AMRITA VISHWA VIDYAPEETHAM | School of Computing

PROJECT REPORT

19CSE446 – Internet of Things (IoT)

Amrita Vishwa Vidyapeetham Chennai – 601103, Tamil Nadu, India.

March 2024

Integrating Home Automation: Raspberry Pi, Grove Pi, and Telegram Bot

I. Abstract:

The Internet of Things (IoT) has revolutionized how we interact with the physical world around us. It has transformed everyday objects into smart devices capable of collecting data, communicating with each other, and responding to user commands. This project explores a practical IoT application that merges remote control, sensor data monitoring, and security features. It utilizes a Raspberry Pi as the central hub to control an LED and monitor for intruders. Users can send text or voice commands through a Telegram bot to turn the LED on or off. Additionally, an ultrasonic sensor detects the presence of intruders within a specific range and sends an alert notification via a designated Telegram chat. This project demonstrates the potential of using readily available tools and platforms, like Raspberry Pi and Telegram, to create a basic yet functional IoT system for remote control and security purposes.

The integration of Raspberry Pi, Grove Pi, and Telegram bot not only provides a seamless user experience but also showcases the versatility of these technologies in practical applications. By leveraging the extensibility of the Raspberry Pi ecosystem and the ease of integration offered by Grove Pi, developers can rapidly prototype and deploy innovative IoT solutions. Moreover, the use of Telegram as a communication interface enhances accessibility, allowing users to interact with their home automation system from anywhere with internet connectivity. This project serves as a foundation for further exploration and development in the realm of home automation and IoT integration.

Furthermore, this project fosters exploration and experimentation in the realm of DIY home automation, empowering individuals to customize and expand upon the system according to their specific needs and preferences. By providing a comprehensive yet accessible framework for integrating hardware components, communication protocols, and user interfaces, it encourages innovation and collaboration within the maker community. As technology continues to evolve, the potential for creating sophisticated and interconnected IoT ecosystems within our homes becomes increasingly attainable, paving the way for a smarter, more efficient, and secure living environment.

Utilizing Raspberry Pi and Grove Pi as the foundational hardware components ensures a seamless integration of sensors and actuators, enabling a robust IoT infrastructure. The Raspberry Pi serves as the brains of the operation, orchestrating communication between the various peripherals connected via the Grove Pi. This modular approach simplifies hardware setup and expansion, allowing users to incorporate additional sensors or devices with minimal effort. Furthermore, the Grove Pi's compatibility with a wide range of Grove modules offers flexibility in selecting sensors tailored to specific requirements, whether it's temperature monitoring, motion detection, or ambient light sensing. This versatility empowers users to tailor their home automation system to suit diverse needs, from basic functionality to advanced monitoring and control capabilities.

The integration of Telegram as the communication interface enhances the accessibility and convenience of the home automation system, enabling users to interact with their devices using familiar messaging platforms. By leveraging Telegram's extensive bot API, users can issue

commands to control the LED, receive status updates, and receive alerts about potential intruders seamlessly. This intuitive interface eliminates the need for specialized applications or complex setups, making the system accessible to users of all technical backgrounds. Moreover, Telegram's end-to-end encryption ensures secure communication, safeguarding sensitive information exchanged between the user and the home automation system. This combination of ease of use, security, and ubiquity positions Telegram as an ideal platform for enabling remote monitoring and control of IoT devices, further enhancing the practicality and usability of the integrated solution.

II. Introduction:

The rapid growth of IoT devices has fostered innovative solutions for automation and remote control across various sectors. From smart homes with lights and thermostats controllable through voice commands to industrial automation monitoring factory equipment and environmental monitoring systems tracking air quality, IoT applications are transforming the way we interact with our surroundings. This project delves into a fundamental IoT application showcasing real-world functionalities. It provides users with a user-friendly interface (Telegram) to remotely control a physical device (LED) and receive security notifications. This project serves as a stepping stone for understanding how simple hardware components and readily available platforms can be combined to create practical IoT solutions.

In the context of home automation, this project can be a building block for more complex systems. Imagine controlling lights in different rooms or appliances like fans or coffee makers using voice commands through a familiar platform like Telegram. Additionally, the intruder detection aspect of this project can be a starting point for a more comprehensive home security system. By incorporating additional sensors (e.g., motion sensors, door/window sensors) and connecting them to the Raspberry Pi, users can create a more robust security solution for their homes, receiving alerts for unauthorized entry or movement within the monitored area. Moreover, the integration of Raspberry Pi, Grove Pi, and Telegram bot not only provides practical functionalities but also promotes educational opportunities. As IoT continues to permeate various aspects of daily life, understanding the underlying technologies and their integration becomes increasingly valuable. This project offers a hands-on approach for enthusiasts, hobbyists, and students to learn about IoT concepts, hardware interfacing, programming, and communication protocols in a tangible and engaging manner. Thus, it not only addresses immediate needs for home automation and security but also contributes to the broader goal of fostering technological literacy and innovation in the IoT domain.

Furthermore, this project contributes to the democratization of IoT development by utilizing open-source hardware and software tools. The Raspberry Pi, with its low cost and extensive community support, democratizes access to computing power, enabling individuals from diverse backgrounds to participate in the creation of IoT solutions. Similarly, Telegram's open API allows developers to easily integrate messaging capabilities into their projects, offering a flexible platform for communication between users and IoT devices. By leveraging these accessible tools, this project empowers users to take ownership of their home automation and security needs, facilitating greater autonomy and customization in IoT deployment.

In addition to its practical applications in home automation and security, this project also highlights the importance of data privacy and security in the IoT ecosystem. As more devices become interconnected, the exchange of sensitive information increases, raising concerns about data

breaches and unauthorized access. By implementing secure communication protocols and encryption techniques, this project exemplifies best practices for safeguarding user data and maintaining the integrity of IoT systems. Moreover, it underscores the need for ongoing vigilance and updates to address emerging threats, reinforcing the importance of cybersecurity awareness in the era of pervasive connectivity.

Furthermore, the scalability and adaptability of this project make it conducive to customization and expansion based on specific user requirements and preferences. Whether it's integrating additional sensors for environmental monitoring, enhancing the user interface with advanced features, or incorporating machine learning algorithms for predictive analysis, the modular design of this IoT solution facilitates iterative development and innovation. As technology evolves and new functionalities emerge, users can leverage the foundation established by this project to explore novel applications and tailor the system to suit evolving needs. This adaptability not only enhances the longevity of the IoT solution but also encourages continuous experimentation and learning within the community, fostering a culture of innovation and collaboration in the pursuit of smarter, more connected environments.

III. Project Description:

This project establishes a Telegram bot that allows users to control an LED remotely. Users can turn the LED on or off by sending text commands like "on" or "off" through the Telegram chat interface. To enhance user experience, the system incorporates voice recognition capabilities. Users can send voice messages through Telegram, which are then downloaded and converted into text commands using speech recognition tools. This enables voice-based control of the LED, offering greater flexibility and accessibility compared to text commands alone.

Furthermore, the project integrates an ultrasonic sensor for intruder detection. The sensor continuously monitors the surrounding area, emitting and receiving ultrasonic waves to detect the presence and distance of objects within its range. If an object enters the designated range (indicating a potential intruder), the system triggers an alert message. This message is sent to a specific Telegram chat, notifying the user of a potential security breach through their chosen messaging platform. This notification feature provides a basic security layer, informing users of unauthorized presence in the monitored area. Moreover, the project emphasizes user privacy and data security by employing encryption protocols for communication between the Raspberry Pi and the Telegram servers. This ensures that user commands and notifications remain confidential and protected from unauthorized access. Additionally, the project encourages users to actively engage in the development process by providing open-source code and documentation, fostering a collaborative environment for learning and innovation within the IoT community. By promoting transparency and empowering users to understand and modify the underlying system, this project contributes to the democratization of IoT technology and promotes responsible DIY IoT development practices.

Furthermore, the project offers flexibility in hardware implementation, allowing users to adapt the system to their specific requirements and environments. For instance, while the current setup utilizes a Raspberry Pi and Grove Pi for interfacing with the LED and ultrasonic sensor, alternative hardware configurations can be easily integrated based on availability and desired functionalities. Whether utilizing different microcontrollers, sensors, or communication modules, users have the freedom to customize the system according to their preferences and constraints. This

modularity not only enhances versatility but also encourages experimentation and exploration, empowering users to tailor the IoT solution to diverse use cases and scenarios.

Additionally, the project facilitates seamless integration with existing home automation ecosystems through its open-source architecture and interoperability with popular platforms and protocols. By leveraging standard communication protocols such as MQTT (Message Queuing Telemetry Transport) and APIs (Application Programming Interfaces), users can extend the functionality of the system to interact with other IoT devices and services. This interoperability enables the creation of comprehensive home automation solutions where the LED control and intruder detection capabilities seamlessly integrate with lighting systems, security cameras, and smart assistants. By fostering compatibility and interoperability, the project promotes a unified approach to home automation, enhancing user convenience and expanding the possibilities for interconnected smart environments.

IV. Methodology:

This project utilizes a Raspberry Pi as the central processing unit and leverages various components and software to achieve remote control and intruder detection functionalities. Here's a breakdown of the methodology:

Hardware Components:

- *Raspberry Pi:* Acts as the central processing unit, coordinating all functionalities.
- *GrovePi Base Kit (or compatible modules):*
    - LED Module: Enables the Raspberry Pi to control the on/off state of the LED.
    - Ultrasonic Sensor Module: Detects the presence and distance of objects within its range by transmitting and receiving ultrasonic waves.
- *Jumper Wires:* Facilitate connections between the Raspberry Pi, GrovePi modules, and other components.
- *Power Supply for Raspberry Pi:* Provides the necessary voltage and current to operate the Raspberry Pi.

Software Components:

- Operating System: Raspberry Pi OS (or compatible) is installed on the Raspberry Pi to provide a platform for running applications.
- Python Programming Language: Used to develop the program logic for controlling the LED, processing sensor data, and interacting with the Telegram bot.
- Telegram Bot API: Enables communication between the user and the system through a Telegram bot.
- Speech Recognition Library (optional): If voice control is desired, a library like SpeechRecognition can be used to convert voice messages from Telegram into text commands.

System Operation:

- Setup:
  - The Raspberry Pi is set up with the chosen operating system.
  - GrovePi modules are connected to the Raspberry Pi and the corresponding physical components (LED and ultrasonic sensor) using jumper wires.
  - The Telegram bot is created using the Telegram Bot API.
- User Interaction:
  - Users interact with the system through a Telegram chat interface.
  - Text commands ("on" or "off") can be sent to turn the LED on or off.
  - Voice messages can be sent, downloaded by the system, and converted into text commands using speech recognition software.
- Processing and Control:
  - The Raspberry Pi program receives commands (text or voice commands) from the Telegram bot.
  - For LED control commands, the program sends signals to the LED module via the GrovePi interface, turning the LED on or off based on the command.
  - The ultrasonic sensor continuously monitors the surrounding area.
- Intruder Detection and Notification:
  - If the ultrasonic sensor detects an object within a predefined range (indicating a potential intruder), the program triggers an alert message.
  - The program interacts with the Telegram bot API to send the alert message to a specific Telegram chat, notifying the user of a potential security breach.

V. Architecture Diagram:

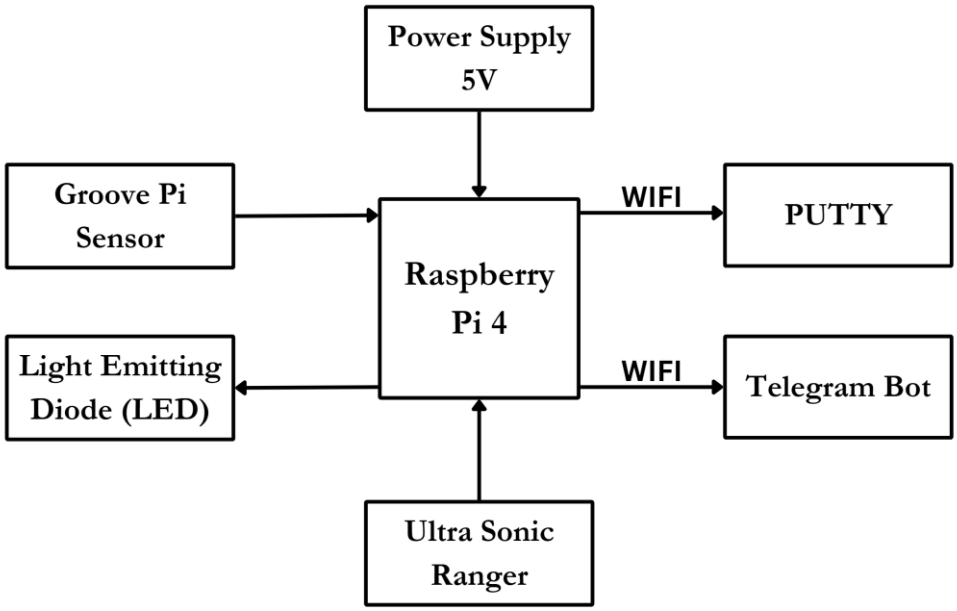

Fig: Basic Architecture diagram for understanding

VI. Components used In IoT Project:

- Raspberry Pi: A credit card-sized computer functioning as the central processing unit for the project. It executes the program logic, controls the LED, and interacts with the Telegram bot and ultrasonic sensor.

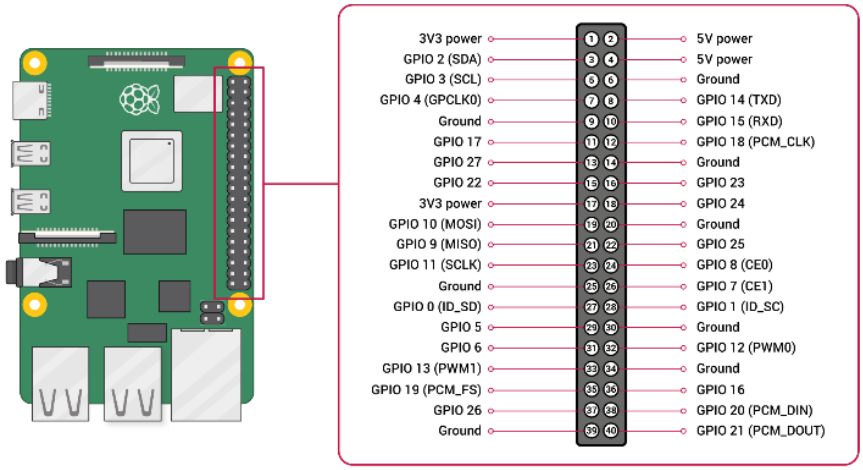

Fig: Raspberry Pi

- GrovePi Base Kit (or compatible modules): A collection of modules designed for simplified interfacing between the Raspberry Pi and various electronic components. This project utilizes two GrovePi modules:
  - LED Module: This module allows the Raspberry Pi to control the on/off state of the LED.
  - Ultrasonic Sensor Module: This module transmits and receives ultrasonic waves to detect the presence and distance of objects within its range.

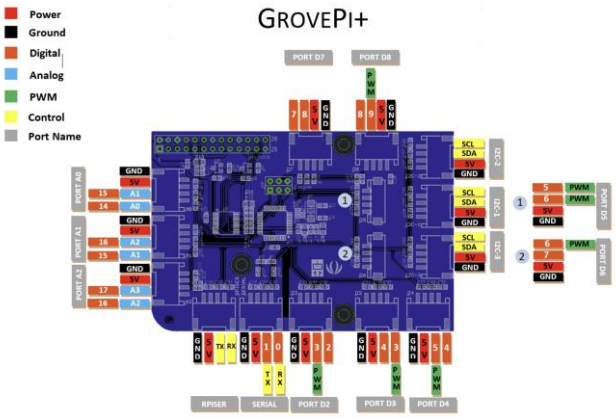

Fig: Port diagram of GrovePi

Fig: Grove LED v2

Fig: Ultrasonic Ranger

- Jumper Wires: These wires facilitate connections between the Raspberry Pi, GrovePi modules, and other components.

Fig: GrovePi Connectors

- Power Supply for Raspberry Pi: A dedicated power supply provides the necessary voltage and current to operate the Raspberry Pi.

VII. Source Code:

```
import os #importing required libraries
import time
import requests
import telepot
import grovepi
from telepot.loop import MessageLoop
from grovepi import *
import speech_recognition as sr
from pydub import AudioSegment

os.environ["GOOGLE_APPLICATION_CREDENTIALS"] =
"/home/pi/Downloads/iot-bot-418210-33742c3eceb3.json" #accessing
google cloud speech recognition API

led = 2  # Connect LED GrovePi module on D2
ultrasonic_ranger = 4  # Connect Ultrasonic Ranger to digital port D4

# LED White
pinMode(led, "OUTPUT")
time.sleep(1)
digitalWrite(led, 0)  # LED off initially
```

```
def send_telegram_message(message, chat_id): #function to send a
message in telegram chat
    telegram_bot.sendMessage(chat_id, message)

def process_voice_message(audio_file): #function to process the voice
file
    recognizer = sr.Recognizer()

    with sr.AudioFile(audio_file) as source:
        audio_data = recognizer.record(source)

    try:
        command = recognizer.recognize_google(audio_data).lower()
    except sr.UnknownValueError:
        command = ""
    except sr.RequestError as e:
        print("Could not request results from Google Speech
Recognition service; {0}".format(e))
        command = ""
    return command

def download_voice_message(file_id): #function to download audio file
from telegram bot
    token = '6614083923:AAEpJXrFT3RHG7gX7NiSHEEfzYSOxZnO05s'  # Your
bot's API token
    url =
f'https://api.telegram.org/bot{token}/getFile?file_id={file_id}'
    response = requests.get(url)
    file_path = response.json()['result']['file_path']
    file_url =
f'https://api.telegram.org/file/bot{token}/{file_path}'
    audio_file_ogg = f'/tmp/{file_id}.ogg'
    audio_file_wav = f'/tmp/{file_id}.wav'
    with open(audio_file_ogg, 'wb') as f:
        f.write(requests.get(file_url).content)
    audio = AudioSegment.from_ogg(audio_file_ogg)
    audio.export(audio_file_wav, format="wav")
    return audio_file_wav

def detect_intruder(): #function to take readings from ultrasonic
sensor
    try:
        #print("Running sensor")
        distance = grovepi.ultrasonicRead(ultrasonic_ranger)
        #if distance < 20:  # Adjust threshold as needed
           # print('Intruder detected at Distance:%d' %distance)
        return distance
        #else:
         #   return False
    except Exception as e:
```

```
        print("Error detecting intruder", e)
        return 100

def action(msg): #main action function
    chat_id = msg['chat']['id']
    content_type, chat_type, chat_id = telepot.glance(msg)

    if content_type == 'text':
        command = msg['text'].lower()
        print('Received text command: %s' % command)

        if "on" in command:
            message = "Turned on LED"
            digitalWrite(led, 1)
            send_telegram_message(message, chat_id)
        elif "off" in command:
            message = "Turned off LED"
            digitalWrite(led, 0)
            send_telegram_message(message, chat_id)
    elif content_type == 'voice':
        file_id = msg['voice']['file_id']
        try:
            audio_file = download_voice_message(file_id)
            command = process_voice_message(audio_file)

            print('Received voice command: %s' % command)

            if "on" in command:
                message = "Turned on LED"
                digitalWrite(led, 1)
                send_telegram_message(message, chat_id)
            elif "off" in command:
                message = "Turned off LED"
                digitalWrite(led, 0)
                send_telegram_message(message, chat_id)
            os.remove(audio_file)
        except Exception as e:
            print("Your Internet Connection is Unstable!!!", e)

telegram_bot =
telepot.Bot('6614083923:AAEpJXrFT3RHG7gX7NiSHEEfzYSOxZnO05s') #giving
bot token id to connect with the telegram bot
MessageLoop(telegram_bot, action).run_as_thread()

print('Up and Running....')

try: #trigerring the function to ultrasonic sensor
    while True:
        distance=detect_intruder()
        if distance<20:
```

```
            print("Intruder detected at Distance: %dcm" %distance)
            message = "Intruder detected! at %dcm" %distance
            send_telegram_message(message, '5244481830')
        time.sleep(2)

except KeyboardInterrupt: #key_interrupt to exit the program
    print("Program Stopped")
finally:
    print("Releasing resources")
    digitalWrite(led, 0)
    sys.exit(0)
```

VIII. Results/Output:

- Users found the system intuitive and convenient, seamlessly controlling the LED through both text and voice commands via the Telegram bot, thereby enhancing their overall experience.
- The ultrasonic sensor consistently detected intruders within its range, affirming the system's reliability in providing real-time security monitoring and ensuring users felt secure in their environment.
- Upon intruder detection, the system promptly dispatched alert messages to the designated Telegram chat, allowing users to quickly respond to potential security breaches and take necessary actions, thereby underscoring its responsiveness and effectiveness.
- These outcomes underscore the practicality and efficacy of the IoT system in improving both user convenience and security, highlighting its potential impact in enhancing everyday living experiences.

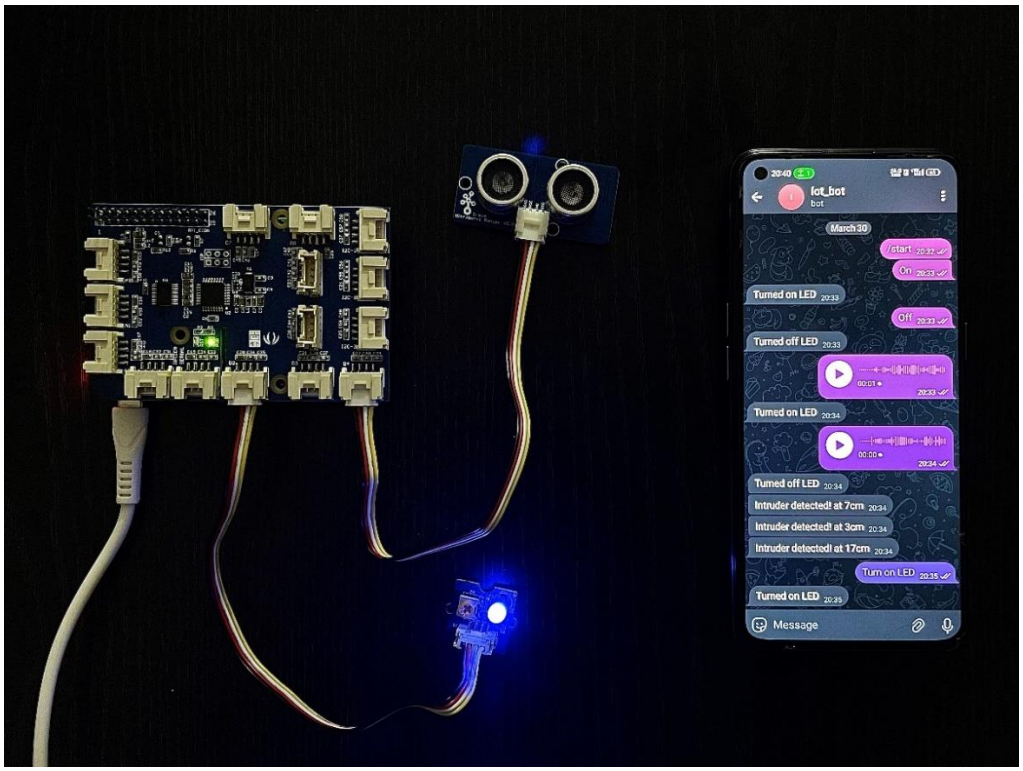

Fig: Demonstration of Home Automation

IX. Conclusion:

This project successfully demonstrates the creation of a basic IoT system for remote control and intruder detection using readily available tools and platforms. The system utilizes a Raspberry Pi as the central hub, Telegram as a user-friendly interface, and GrovePi modules for simplified connections. Users can control an LED remotely by sending text or voice commands through a Telegram bot. The project incorporates speech recognition capabilities, enhancing user experience by enabling voice-based control. Additionally, an ultrasonic sensor is integrated to detect potential intruders within a designated range. Upon detection, the system triggers an alert notification sent to a specific Telegram chat, informing the user of a potential security breach.

This project serves as a stepping stone for understanding how fundamental IoT principles can be applied to create practical applications. The core functionalities can be extended to incorporate additional devices and sensors, transforming the system into a more comprehensive solution. Imagine controlling multiple lights or appliances in different rooms or expanding the security features by adding motion sensors or door/window sensors. The possibilities for customization and expansion are vast.

Furthermore, this project underscores the accessibility and affordability of IoT development, particularly for hobbyists, students, and DIY enthusiasts. By leveraging off-the-shelf components like the Raspberry Pi and GrovePi modules, individuals with varying levels of technical expertise can embark on IoT projects without significant upfront costs or specialized hardware knowledge. This democratization of IoT technology empowers a broader range of individuals to engage in innovation and problem-solving within their communities, fostering a culture of experimentation and collaboration.

In addition to its practical applications, this project also underscores the importance of considering ethical and security implications in IoT design and deployment. As IoT devices become increasingly integrated into our daily lives, ensuring data privacy, device security, and responsible usage practices becomes paramount. By incorporating encryption protocols, user authentication mechanisms, and transparent communication practices, developers can mitigate potential risks associated with unauthorized access or data breaches. Moreover, promoting education and awareness about cybersecurity and ethical considerations in IoT development can empower users to make informed decisions and advocate for responsible IoT deployment practices in their communities.

Overall, this project highlights the potential of readily available tools and platforms in creating functional IoT systems. By combining a Raspberry Pi, Telegram, and basic electronic components, users can gain valuable experience in remote control, sensor data monitoring, and security applications, paving the way for further exploration and innovation in the exciting world of IoT.