VISVESVARAYA TECHNOLOGICAL UNIVERSITY

Jnana Sangama, Belagavi – 590014.

An Internship Report

On

"IOT Theft Detection using Raspberry Pi"

Submitted in partial fulfillment of the requirement for the award of degree of

BACHELOR OF ENGINEERING

In

COMPUTER SCIENCE & ENGINEERING &
INFORMATION SCIENCE & ENGINEERING

By

Name: Chandana N **(USN: 1AP21CS013)**
Name: Chandana S **(USN:1AP21CS014)**
Name: Harshitha B P **(USN: 1AP21IS012)**

2024 - 2025

DEPARTMENT OF INFORMATION SCIENCE AND ENGINEERING

A P S COLLEGE OF ENGINEERING

Anantha Gnana Gangothri,
NH-209, Kanakapura Road, Somanahalli, Bengaluru-560116.

Evaluation Sheet

Title of the Project**: IoT Theft Detection using Raspberry Pi**

Name of the Students: Chandana N **(1AP21CS013)**
Chandana S **(1AP21CS014)**
Harshitha B P **(1AP21IS012)**

External Supervisor:

_____

Internal Supervisor:

_____

_____

Date:

Place:

Project Completion Certificate

I, Harshitha B P (Roll No: 1AP21IS012), hereby declare that the material presented in the Project Report titled **"IoT Theft Detection using Raspberry Pi"** represents original work carried out by me in the **Department of Computer Science and Engineering** at the **APS college of Engineering, Bangalore** during the tenure **2 October, 2024 – 12, December, 2024**.

With My signature, I certify that:

- I have not manipulated any of the data or results.
- I have not committed any plagiarism of intellectual property and have clearly indicated and referenced the contributions of others.
- I have explicitly acknowledged all collaborative research and discussions.
- I understand that any false claim will result in severe disciplinary action.
- I understand that the work may be screened for any form of academic misconduct.

Date: **Student Signature**:

In my capacity as the supervisor of the above-mentioned work, I certify that the work presented in this report was carried out under my supervision and is worthy of consideration for the requirements of the B.Tech. Internship Work.

Advisor's Name: **Guide Name: AKHIL SAI**

Advisor's Signature **Guide Signature**

Project Completion Certificate

I, Chandana N (Roll No: 1AP21CS013), hereby declare that the material presented in the Project Report titled **"IoT Theft Detection using Raspberry Pi"** represents original work carried out by me in the **Department of Computer Science and Engineering** at the **APS college of Engineering, Bangalore** during the tenure **2 October, 2024 – 12, December, 2024**.

With My signature, I certify that:

- I have not manipulated any of the data or results.
- I have not committed any plagiarism of intellectual property and have clearly indicated and referenced the contributions of others.
- I have explicitly acknowledged all collaborative research and discussions.
- I understand that any false claim will result in severe disciplinary action.
- I understand that the work may be screened for any form of academic misconduct.

Date: **Student Signature**:

In my capacity as the supervisor of the above-mentioned work, I certify that the work presented in this report was carried out under my supervision and is worthy of consideration for the requirements of the B.Tech. Internship Work.

Advisor's Name: **Guide Name: AKHIL SAI**

Advisor's Signature **Guide Signature**

Project Completion Certificate

I, Chandana S (Roll No: 1AP21CS014), hereby declare that the material presented in the Project Report titled **"IoT Theft Detection using Raspberry Pi"** represents original work carried out by me in the **Department of Computer Science and Engineering** at the **APS college of Engineering, Bangalore** during the tenure **2 October, 2024 – 12, December, 2024**.

With My signature, I certify that:

- I have not manipulated any of the data or results.
- I have not committed any plagiarism of intellectual property and have clearly indicated and referenced the contributions of others.
- I have explicitly acknowledged all collaborative research and discussions.
- I understand that any false claim will result in severe disciplinary action.
- I understand that the work may be screened for any form of academic misconduct.

Date: **Student Signature**:

In my capacity as the supervisor of the above-mentioned work, I certify that the work presented in this report was carried out under my supervision and is worthy of consideration for the requirements of the B.Tech. Internship Work.

Advisor's Name: **Guide Name: AKHIL SAI**

Advisor's Signature **Guide Signature**

Table of Contents:-

Abstract

The IoT-based theft detection system leverages advanced sensor technology and the Internet of Things (IoT) to provide an efficient and reliable security solution for homes, offices, and industrial spaces. The system incorporates ultrasonic sensors, light sensors, and temperature and humidity sensors, working in tandem with a Raspberry Pi to monitor environmental changes and detect potential security breaches. When anomalies such as unauthorized entry or tampering are detected, the system activates a buzzer for immediate alerts and sends real-time notifications to the user through a connected IoT platform. Data from sensors is uploaded to cloud services like ThingSpeak for continuous monitoring, enabling users to analyze security patterns and respond promptly to threats. With its ability to integrate seamlessly into existing security setups and its scalability for larger infrastructures, this system represents a significant step toward intelligent, automated theft prevention. The use of features like real-time data visualization, remote monitoring, and cloud-based analytics not only enhances its utility but also ensures adaptability to evolving security challenges. Designed to operate in a variety of environments, the system offers a cost-effective, user-friendly, and innovative approach to safeguarding assets and ensuring peace of mind.

CHAPTER 1

INTRODUCTION

In the modern era, security is a paramount concern for residential, commercial, and industrial spaces. Traditional security systems, such as manual surveillance and basic alarm systems, are often inadequate in providing real-time monitoring and alerts. To address these challenges, this project focuses on developing an IoT-based theft detection system using Raspberry Pi.

The system integrates multiple sensors, including a light sensor (LDR), ultrasonic sensor, and DHT11, to monitor environmental conditions and detect potential threats. By leveraging IoT technology, the data collected from these sensors is transmitted to the ThingSpeak cloud platform for real-time monitoring and analysis. Alerts are generated using a buzzer and LED, ensuring immediate action during an intrusion or theft attempt.

The project demonstrates how IoT can enhance security systems by making them more intelligent, automated, and accessible from anywhere. It not only provides a cost-effective solution but also ensures proactive measures to safeguard property and assets.

1.1 OBJECTIVE

The project aims to design an IoT-based theft detection system using light (LDR) and ultrasonic sensors for real-time monitoring and alerting. Sensor data is transmitted to the ThingSpeak cloud for remote access and analysis. The system activates visual and audible alerts to deter theft, offering a scalable, cost-effective security solution.

1.2 PROBLEM STATEMENT

- **Ultrasonic Sensor Problem**: Traditional security systems struggle with accurately detecting motion or presence, leading to false alarms or missed intrusions. Ultrasonic sensors can address this by providing more precise distance measurements, but they may suffer from environmental interference or limited range in certain settings.

- **Light Sensor Problem**: Existing systems often fail to detect subtle changes in lighting conditions that could indicate security breaches, such as unauthorized entry or movement. Light sensors offer a potential solution, but they may be affected by external light sources (e.g., streetlights) or not detect motion effectively in low-light environments.
- **Temperature and Humidity Sensor Problem**: Traditional security systems often overlook environmental factors like temperature fluctuations and humidity changes, which could indicate tampering, abnormal activity, or potential risks such as water leaks or mold. While temperature and humidity sensors can enhance detection, they may have limited sensitivity or accuracy in varying climates, leading to false readings or inconsistent performance in environments with extreme or fluctuating conditions.

CHAPTER 2
APPLICATION

The IoT-based theft detection system offers diverse applications across multiple domains by ensuring real-time monitoring, alerting, and data transmission to enhance security and prevent unauthorized access. Below are the detailed applications:

1. Residential Security

The system significantly improves home security by detecting intrusions, break-ins, or unauthorized activities. Sensors like the light sensor and ultrasonic sensor monitor the environment for anomalies such as unexpected motion or changes in lighting. Real-time alerts are sent to homeowners via a cloud platform, enabling them to act promptly even when they are away. The integration of IoT makes it a cost-effective and scalable solution, suitable for apartments, villas, and gated communities.

2. Commercial Establishments

In offices, shops, and warehouses, the system helps secure valuable assets and sensitive areas. Unauthorized access or proximity to restricted zones is detected through the ultrasonic sensor, while the buzzer and LED provide immediate alerts to deter intruders. Cloud connectivity ensures that security personnel or business owners can monitor the premises remotely, reducing the risk of theft and enhancing overall operational security.

3. Industrial Safety

Factories and industrial units often require monitoring of large spaces and restricted areas. The system can identify unauthorized personnel or detect any abnormal activity within the premises. With its capability to send data to a centralized cloud platform, the system provides real-time monitoring and ensures quick response to potential threats, minimizing the risk of theft or accidents.

4. Smart Cities

As part of a broader IoT network, the system plays a crucial role in smart city initiatives. It can be integrated with centralized monitoring systems to enhance public safety by securing public buildings, transportation hubs, and other sensitive areas. The use of real-time data transmission and cloud storage ensures efficient management of resources and rapid response to security breaches.

5. Educational Institutions

Schools, colleges, and universities can benefit from the system by securing areas like laboratories, libraries, and storerooms. The sensors ensure that restricted areas are protected from unauthorized access, while the alerts notify staff or security teams in case of any unusual activity. This enhances the overall safety of the institution and protects valuable equipment and resources.

6. Healthcare Facilities

Hospitals and clinics can deploy the system to monitor restricted zones such as pharmaceutical storage areas, operation theaters, and staff-only zones. The real-time detection of intrusions and cloud-based monitoring provides an added layer of security, ensuring compliance with safety standards and preventing unauthorized access to critical areas.

7. Retail and Hospitality

Retail stores and hotels can utilize the system to secure customer areas, storage spaces, and restricted zones. Unauthorized access to cash registers, safes, or sensitive sections is quickly identified, and immediate alerts are generated. This enhances the overall safety of the establishment, ensuring customer trust and operational efficiency.

By adapting to different environments and requirements, the system serves as a versatile, scalable, and efficient security solution, making it suitable for residential, commercial, industrial, and public spaces. The combination of real-time monitoring, remote access, and instant alerting ensures a proactive approach to theft prevention and enhanced security**.**

CHAPTER 3

SYSTEM REQUIREMENTS

3.1 HARDWARE COMPONENTS:

1. Raspberry Pi

The Raspberry Pi serves as the core controller of the system. It processes the data from various sensors, triggers alerts, and communicates with the cloud platform (ThingSpeak). The Raspberry Pi has multiple GPIO (General Purpose Input/Output) pins to connect and control sensors and output devices like the buzzer and LED. It also provides Wi-Fi connectivity for cloud communication.

Fig 3.1 Raspberry pi

2. DHT11 Sensor

The DHT11 is a low-cost sensor used to measure temperature and humidity. It provides digital output, which is processed by the Raspberry Pi. The sensor is critical for monitoring the environmental conditions of the area being monitored and helps in detecting abnormal conditions that could indicate a security risk.

Fig 3.2 DHT11 Sensor(Temperature and Humidity sensor)

3. Ultrasonic Sensor (HC-SR04)

The HC-SR04 ultrasonic sensor is used to measure the distance between the sensor and any object in front of it. It works by emitting ultrasonic waves and measuring the time it takes for them to return after hitting an object. The distance is then calculated, which can help in detecting any movement or objects that may indicate an intrusion or theft.

Fig 3.3 Ultrasonic Sensor (HC-SR04)

4. Light Sensor (LDR)

The Light Dependent Resistor (LDR) is used to detect changes in ambient light levels. When the light intensity changes, such as when an intruder blocks the light, the resistance of the LDR changes, allowing the Raspberry Pi to detect such variations. This is useful for identifying unusual conditions, such as a break-in during the night or the turning off of lights when unauthorized activity occurs.

Fig 3.4 Light Sensor (LDR)

5. Buzzer

The buzzer is used as an audible alert when an intrusion is detected. It is powered by the Raspberry Pi and is triggered when the system detects abnormal conditions from the sensors, such as movement detected by the ultrasonic sensor or a change in lighting from the LDR sensor. The buzzer sounds to notify individuals nearby about the possible security breach.

Fig 3.5 Buzzer

6. LED

The LED is used as a visual alert mechanism. It lights up in response to specific sensor inputs, such as detecting an object within the range of the ultrasonic sensor or triggering the system due to a light level change. The LED acts as a visual deterrent, signaling a security event.

Fig 3.6 LED

7. Power Supply

The Raspberry Pi and connected components require a constant power supply to operate effectively. The Raspberry Pi typically uses a 5V power supply through a micro-USB port, while other components like the sensors and buzzer may be powered via the Raspberry Pi's GPIO pins or external sources.

Fig 3.7 Power supply

8. Wi-Fi Module (Integrated in Raspberry Pi)

The Raspberry Pi has an integrated Wi-Fi module that enables wireless internet connectivity. This is essential for uploading sensor data to the ThingSpeak cloud platform in real-time and for remote monitoring through the platform.

Fig 3.8 Wi-Fi Module (Integrated in Raspberry Pi)

3.2 SOFTWARE COMPONENTS:

1. Raspberry Pi OS

Raspberry Pi OS (formerly Raspbian) is the operating system that runs on the Raspberry Pi. It is based on Debian Linux and provides the necessary environment to interface with hardware, manage sensor inputs, and run Python scripts to control the security system.

2. Python Programming Language

Python is the primary programming language used to control the hardware components and communicate with ThingSpeak. Python scripts read sensor data, process inputs, control outputs (like the buzzer and LED), and upload data to the ThingSpeak platform. The extensive libraries available in Python make it ideal for this IoT-based project, as it easily integrates with the sensors and GPIO pins on the Raspberry Pi.

3. Adafruit Libraries

The Adafruit libraries are used for interfacing with sensors like the DHT11 temperature and humidity sensor. These libraries simplify the process of collecting data from sensors and help in managing the GPIO pins for other hardware components.

4. ThingSpeak Cloud Platform

ThingSpeak is an open-source IoT platform that allows users to collect, analyze, and visualize data from IoT devices. The platform enables the Raspberry Pi to send sensor data to the cloud, where it can be accessed remotely through a web interface. ThingSpeak also provides API keys for easy integration with IoT devices, making it easier to transmit sensor data.

5. Requests Library (Python)

The Requests library in Python is used for sending HTTP requests to the ThingSpeak API to upload sensor data. This library handles communication between the Raspberry Pi and the ThingSpeak cloud, enabling seamless data transfer and real-time monitoring.

6. GPIO Library (Python)

The RPi.GPIO library in Python is used to manage the GPIO pins on the Raspberry Pi. It allows the user to control the various sensors, the buzzer, and the LED. This library enables the Raspberry Pi to interact with physical components like sensors, motors, and alarms.

7. Web Interface (HTML/CSS)

The web interface is used to display the sensor data in an organized format. It is built using HTML for the structure, CSS for styling, and JavaScript (in some cases) for real-time updates. The data fetched from the ThingSpeak API is displayed on a web page for monitoring and analysis, providing users with an accessible and easy-to-use interface.

8. JavaScript (for Real-Time Data Fetching)

JavaScript is used to dynamically fetch and update the sensor data on the web page. It makes asynchronous requests to the ThingSpeak API to retrieve the latest sensor data and update the webpage without requiring a page reload, offering a seamless user experience for monitoring real-time security dat

CHAPTER 4

FLOWCHART

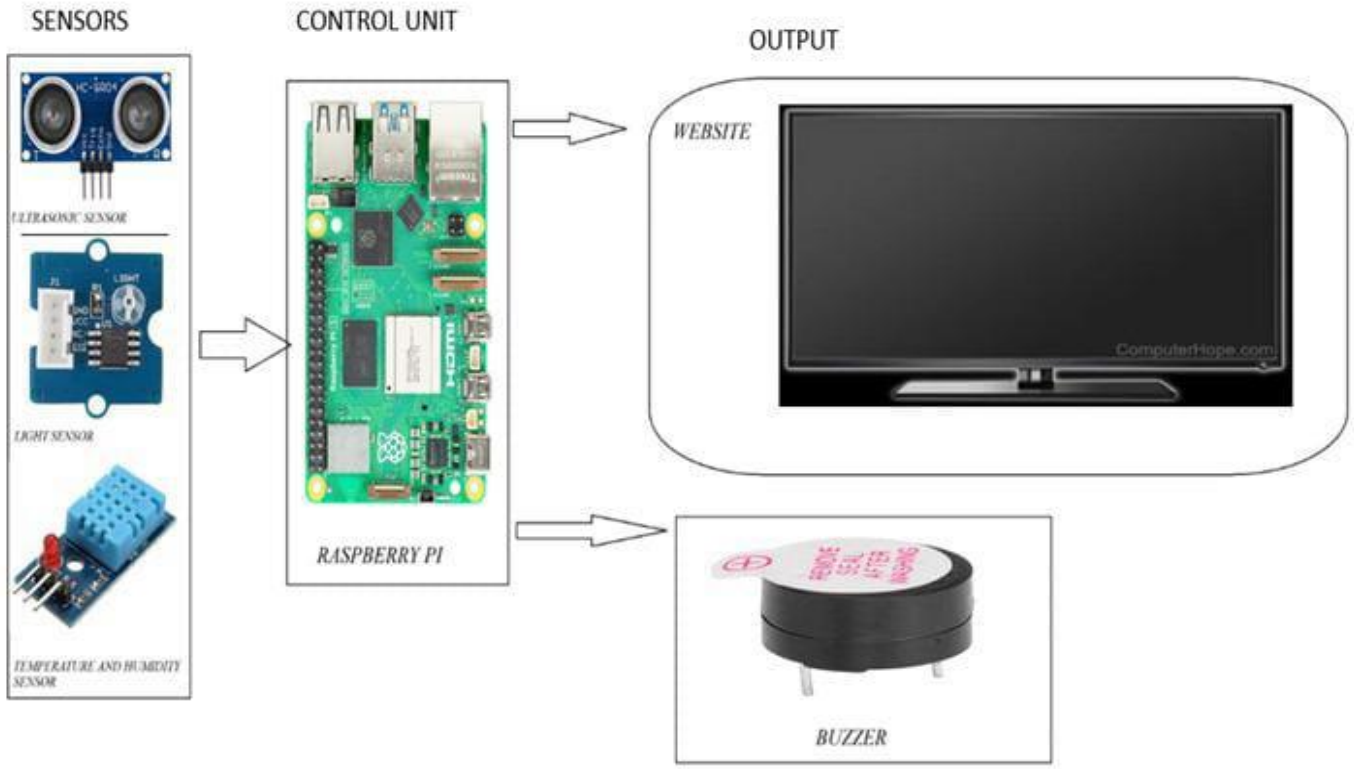


Fig:4.1 Flowchart

The system consists of sensors, a control unit, and output devices, working together for data collection and monitoring. The sensors, including an ultrasonic sensor, light sensor, and temperature/humidity sensor, gather environmental data. The data is sent to a Raspberry Pi, which acts as the control unit, processing the inputs. Based on this data, the system triggers outputs such as a buzzer for sound alerts or displays information on a website accessed through a screen. This

CHAPTER 5

CONCLUSION

The IoT-based theft detection system represents a significant step toward enhancing security through the integration of various sensors and real-time data processing. By utilizing components such as the Raspberry Pi, ultrasonic sensors, light sensors, and environmental monitoring tools like the DHT11, the system is capable of detecting unauthorized access and intrusions, providing both immediate alerts and real-time monitoring. The cloud integration with ThingSpeak ensures that the system is scalable and accessible from anywhere, enabling users to monitor data remotely and receive notifications in case of any anomalies. The system's ability to combine sensors for environmental monitoring, object detection, and real-time alerting offers a comprehensive security solution that is both cost-effective and highly efficient. Additionally, the use of open-source technologies, such as Python for programming and ThingSpeak for cloud integration, ensures that the system remains flexible and adaptable to various environments, whether residential, commercial, or industrial. The application of this system extends to numerous sectors, from home security to industrial safety, providing robust protection against theft and unauthorized access. The combination of hardware and software components facilitates seamless operation, making it an ideal choice for modern security requirements. As the need for advanced security solutions continues to grow, this IoT-based system stands out as a forward-thinking solution that integrates real-time data monitoring with intelligent decision-making, ensuring enhanced safety and security for a wide range of applications.

CHAPTER 6

FUTURE WORK

The future of the IoT-based theft detection system holds great potential for enhancing its functionality and efficiency. One key area for improvement is integrating additional sensors like motion detectors, cameras with image recognition, or RFID technology, which would allow for better identification of individuals and tracking of objects, reducing false alarms. Machine learning algorithms could also be added to analyze patterns in sensor data, helping the system distinguish between normal and abnormal activities more effectively. Expanding connectivity by integrating the system with other IoT devices or home automation systems would create a more comprehensive security solution. Enhancing cloud-based data analytics with advanced visualization tools could provide deeper insights into security trends, allowing users to respond proactively. The system could also be miniaturized and optimized for various environments, from homes to industrial areas. Future developments may include incorporating emerging technologies like 5G for faster data transmission and edge computing for real-time processing. Lastly, implementing blockchain technology could improve data security and transparency. These advancements will enable the system to provide more robust and reliable security solutions.

There are several potential improvements that could enhance the performance and functionality of the system:

- Sensor Range Expansion: Consider using more advanced ultrasonic sensors with longer range or adding additional sensors for broader coverage.
- Wireless Connectivity: Incorporating wireless communication (e.g., Wi-Fi or Bluetooth) for remote monitoring without the need for a constant internet connection could improve flexibility.
- Mobile Application: Developing a mobile app to monitor the data in real-time could make the system more user-friendly and accessible.
- AI-based Detection: Implementing machine learning algorithms could enhance the system's ability to identify potential intruders based on patterns and anomalies in sensor data, leading to more intelligent alerts

CHAPTER 7

APPENDIX

➢ Hardware Components:

- **Raspberry Pi 4 Model B**: Central processing unit, manages sensors and communicates with ThingSpeak.
- **DHT11 Temperature and Humidity Sensor**: Monitors environmental conditions (temperature and humidity).
- **HC-SR04 Ultrasonic Sensor**: Detects movement and measures distance to identify potential intruders.
- **LDR (Light Dependent Resistor)**: Senses light conditions, triggering alerts if unusual darkness is detected.
- **Active Buzzer**: Provides audible alerts when predefined conditions are met (e.g., movement or light change).
- **LED**: Acts as a visual alert when an object is detected by the ultrasonic sensor.

➢ Software Components:

- **Raspberry Pi OS**: Operating system that supports Python and system operations.
- **Python**: Programming language for data collection, processing, and communication.
- **Adafruit Sensor Libraries**: Provides functions for reading data from the sensors and controlling GPIO pins.
- **ThingSpeak API**: Cloud platform for uploading and analyzing sensor data.
- **Requests Library**: Simplifies HTTP communication between the Raspberry Pi and ThingSpeak.

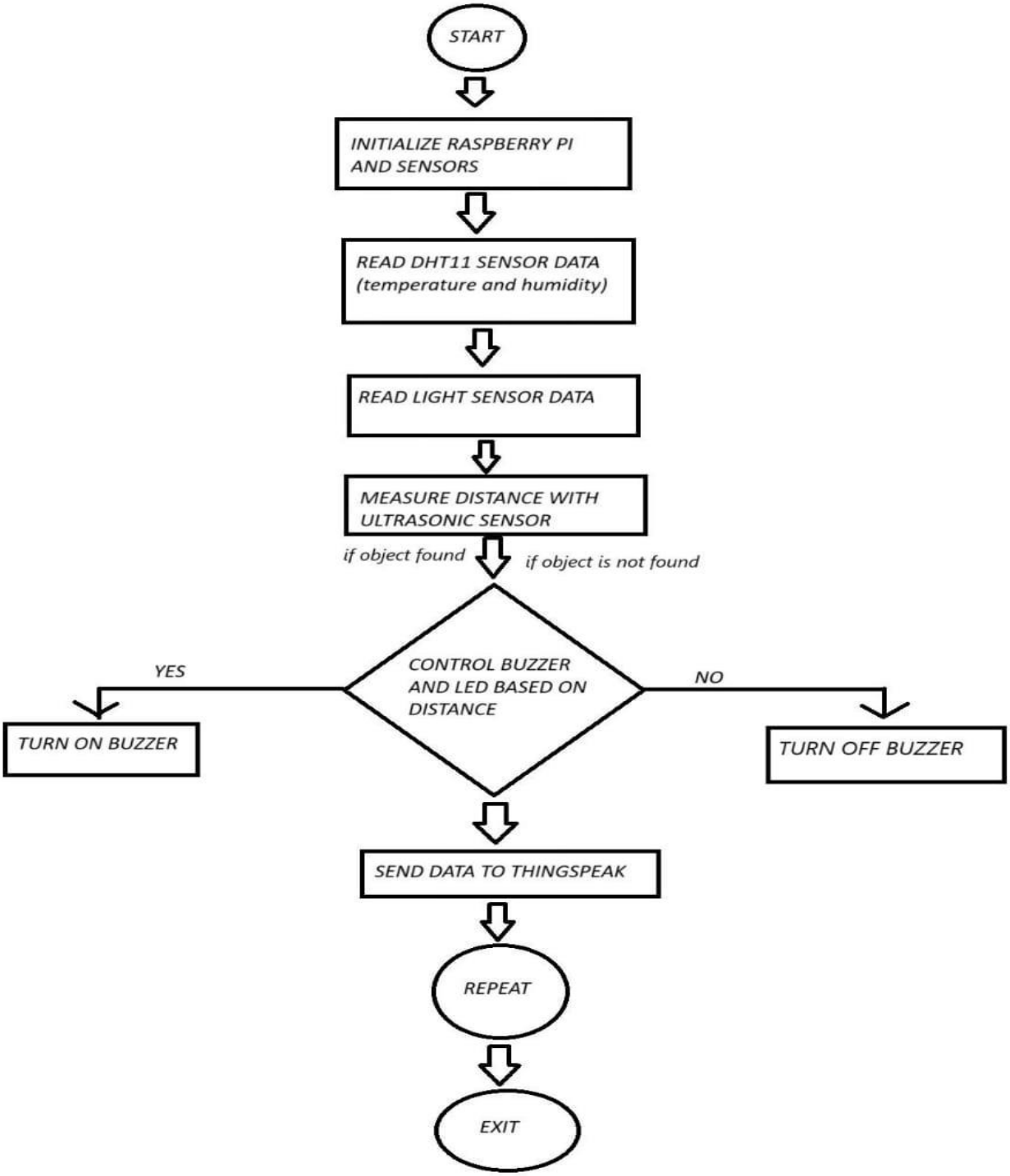
➢ System Workflow and Operations:

- Data Collection: Sensors (DHT11, ultrasonic, LDR) continuously gather data on temperature, humidity, distance, and light levels.
- Data Processing: Raspberry Pi processes sensor data to determine actions, such as triggering alerts when unusual conditions are detected.
- Data Transmission: Processed data (temperature, humidity, light status, and distance) is sent to ThingSpeak for remote monitoring via HTTP requests.

- **Common Issues and Solutions:**
  - **Sensor Data Not Accurate**: Check wiring and ensure sensor pins are correctly connected to the Raspberry Pi and match the script.
  - **ThingSpeak Data Not Updating**: Verify internet connection, correct API key, channel ID, and ensure ThingSpeak is online.
  - **Buzzer or LED Not Responding**: Confirm GPIO pins in the script match the physical pins; test components individually for functionality.
  - **System Not Triggering Alerts**: Ensure ultrasonic sensor range is calibrated, LDR sensitivity is set correctly, and alert conditions are met in the code.
- **System Limitations and Considerations:**
  - **Range of Sensors**: Ultrasonic sensor has a limited range (2 cm to 4 meters), which may not be suitable for large or fast-moving areas.
  - **Environmental Factors**: DHT11 sensor accuracy can be affected by extreme environmental conditions; best used indoors.
  - **Power Dependency**: Continuous power is required for the Raspberry Pi and sensors; power interruptions may cause malfunction or data loss.

CHAPTER 8

PSEUDO CODE



Here is a pseudo code for the IoT-based theft detection system that includes all the key functionalities such as sensor readings, data processing, and sending the data to ThingSpeak, along with triggering alarms based on specific conditions:

```
BEGIN

  // Initialize sensors and hardware components
  Initialize DHT11 sensor
  Initialize ultrasonic sensor (TRIG, ECHO)
  Initialize LDR sensor
  Initialize Buzzer and LED
  Initialize ThingSpeak API for data transmission

  // Main loop
  WHILE system is running
    // Read Temperature and Humidity from DHT11 sensor
    temperature, humidity = read DHT11 sensor

    IF temperature and humidity are valid THEN
      PRINT "Temperature: " + temperature + "°C"
      PRINT "Humidity: " + humidity + "%"
    ELSE
      PRINT "Failed to retrieve temperature and humidity data"
    END IF

    // Read Light Status from LDR sensor
    light_status = read LDR sensor

    IF light_status == 0 THEN
      PRINT "Dark detected: Turning Buzzer ON"
      TURN ON Buzzer
    ELSE
      PRINT "Normal light condition: Turning Buzzer OFF"
      TURN OFF Buzzer
    END IF

    // Measure Distance using Ultrasonic sensor
    distance = measure_distance(TRIG, ECHO)
    PRINT "Distance: " + distance + " cm"

    // Send sensor data to ThingSpeak
    payload = {
      "api_key": API_KEY,
      "field1": humidity,
      "field2": temperature,
      "field3": light_status,
      "field4": distance
```

```
    }
    response = send_data_to_thingspeak(payload)

    IF response is successful THEN
       PRINT "Data sent to ThingSpeak"
    ELSE
       PRINT "Failed to send data to ThingSpeak"
    END IF

    // Check for object detection with ultrasonic sensor
    IF distance is less than 50 cm AND greater than 2 cm THEN
       PRINT "Object detected! Activating LED and Buzzer"
       TURN ON Buzzer
       TURN ON LED
       WAIT for 0.5 seconds
       TURN OFF Buzzer
       TURN OFF LED
    ELSE
       TURN OFF Buzzer
       TURN OFF LED
    END IF

    WAIT for 0.5 seconds
  END WHILE

  // Clean up on exit
  ON KeyboardInterrupt
    CLEAN UP GPIO pins
    PRINT "Program stopped by user"
  END ON

END
```

Explanation:

1. Sensor Initialization: Initializes the sensors for temperature/humidity, ultrasonic, and light.

2. Main Loop: Continuously reads sensor data, processes it, and sends it to ThingSpeak. It also triggers the buzzer and LED based on conditions (e.g., light status or object detection).

3. ThingSpeak Integration: Sends the collected sensor data (temperature, humidity, light status, and distance) to ThingSpeak for real-time monitoring.

4. Object Detection: The system checks if an object is within a certain range (2 cm to 50 cm) and triggers an alert (buzzer and LED) if it detects an object.

5. Exit Handling: Ensures proper cleanup of GPIO pins when the user stops the program.

REFERENCES

- **https://www.youtube.comYouTube.**
- **https://www.google.co.in.**
- **https://nevonprojects.com/iot-theft-detection-using-raspberry-pi/.**
- **https://youtu.be/sG8saNEqoQE?si=nr4DLTWlaMIuSkhH.**
- **https://www.electrosal.com/product/iot-theft-detection-using-raspberry-pi/.**