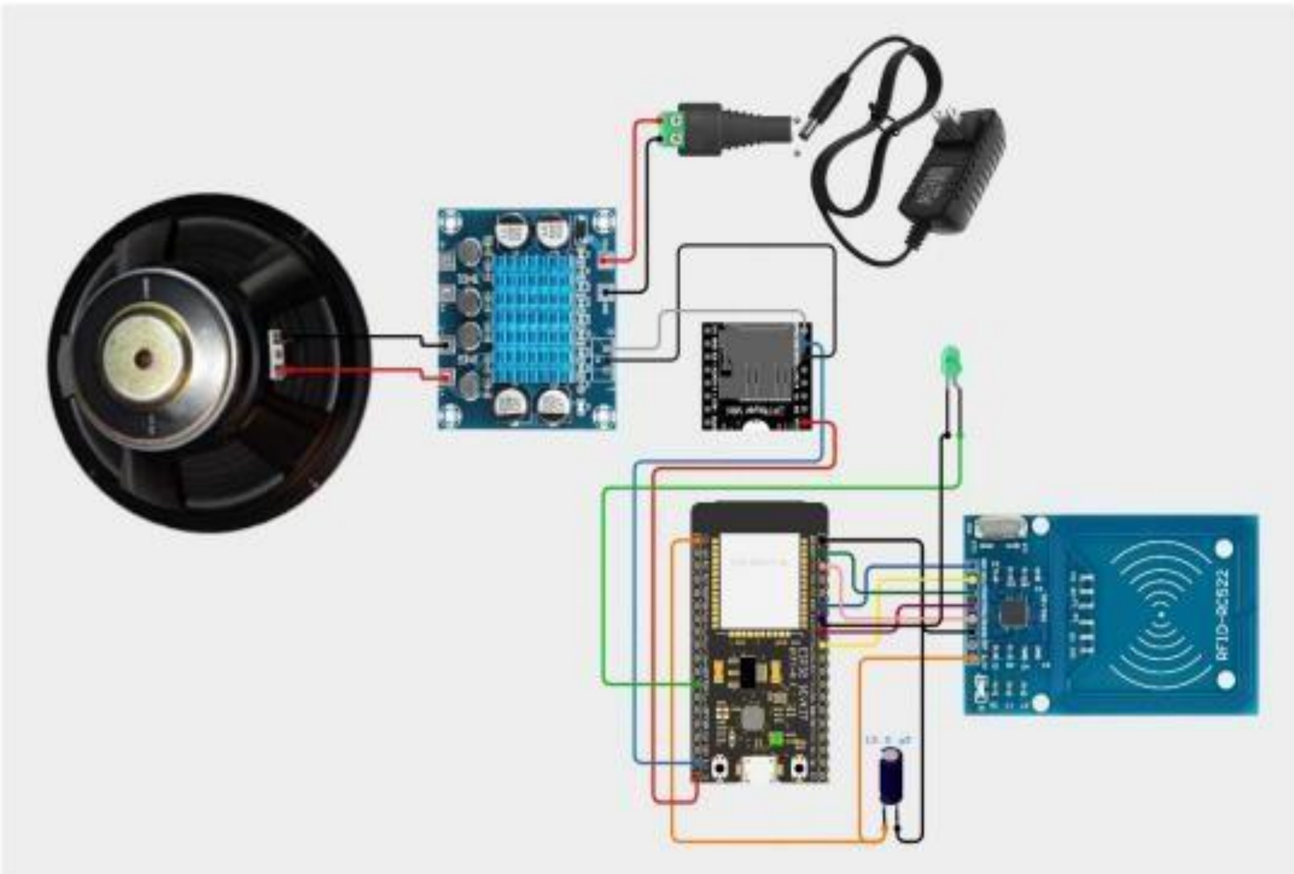


Figure 4. Traffic Light System

ISSN No. 2454-6186 | DOI: 10.47772/IJRISS | Volume X Issue VI June 2026

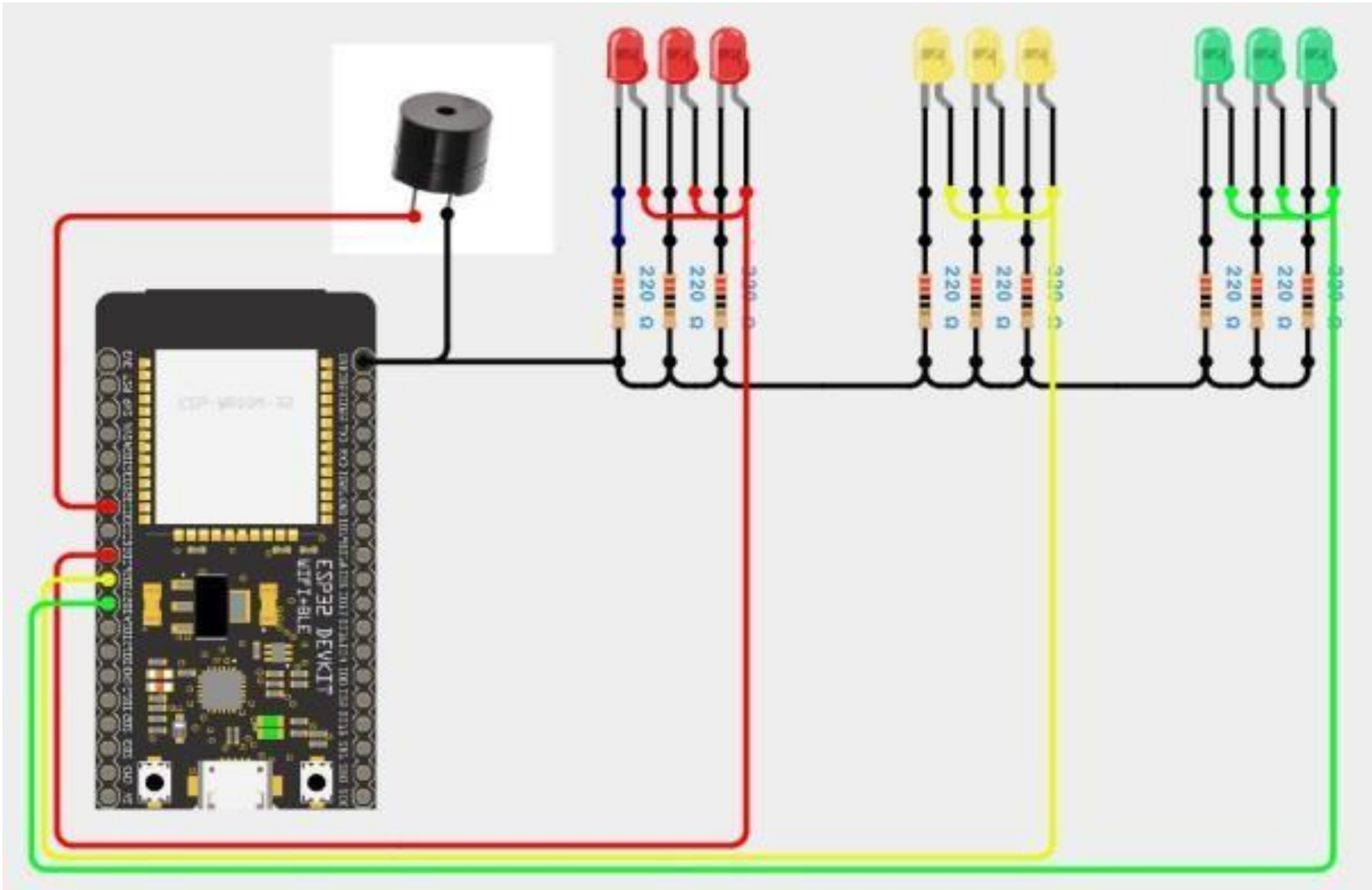


Figure 5. Source Code: RFID System (Same Right and Left)

```
#include <WiFi.h>
#include <MFRC522.h>
#include <SoftwareSerial.h>
#include <DFRobotDFPlayerMini.h>

// RFID setup
#define SS_PIN    21
#define RST_PIN   22
MFRC522 rfid(SS_PIN, RST_PIN);

// MP3 setup
SoftwareSerial mp3Serial(16, 17); // RX, TX
DFRobotDFPlayerMini mp3;

// LED pin setup (Choose an appropriate pin)
#define LED_PIN 13 // Example: Pin 13 for LED

const char* ssid = "hakdogers";
const char* password = "namoyoko";

// ESP 3 IP address for traffic light control
```

```
const char* esp3_ip = "192.168.41.225";

// Setup Wi-Fi and serial monitor
void setup() {
  Serial.begin(115200);
  WiFi.begin(ssid, password);

  // Initialize LED pin
  pinMode(LED_PIN, OUTPUT);
  digitalWrite(LED_PIN, LOW);  // Turn LED off initially

  while (WiFi.status() != WL_CONNECTED) {
    delay(1000);
    Serial.println("Connecting to WiFi...");
  }

  Serial.println("Connected to WiFi");

  // Initialize RFID
  SPI.begin();
  rfid.PCD_Init();

  // Initialize MP3
  mp3Serial.begin(9600);
  if (!mp3.begin(mp3Serial)) {
    Serial.println("DFPlayer Mini not found.");
    while (true);
  }

  mp3.volume(5); // Set volume (0-30)
}

// Main loop
void loop() {
  if (rfid.PICC_IsNewCardPresent() && rfid.PICC_ReadCardSerial()) {
    String uid = "";
    for (byte i = 0; i < rfid.uid.size; i++) {
      uid += String(rfid.uid.uidByte[i], HEX);
    }
    Serial.println("UID: " + uid);

    // Turn on LED when RFID is tapped
    digitalWrite(LED_PIN, HIGH);

    // Send request to ESP 3 to change traffic light
    WiFiClient client;
    if (client.connect(esp3_ip, 80)) {
      client.print("GET /changeLight?uid=" + uid + " HTTP/1.1\r\n");
      client.print("Host: " + String(esp3_ip) + "\r\n");
      client.print("Connection: close\r\n\r\n");
      client.stop();
    }

    // Play Folder 01 Audio 006
    mp3.playFolder(1, 6);  // Play Folder 01, Audio 006
    Serial.println("Playing Folder 01 Audio 006");
```

ISSN No. 2454-6186 | DOI: 10.47772/IJRISS | Volume X Issue VI June 2026

```
    delay(3000);  // Wait for the audio to finish (3 seconds)

    // Play Folder 01 Audio 001
    mp3.playFolder(1, 1);  // Play Folder 01, Audio 001
    Serial.println("Playing Folder 01 Audio 001");

    // Turn off LED after the audio finishes
    digitalWrite(LED_PIN, LOW);
  }
}
```

Figure 6. Source Code: Traffic Light System

```
#include <WiFi.h>
#include <WebServer.h>

const char* ssid = "hakdogers";
const char* password = "namoyoko";

// Traffic light pins
#define RED_PIN    25
#define YELLOW_PIN 26
#define GREEN_PIN  27
#define BUZZER_PIN 32  // Buzzer pin (change as per your wiring)

// Create a web server
WebServer server(80);

unsigned long previousMillis = 0;
const long redDuration = 27000;  // 27 seconds
const long greenDuration = 30000;  // 30 seconds
const long yellowDuration = 3000;  // 3 seconds
const long buzzerBeepDuration = 200; // Time for each beep in
milliseconds
const long buzzerInterval = 500; // Interval between each beep in
milliseconds

enum LightState { RED, GREEN, YELLOW };
LightState currentState = GREEN;  // Start with green light

bool forceCycle = false;  // Flag to force a specific light cycle after
UID

void setup() {
  Serial.begin(115200);
  WiFi.begin(ssid, password);

  // Wait until WiFi is connected
  while (WiFi.status() != WL_CONNECTED) {
    delay(1000);
    Serial.println("Connecting to WiFi...");
  }

  Serial.println("Connected to WiFi");
```

```
  // Initialize Traffic Light Pins
  pinMode(RED_PIN, OUTPUT);
  pinMode(YELLOW_PIN, OUTPUT);
  pinMode(GREEN_PIN, OUTPUT);
  pinMode(BUZZER_PIN, OUTPUT); // Initialize buzzer pin

  // Set up web server routes
  server.on("/changeLight", HTTP_GET, handleChangeLight);
  server.on("/verifyCrossing", HTTP_GET, handleVerifyCrossing);  //
This is the verification route
  server.begin();
}

void loop() {
  server.handleClient();
  trafficLightCycle();
}

void trafficLightCycle() {
  unsigned long currentMillis = millis();

  // If forceCycle is true, ensure the traffic light is managed
correctly
  if (forceCycle) {
    switch (currentState) {
      case YELLOW:
        digitalWrite(RED_PIN, LOW);
        digitalWrite(GREEN_PIN, LOW);
        digitalWrite(YELLOW_PIN, HIGH);
        if (currentMillis - previousMillis >= yellowDuration) {
          previousMillis = currentMillis;
          currentState = RED;  // Go to red after yellow
        }
        break;

      case RED:
        digitalWrite(RED_PIN, HIGH);
        digitalWrite(YELLOW_PIN, LOW);
        digitalWrite(GREEN_PIN, LOW);

        // Trigger the buzzer 5 seconds before the red light finishes
every time red is active
        if (currentMillis - previousMillis >= redDuration - 5000 &&
currentMillis - previousMillis < redDuration) {
          beepBuzzer();
        }

        if (currentMillis - previousMillis >= redDuration) {
          previousMillis = currentMillis;
          currentState = GREEN;  // Go to green after red
          forceCycle = false;  // Reset the forced cycle flag
        }
        break;

      case GREEN:
        // We will stay on green until the timer expires
```

ISSN No. 2454-6186 | DOI: 10.47772/IJRISS | Volume X Issue VI June 2026

```
        break;
      }
  } else {
    // Normal light cycle (green → yellow → red)
    switch (currentState) {
      case GREEN:
        digitalWrite(RED_PIN, LOW);
        digitalWrite(YELLOW_PIN, LOW);
        digitalWrite(GREEN_PIN, HIGH);
        if (currentMillis - previousMillis >= greenDuration) {
          previousMillis = currentMillis;
          currentState = YELLOW; // Proceed to yellow after green
        }
        break;

      case YELLOW:
        digitalWrite(RED_PIN, LOW);
        digitalWrite(YELLOW_PIN, HIGH);
        digitalWrite(GREEN_PIN, LOW);
        if (currentMillis - previousMillis >= yellowDuration) {
          previousMillis = currentMillis;
          currentState = RED; // Proceed to red after yellow
        }
        break;

      case RED:
        digitalWrite(RED_PIN, HIGH);
        digitalWrite(YELLOW_PIN, LOW);
        digitalWrite(GREEN_PIN, LOW);

        // Trigger the buzzer 5 seconds before the red light finishes
        if (currentMillis - previousMillis >= redDuration - 5000 &&
currentMillis - previousMillis < redDuration) {
          beepBuzzer();
        }

        if (currentMillis - previousMillis >= redDuration) {
          previousMillis = currentMillis;
          currentState = GREEN; // Proceed to green after red
        }
        break;
    }
  }
}

// Buzzer function that makes multiple beeps
void beepBuzzer() {
  static unsigned long lastBuzzerTime = 0;
  static bool buzzerState = false;
  unsigned long currentMillis = millis();

  if (currentMillis - lastBuzzerTime >= buzzerInterval) {
    buzzerState = !buzzerState;
    digitalWrite(BUZZER_PIN, buzzerState ? HIGH : LOW);
    lastBuzzerTime = currentMillis;
  }
}
```

```
void handleChangeLight() {
  String uid = server.arg("uid");
  Serial.println("Received UID: " + uid);

  if (uid.length() > 0) {
    forceCycle = true;
    currentState = YELLOW;
    previousMillis = millis();
    server.send(200, "text/plain", "Light Changed");
  }
}

// Handle UID verification request from the second ESP32 (RFID ESP32)
void handleVerifyCrossing() {
  String uid = server.arg("uid");
  Serial.println("Received UID for verification: " + uid);

  if (uid.length() > 0) {
    // If UID is valid, confirm successful crossing
    Serial.println("Successfully verified crossing for UID: " + uid);

    // Immediately force the traffic light to green after crossing
verification
    currentState = GREEN;  // Force the traffic light to green
    previousMillis = millis();  // Reset the timer for the green light

    forceCycle = false; // Reset the flag for the normal cycle

    Serial.println("Traffic light is forced to green now.");
    server.send(200, "text/plain", "Crossing Verified and Light Forced
to Green");
  } else {
    server.send(400, "text/plain", "Invalid UID");
  }
}
```

Hardware Components

1. Power Supply
    - 12V Adapter
    - Battery (Optional, for backup power)
2. Microcontroller
    - ESP32 (microcontroller to process signals and control the traffic light)
3. RFID System
    - RFID Reader Module (e.g., MFRC522 or RC522)

ISSN No. 2454-6186 | DOI: 10.47772/IJRISS | Volume X Issue VI June 2026

- RFID Cards or Key Tags (for visually impaired users to trigger the system)

4. Traffic Light System
   - LEDs (Red, Yellow, Green) or a Traffic Light Module
   - Relay Module (if using high-power stoplights instead of LEDs)
5. Speaker
   - For audio safety indicator
6. SD Card
   - 8GB and above, preferably 32GB
7. MP3-TP-16P
   - Stored wav files
8. TP3110
   - For amplification of audio
9. LED

Procedures

1. User Arrival
   - A visually impaired individual approaches the pedestrian crossing equipped with the RFID system
2. Tapping the RFID Credential
   - ESP32 (microcontroller to process signals and control the traffic light)
3. Tag Identification
   - The RFID reader scans the unique identification (UID) from the card or keychain
   - The UID is sent to a microcontroller (ESP32) for verification and processing.
4. Signal Processing
   - Once verified, the ESP32 activates, which sends a signal to the traffic light control system
5. Traffic Light Activation
   - The traffic light switches to red for vehicles, stopping oncoming traffic to create a safe crossing environment
6. Audio and Visual Alerts
   - An audio announcement (e.g., “You can now cross the road safely”) is triggered to guide the visually impaired pedestrian

ISSN No. 2454-6186 | DOI: 10.47772/IJRISS | Volume X Issue VI June 2026

7. Pedestrian Crossing

   - The individual safely crosses the pedestrian lane within the designated time frame

   - A countdown timer may be included to signal the remaining crossing time

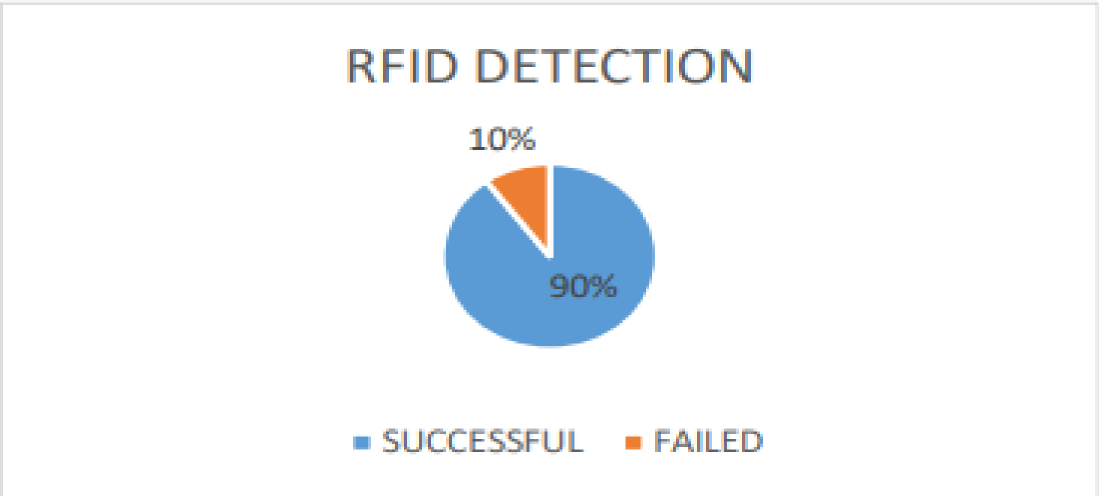
8. System Reset

   - After the crossing time elapses, the system automatically resets

RESULTS AND DISCUSSION

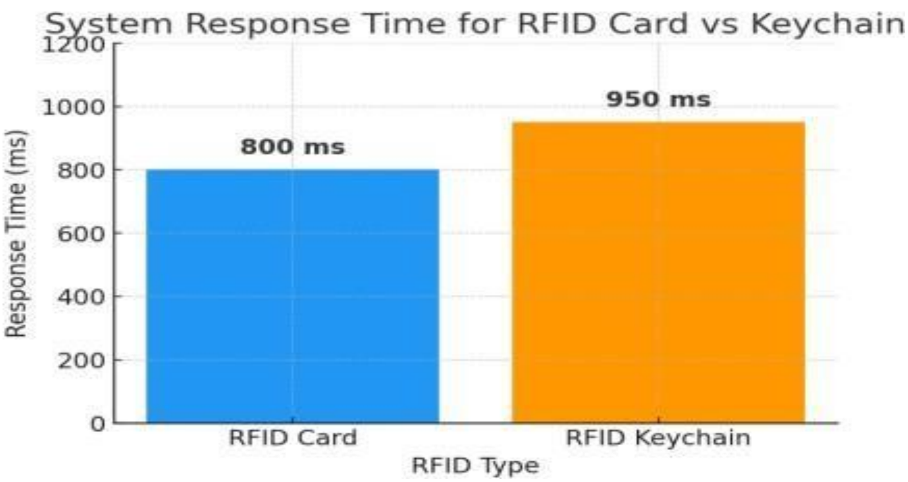
This chapter presents the data collected from trials conducted by the developers to evaluate the system's stability and effectiveness. The following analysis and interpretation are based on the functionalities of the Smart Pedestrian Crossing System for the Visually Impaired.

Figure 7. Success Rate of RFID Detection



The researcher conducted 10 trials for the system checking and assess the accuracy of the RFID detection system. The pie chart illustrates the percentage of successful and failed RFID detections. The system achieved a 90% success rate, meaning that most RFID cards and keychains were detected correctly. The 10% failure rate could be due to improper card positioning, distance issues, or interference from external factors. A high success rate indicates that the RFID reader is reliable and functions well for pedestrian use.

Figure 8. Response Time for RFID vs Keychain

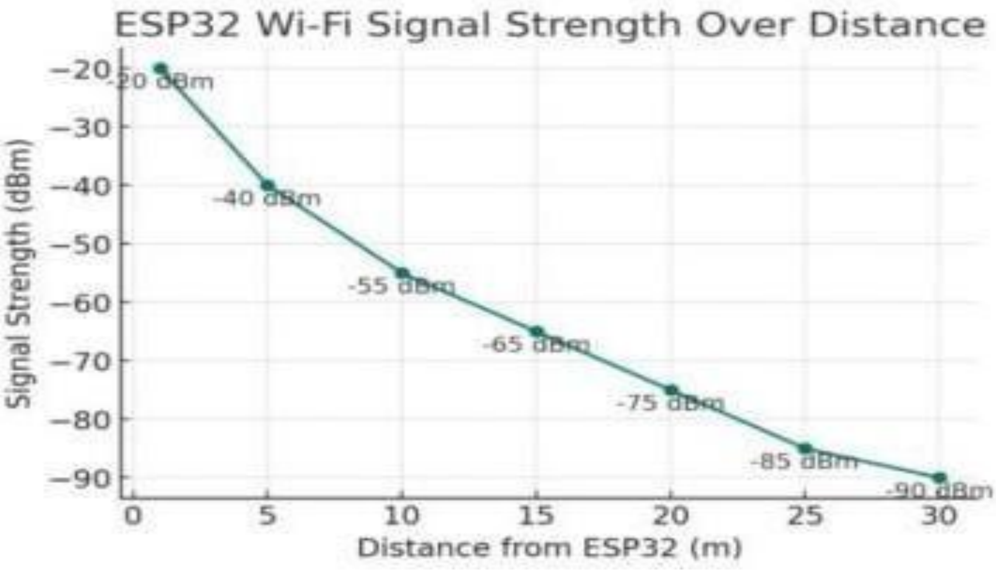


This bar graph compares the response time of the RFID Card and RFID Keychain. The RFID Card had a faster

ISSN No. 2454-6186 | DOI: 10.47772/IJRISS | Volume X Issue VI June 2026

response time (800 ms) compared to the RFID Keychain (950 ms), meaning the system processes the card slightly quicker. This difference could be due to variations in the RFID chip type or antenna strength. A lower response time ensures that visually impaired pedestrians experience minimal delay when activating the crossing system.

Figure 9. Strength of ESP32 in Wireless Communication



This figure shows the signal strength of ESP32 in Wireless Communication in terms of distance. The ESP32 Wi-Fi Signal Strength Line Graph shows that as the distance from the ESP32 increases, the signal strength decreases. The x-axis represents distance (in meters) and the y-axis shows signal strength (in dBm). At close range (e.g., -20 dBm), the connection is strong and stable. However, beyond 10 to 15 meters (-55 dBm to -65 dBm), the signal weakens but remains usable. After 20 meters (-75 dBm), the connection becomes unstable, and at -90 dBm, it fails. For reliable connection, the ESP32 should place between the range of 10 to 15 meters.

Table 1. Average Travel Time of Visually Impaired in Relevance to Distance

Distance (m)	Average time (s)	Walking speed (m/s)
4	7.84	0.51
6	9.73	0.62
7	11.22	0.62
8	12.58	0.64
10	15.53	0.64

The table shows that the average walking speed of visually impaired individuals ranged between 0.51 to 0.64 m/s. The time taken to cross increased proportionally with distance, ensuring a linear relationship between distance and crossing time. The findings suggest that a traffic light system should allocate at least 20 seconds for visually impaired pedestrians crossing a 8-10 meters lane safely.

Simulation Report

This test is not meant to mock individuals with disabilities; rather, it is part of a study on the speed at which visually impaired individuals cross a pedestrian lane. The images below are merely a simulation.

ISSN No. 2454-6186 | DOI: 10.47772/IJRISS | Volume X Issue VI June 2026

Figure 10. 4-meter pedestrian lane (near EARIST front gate)

Figure 11. 6-meter pedestrian lane (Crossing Nagtahan)

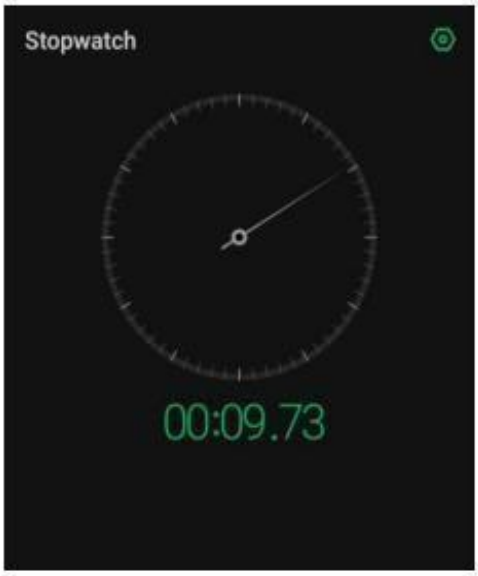


Figure 12. 7-meter pedestrian (near SAVEMORE, Nagtahan)

Figure 13. 8-meter pedestrian (in front of SAVEMORE, Nagtahan)

ISSN No. 2454-6186 | DOI: 10.47772/IJRISS | Volume X Issue VI June 2026

Figure 14. 10-meter pedestrian (near KFC @ Nagtahan Crossing)

CONCLUSION

The study concludes that the Smart Pedestrian Crossing System for the Visually Impaired is a viable and effective solution for enhancing pedestrian safety as it meets its objectives of securing visually impaired individual to cross the pedestrian safely. The integration of RFID technology and ESP32 microcontrollers allows visually impaired individuals to independently activate stoplights and cross streets safely. This study not only emphasizes the effectiveness but also indicates the need for improvements. Minor refinements in RFID accuracy and signal strength are recommended to enhance overall performance of the system. There are some key takeaways for this project.

The key take aways are as follows:

1. System’s Accuracy - The system successfully detects RFID credentials with high accuracy (90%), though minor improvements can be made to reduce the failure rate.

2. Response Time - The RFID card has a faster response time than the keychain, making it the preferred option for users.

3. Signal Strength - The ESP32 Wi-Fi module provides reliable connectivity up to 15 meters, beyond which signal strength declines significantly.

4. Crossing time - The distance and time are proportionate to each other, the longer the crossing time it will be more safer to visually impaired individual to cross the pedestrian.

With proper implementation and refinement, the system has the potential to increase mobility and independence for visually impaired pedestrians.

RECOMMENDATIONS

Based on the study's findings, the following recommendations are suggested to enhance the functionality and reliability of the Smart Pedestrian Crossing System:

1. Enhancing RFID Detection Accuracy
    - Positioning adjustments: Ensure the RFID reader is placed at an optimal height and angle for easy accessibility by visually impaired users.
    - Reducing interference: Test different RFID frequencies or shielding methods to minimize signal disruptions caused by environmental factors.
    - Improved hardware: Consider upgrading to a more sensitive RFID reader with greater detection range and faster processing speed.
2. Optimizing Response Time for Faster User Experience
    - Alternative RFID technologies: Research and test advanced RFID chips with lower latency to further reduce response time.
    - Firmware optimizations: Improve system coding to minimize delays in signal processing and stoplight activation.
3. Ensuring Stable ESP32 Wi-Fi Communication
    - Strategic placement: Install the ESP32 module within the recommended 10–15 meter range to maintain a strong and stable connection.
    - Signal boosters: Consider using Wi-Fi extenders or high-gain antennas to improve connectivity over longer distances.
    - Interference reduction: Avoid placing the ESP32 near large metal objects or other wireless devices that may cause signal degradation.
4. User Training and Awareness Programs
    - Conduct educational sessions for visually impaired individuals to familiarize them with the system’s functionality.
    - Provide audio or Braille instructions near pedestrian crossings to guide users on how to use the RFID system.
    - Collaborate with local authorities and disability advocacy groups to implement the system in high-traffic areas for increased accessibility.

REFERENCES

1. Bermudo, E. F. C., Nahiwan, P. A. D., Redoble, B. E. J. S., Reyes, B. C. J. M., Singzon, J. P. Y., Asaba,H. M., & Ong, C. Y. (2020). Developing an Assistive Device for the Visually Impaired Using Ultrasonic Sensors for Distance and Solidity Determination. De La Salle University Research Congress 2020. Retrieved from https://www.dlsu.edu.ph/wp-content/uploads/pdf/conferences/research-congressproceedings/2020/HCT-05.pdfTai, N. C., & Hu, S. C. (2017). Development of a natural user interface-based cyclist signaling vest. International Journal of Automation and Smart Technology, 7(4),

157–162. https://doi.org/10.5875/ausmt.v7i4.1342
2. Department of Social Welfare and Development (2019). DSWD provides assistive devices to persons with disability. Retrieved from https://www.dswd.gov.ph/dswd-providesassistive-devices-to-persons-with-disability/Usha, S., Karthik, M., Lalitha, R., Jothibasu, M., & Krishnamoorthy, T. (2021). Automatic turning ON/OFF bike indicator using offline GPS navigation system. IOP Conference Series: Materials Science and Engineering. https://doi.org/10.1088/1757-899X/1055/1/012032
3. National Council on Disability Affairs (2021). A Resolution Approving The Guidelines For The Provision Of Assistive Devices And Technology Services In The Philippines. Retrieved from https://ncda.gov.ph/a-resolution-approving-the-guidelines-for-theprovision-of-assistive-devices-and-technology-services-in-the-philippines-ncdaboard-resolution-no-4-s-2021/
4. National Council on Disability Affairs (2025). Disability Stats Summary per Region, Province, and City Municipality. Retrieved from https://ncda.gov.ph/disability-statssummary-per-region-province-and-citymunici/
5. Smith, J., Lee, K., & Johnson, H. (2020). Smart city solutions for visually impaired individuals: The role of RFID and audio guidance systems. IEEE Transactions on Smart Cities, 12(4), 101-112.

ABOUT THE AUTHORS

Herjed R. Flores is a computer engineering student of Eulogio “Amang” Rodriguez Institute of Science and Technology in the Philippines, with proficient knowledge in Embedded Systems, and Computer Network and Security.

Andre Axel M. Grageda acted as the Technical Documentation and Research Analyst of the project. A Bachelor of Science in Computer Engineering student at Eulogio "Amang" Rodriguez Institute of Science and Technology, he was responsible for preparing the project manuscript, organizing technical documentation, and ensuring clarity and consistency in the presentation of the study. He also assisted in data analysis, formatting, and compliance with academic writing standards.

Mark Ian Guillen is a computer engineering student of Eulogio “Amang” Rodriguez Institute of Science and Technology in the Philippines with proficient knowledge in Embedded Systems and Microprocessors.

Charles Miranda is a computer engineering student of Eulogio “Amang” Rodriguez Institute of Science and Technology in the Philippines, possessing proficient knowledge in embedded systems, microprocessors, and printed circuit board (PCB) design and simulation.

Gerrold M. Sillacay is a computer engineering student of Eulogio “Amang” Rodriguez Institute of Science and Technology in the Philippines, bringing a diverse technical background that includes computer programming, operating systems, network administration, and virtualization.

Engr. Minerva C. Zoleta, a Professional Computer Engineer, is a dedicated Computer Engineering Professor at the Eulogio “Amang” Rodriguez Institute of Science and Technology in the Philippines, specializing in Embedded Systems, Operating Systems, and Computer Network and Security. With a strong background in academia and industry. She has been instrumental in shaping the next generation of Engineers through innovative teaching methods and hands-on research. Engr. Zoleta holds a Master’s degree in Electrical Engineering major in Computer Engineering at Technological University of the Philippines, Manila and is pursuing her doctorate degree in Engineering with specialization in Computer Engineering AT Technological Institute of the Philippines. She has presented published research on topics such as Embedded System, IoT applications, and wireless communication international conferences and journals. Passionate about technology-driven solutions, she has led various projects integrating smart systems into real-world applications, contributing to the advancement of local and international engineering communities.

Albert Einstein is a student of Master of Science in Electrical Engineering major in Computer Engineering at Technological University of the Philippines. A Speaker, Trainer and Consultant with 18 Trainings in 5 countries. Chief Solutions Strategist of NPH Management and Consultancy and THINK Inside the Box. Speaker to Computer World, featured in Computer World Magazine. One of Fuji Xerox Philippines, foremost Authority in Solution Selling & Training DMS, Security & Cloud.