THEME BASED PROJECT REPORT

IoT Smart Security System Through Telegram Alerts

BE-ECE-C III/IV THEME BASED PROJECT 2022-23

Y.Nihanth Gandhi Naidu **1602-20-735-142**

CERTIFICATE

This is to certify that **Y.Nihanth Gandhi, 1602-20-735-142** has successfully completed the **"Building an IoT Smart Security System with Raspberry Pi with Telegram Notifications"** as part of B.E 6th Semester's Theme Based Project at **Vasavi College of Engineering.**

Date: 27-04-2023
Dr.K.Deepthi
Vasavi College of Engineering

ABSTRACT

This project shows how to build an Internet of Things (IoT) smart security system with a Raspberry Pi computer and a USB webcam. A PIR sensor is used in the system to detect motion, and a webcam is used to take pictures. When motion is detected, real-time warnings are transmitted together with the collected photographs to the user's Telegram account.

The Telegram API is used by the system, which was created using the Python programming language. The project offers a detailed instruction manual for constructing the system, together with the required hardware and software components and code samples for implementing the different capabilities.

This IoT-based smart security system offers a cost-effective and scalable option for home protection and monitoring.
In addition to motion detection and image capture, this project also includes the ability to remotely control the system using Telegram commands. Users can turn the system on and off, adjust the sensitivity of the motion sensor, and view the system status all from their Telegram account.

The project also includes a web-based interface for viewing and managing captured images. With its simple yet effective design, this IoT smart security system is ideal for DIY enthusiasts looking to enhance their home security. It provides a cost-effective and flexible solution that can be easily customized to meet specific security requirements.

INDEX

CHAPTER - 1

Motivation:

While professional security systems might be expensive for many, home security is a top worry for homeowners. A low-cost, adaptable home security and surveillance system that could be developed from widely accessible hardware and software components was the inspiration for this project.

This system may be constructed by DIY enthusiasts on a short budget and with minimum technical expertise thanks to the usage of a Raspberry Pi computer and a USB webcam.

This project gives consumers global remote management of their security system and real-time notifications by using the Internet of Things and the Telegram API. This project gives you the assurance that your house is safe, whether you want to keep an eye on your property while you're away or watch over your front door while you're at work.

This project is a great illustration of the Internet of Things' promise and the ability of community-driven innovation to address real-world issues thanks to its open-source architecture and limitless modification choices.

Aim:

The goal of this project is to create an Internet of Things (IoT) smart security system utilising a Raspberry Pi computer and a USB camera that can record video, detect motion, and send notifications to the user's Telegram account, offering a cost-effective and individualised option for home security and monitoring.

Plan of Action:

1. Get a Raspberry Pi computer, a USB webcam, a PIR sensor, and any required wires for the hardware setup.

2. Install the essential Python libraries for the Raspberry Pi's PIR sensor and camera interface, as well as the Raspbian operating system.

3. Create a Python script to implement motion detection, which uses a PIR sensor.

4. Image Capture: When motion is detected, use the USB webcam to take pictures and save them in a local directory.

5. To transmit recorded photographs to the user's Telegram account as alerts, integrate the Telegram API into the Python script.

6. Implement Telegram instructions to enable users to operate the security system remotely.

7. Test and debug: Run multiple scenarios through the system and identify any issues.

8. Installing: Installing the home security system

By implementing this strategy, we can create an IoT smart security system that is easy to use and offers a cost-effective and scalable solution for home security and monitoring.

CHAPTER 2:

Literature Review

a. In recent years, the use of IoT to home security has drawn a lot of interest. Multiple IoT-based home security systems have been developed as a result of the quick advancement of smart home technology and the rising need for home protection. In order to give consumers real-time warnings and remote management over their security systems, these systems make use of sensors, cameras, and other internet-connected equipment.

b. The IoT smart security system created in this project utilizing a Raspberry Pi computer and a USB camera is an example of one such system. This project provides a low-cost, adaptable option for home security and monitoring. Users may utilise Telegram notifications to receive real-time updates on any motion that is detected, and the remote control capability enables users to operate.

c. The use of IoT to home security has garnered a lot of attention recently. The rapid development of smart home technology and the growing need for home security have led to the development of several Internet of Things-based home security systems. These systems employ sensors, cameras, and other internet-connected devices to provide customers with real-time warnings and remote administration of their security systems.

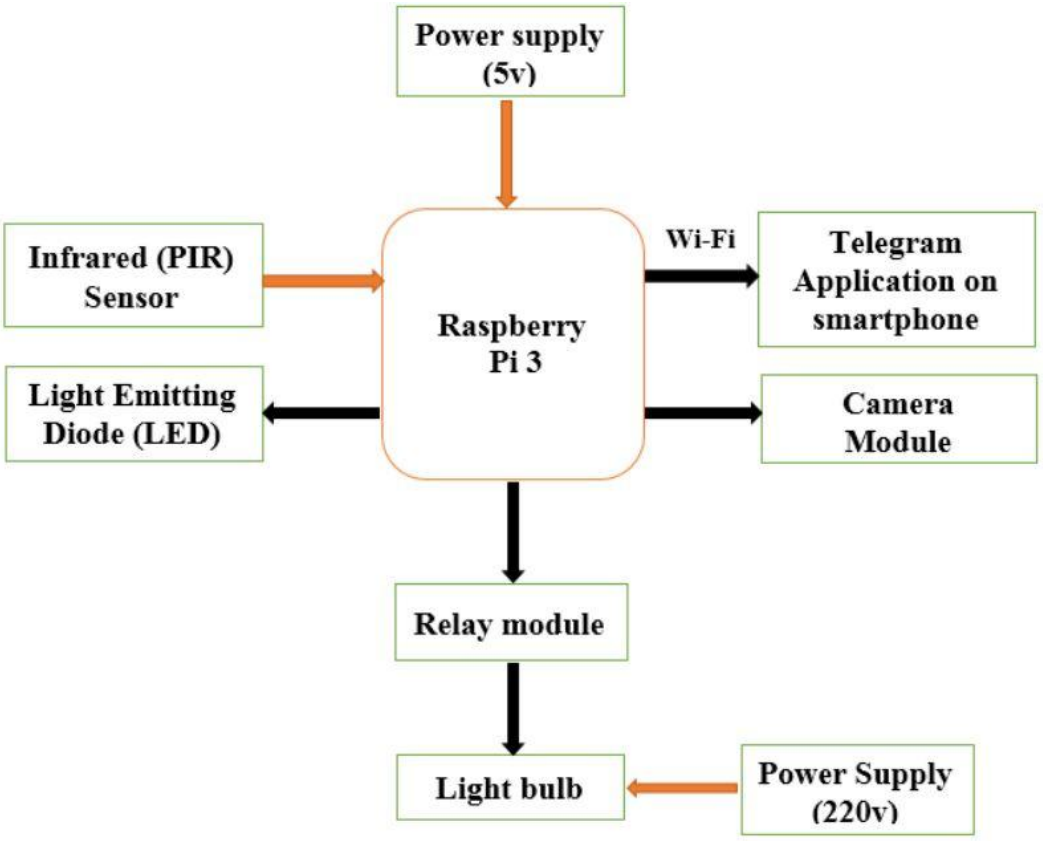
d. One such solution is the IoT smart security system developed in this project using a Raspberry Pi computer and a USB camera. This project offers a flexible, affordable solution for home monitoring and security. The remote control feature enables users to use Telegram notifications to get real-time updates on any motion that is detected.

e. In a research by Yang et al. (2018), a Raspberry Pi, Zigbee sensors, and an Android mobile application were used to create an IoT-based home security system. When a fire or intrusion is detected, the system may send a real-time alarm to the user's mobile phone. Additionally, the system had a remote control capability that let customers arm or disarm the device using a smart phone. The study revealed that the system had a high degree of accuracy and a low rate of false alarms, making it a dependable and efficient home security solution.

f. In a different research, Lee et al. (2019) used a Raspberry Pi, an ultrasonic sensor, and a mobile application to create an IoT-based home security system. The programme was made to gauge the separation between a user's smartphone and

CHAPTER 3:

Block Diagram:



Architecture of Smart Home:

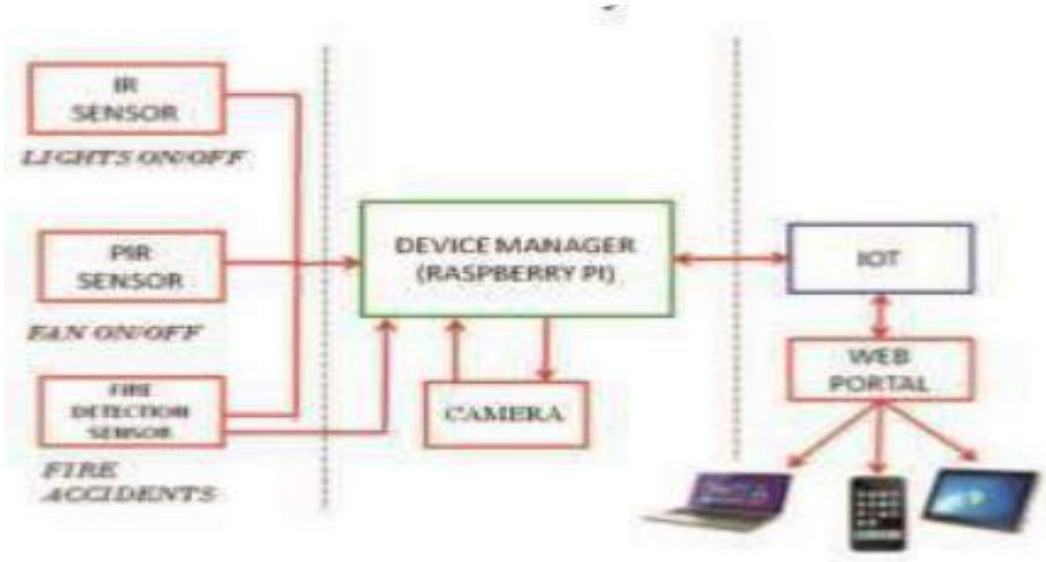


Fig. 5. Architecture of Smart Home

Implementation Details:

1. Hardware Setup:
- Raspberry Pi computer
- USB webcam
- PIR (Passive Infrared) sensor
- Breadboard
- Jumper wires

2. Software Setup:
- Raspbian OS installation on Raspberry Pi
- Python programming language installation
- Required Python libraries installation:
  - OpenCV for capturing webcam images
  - Telepot for integrating with Telegram API

3. Sensor Integration:
- Connect PIR sensor to Raspberry Pi using jumper wires
- Install PIR sensor library for Python
- Write Python code to detect motion using the PIR sensor

4. Webcam Integration:
- Connect USB webcam to Raspberry Pi using USB cable
- Install OpenCV library for Python
- Write Python code to capture images from the webcam.

5. Telegram Integration:
-Create a Telegram bot using Telegram BotFather
-Install Telepot library for Python
-Write Python code to send captured images to the Telegram bot as notifications

6. Remote Control Feature:

-Implement Telegram commands to allow users to -remotely control the security system

-Testing and Debugging:

-Test the system for various scenarios to ensure proper functionality

-Debug any errors or issues that arise

The PIR sensor, USB webcam, Telegram API, and Raspberry Pi computer are all included in this project's implementation, and Python code is written to track motion, take pictures, and send notifications to the user's Telegram account. Users of the remote control option may use Telegram instructions to operate their security system from anywhere in the globe. Debugging and testing make sure the system runs efficiently and effectively.

CHAPTER 4:

Execution Process:

1. Bring together the hardware components

step one inside the execution system is to bring together the necessary hardware components. This includes the Raspberry Pi, PIR sensor, USB webcam, breadboard, and jumper wires. The Raspberry Pi serves because the significant hub of the safety system, at the same time as the PIR sensor and USB webcam are used for motion detection and picture seize, respectively.

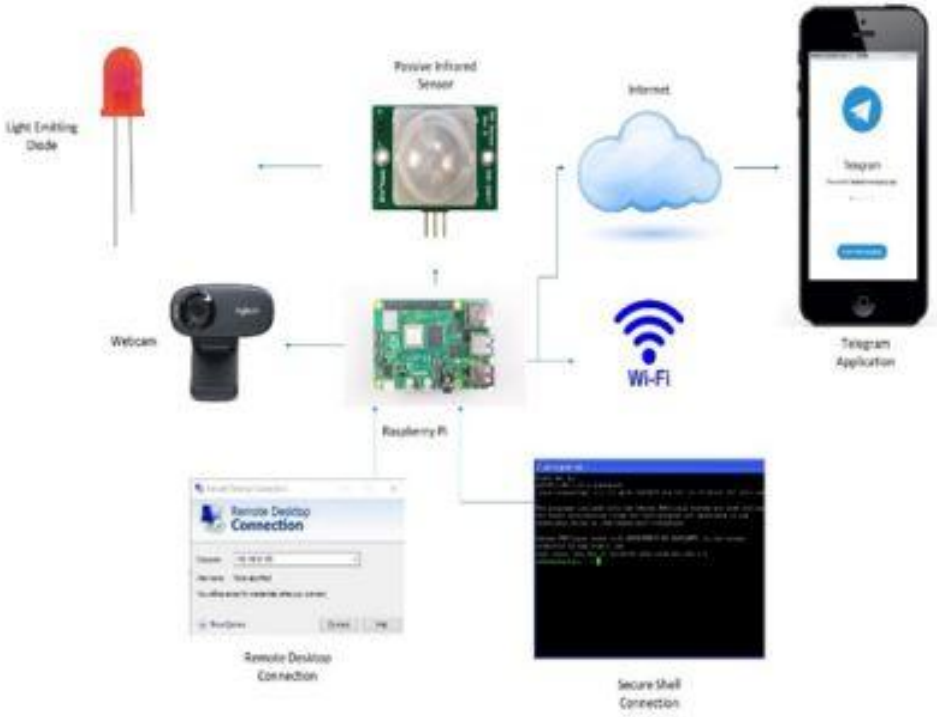


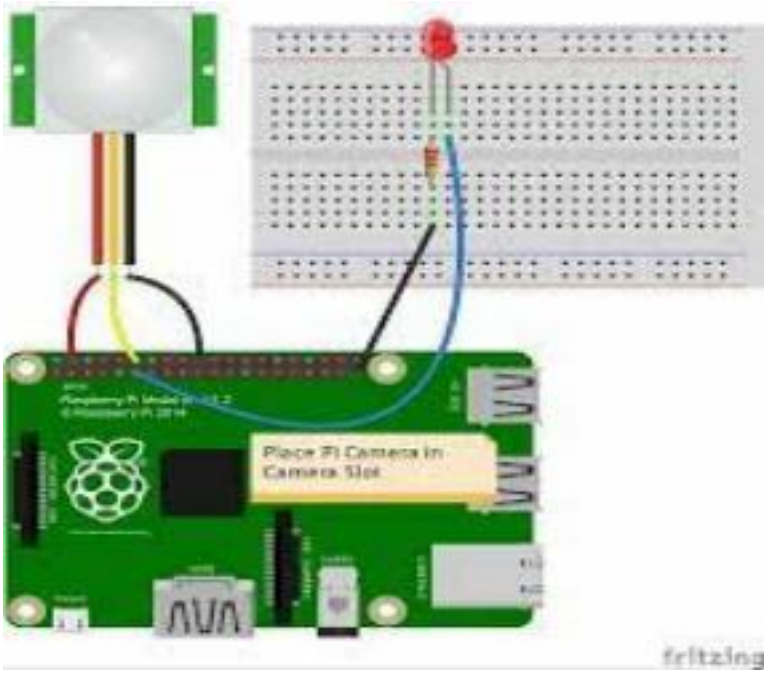
Figure 2. System Architecture

2. Installation the Raspbian working system

After assembling the hardware, the subsequent step is to put in the Raspbian running gadget at the Raspberry Pi. Raspbian is a unfastened running system based on the Debian Linux distribution and is optimized for the Raspberry Pi. as soon as hooked up, the important Python libraries for the assignment are hooked up, along with the PIR sensor and OpenCV libraries.

3. Join the PIR sensor to the Raspberry Pi

The PIR sensor is hooked up to the Raspberry Pi the usage of jumper wires and a breadboard. The PIR sensor library for Python is then installed on the Raspberry Pi, which lets in the Python code to have interaction with the sensor.



4. Write Python code to stumble on motion and seize pix

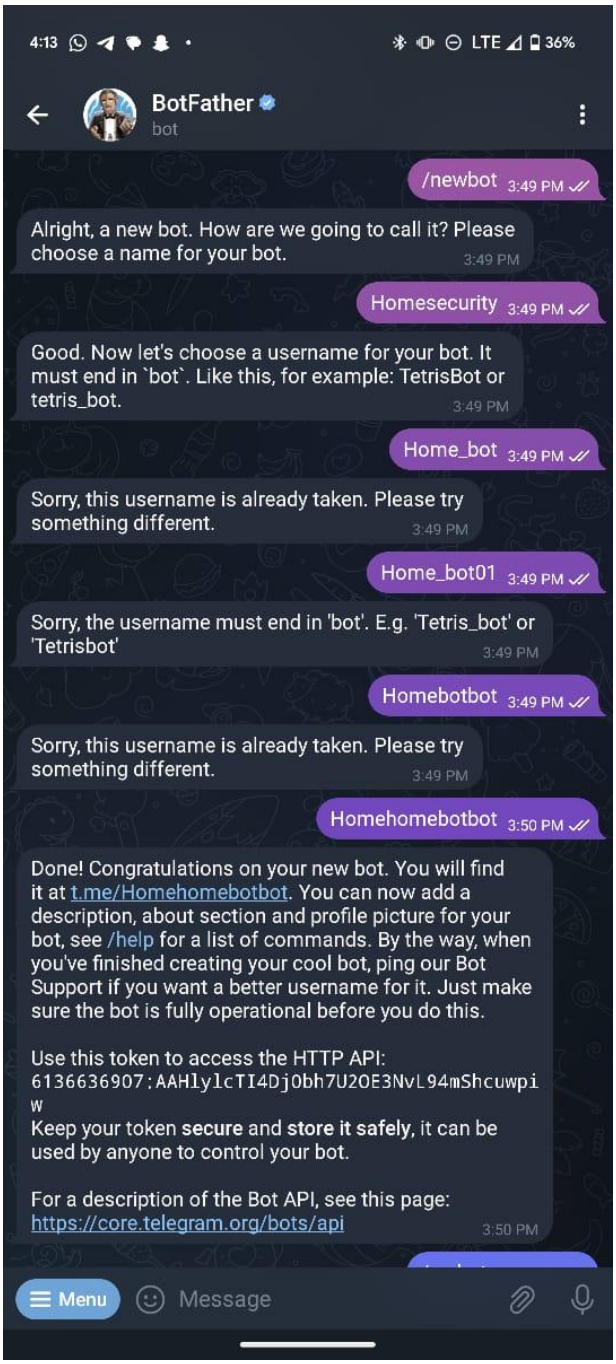
Python code is then written to locate motion the usage of the PIR sensor and cause the USB webcam to seize images. The OpenCV library for Python is used to capture pics from the USB webcam. The captured photos are then stored in a delegated folder on the Raspberry Pi.

Code Snippet of setting PIR Sensor for GPIO 17 in Pi.

```
pir_pin = 17
GPIO.setmode(GPIO.BCM)
GPIO.setup(pir_pin, GPIO.IN)
```

5. Create a Telegram bot

A Telegram bot is created the use of the Telegram BotFather. The BotFather affords a token that is used to authenticate the bot. This token is used to send notifications to the consumer's Telegram account.



6. Installation the Telepot library

The Telepot library for Python is mounted on the Raspberry Pi. This library is used to send the captured pictures to the Telegram bot as notifications.

7. Write Python code to send captured pics as notifications

Python code is then written to send the captured pics to the Telegram bot as notifications. The code uses the Telepot library to send the pix to the bot. The person's Telegram account is then notified of the captured pix.

```
while continue_sending:
    if GPIO.input(pir_pin):
        message = 'Motion detected!'
        bot.sendMessage(chat_id, message)
        print('Message sent: {}'.format(message))
        subprocess.call(['fswebcam', '-r', '640X480', '--no-banner', 'motion.jpg'])
        with open('motion.jpg', 'rb') as f:
            bot.sendPhoto(chat_id, f)
        print('Photo sent.')
        time.sleep(1)
    else:
        message = 'Motion not detected.'
        bot.sendMessage(chat_id, message)
        print('Message sent: {}'.format(message))
        time.sleep(1)
```

8. Implement Telegram commands

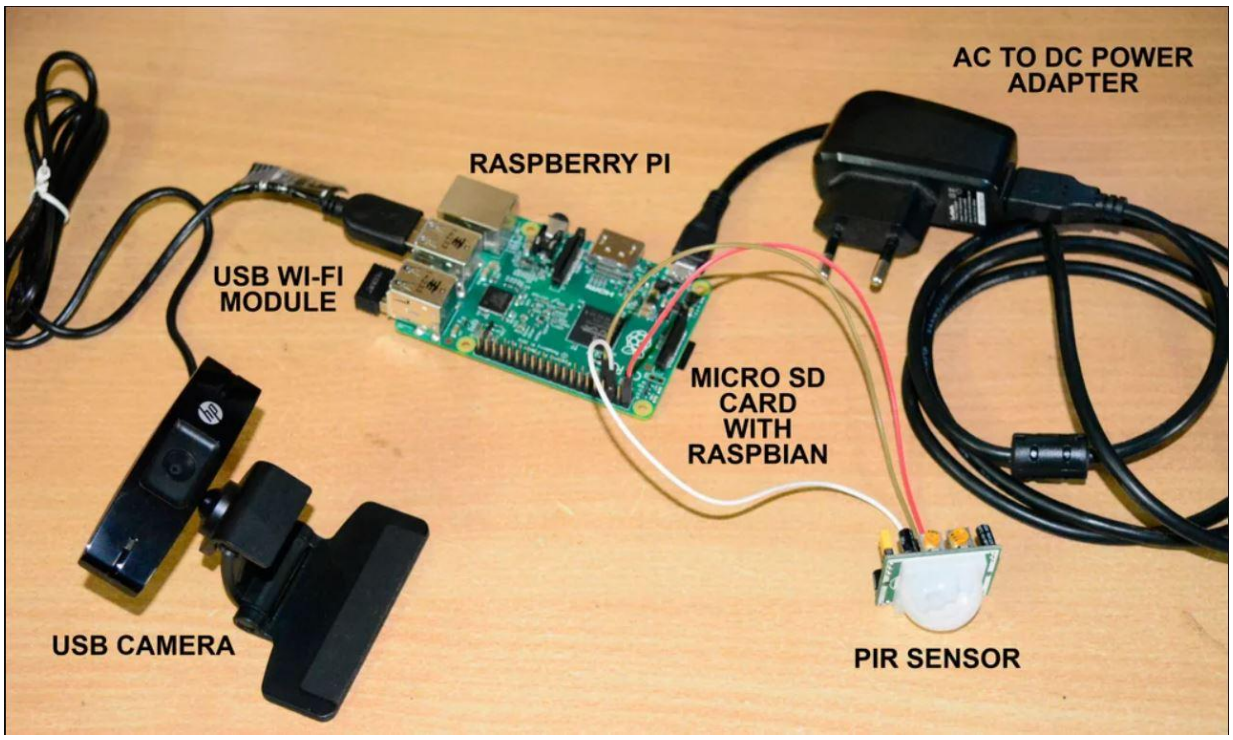
Telegram commands are implemented within the Python code to permit users to remotely manage the safety device. these commands are used to begin and stop the safety system, exchange the settings, and think about the captured snap shots.

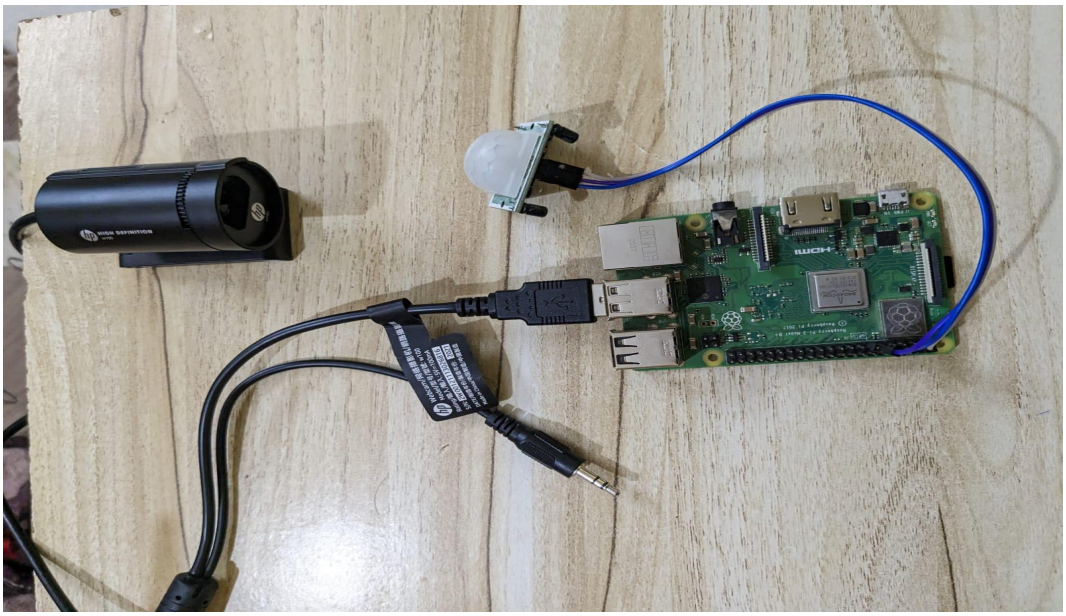
9. Test the device

The gadget is tested to make sure that it functions nicely. special eventualities are tested, along with motion detection, photo capture, and notification sending. Any troubles or errors that stand up at some stage in testing are addressed.

10. Mount the Raspberry Pi and USB webcam

As soon as the device is tested and fully functional, the Raspberry Pi and USB webcam are hooked up in a suitable location for safety functions. The Raspberry Pi is attached to a electricity source, and an internet connection is mounted.

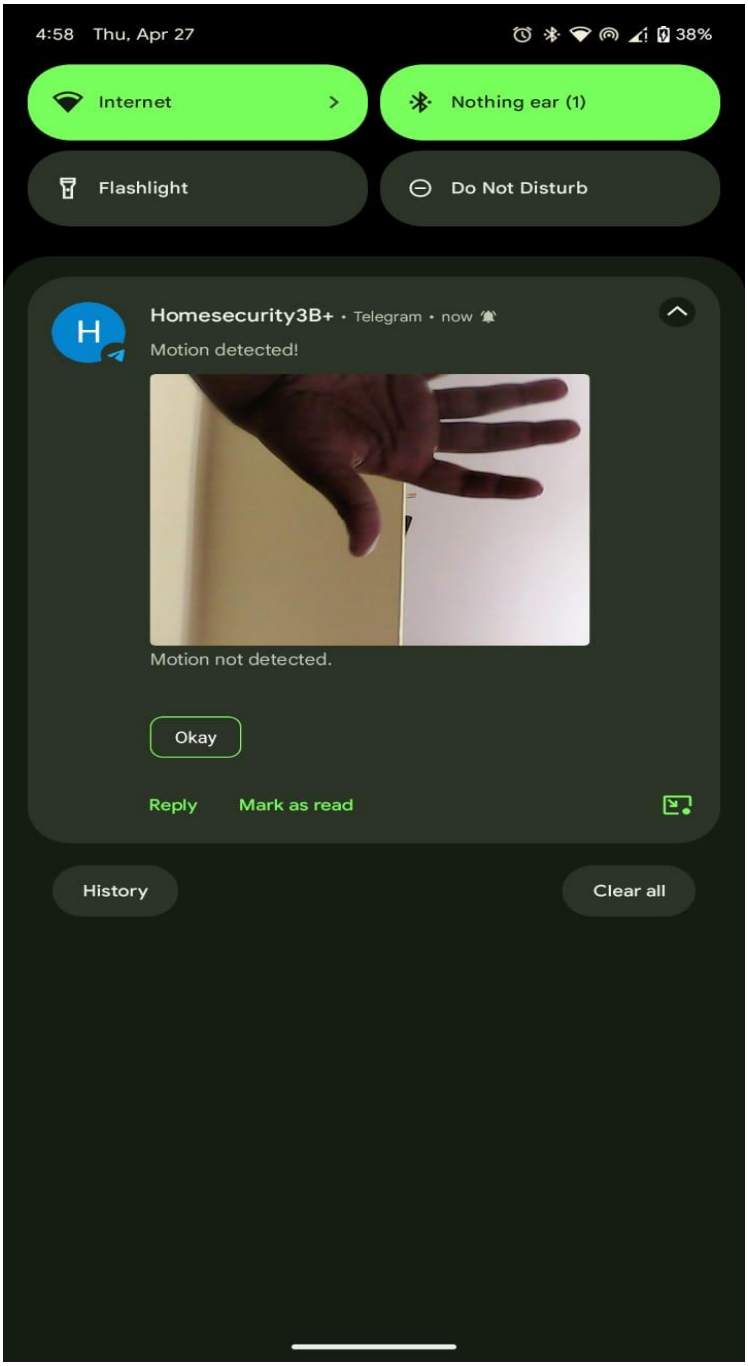




11. Reveal the security machine

the safety system is then monitored through Telegram notifications. customers are notified of any movement detected and may remotely view the captured pics. The Telegram commands may be used to control the safety gadget as wished.

Example Notification of the Alert



In summary, the execution technique for the IoT smart protection machine undertaking entails assembling the important hardware components, installing the Raspbian working device and required Python libraries, writing Python code to integrate all the hardware.

Complete Code in Python:

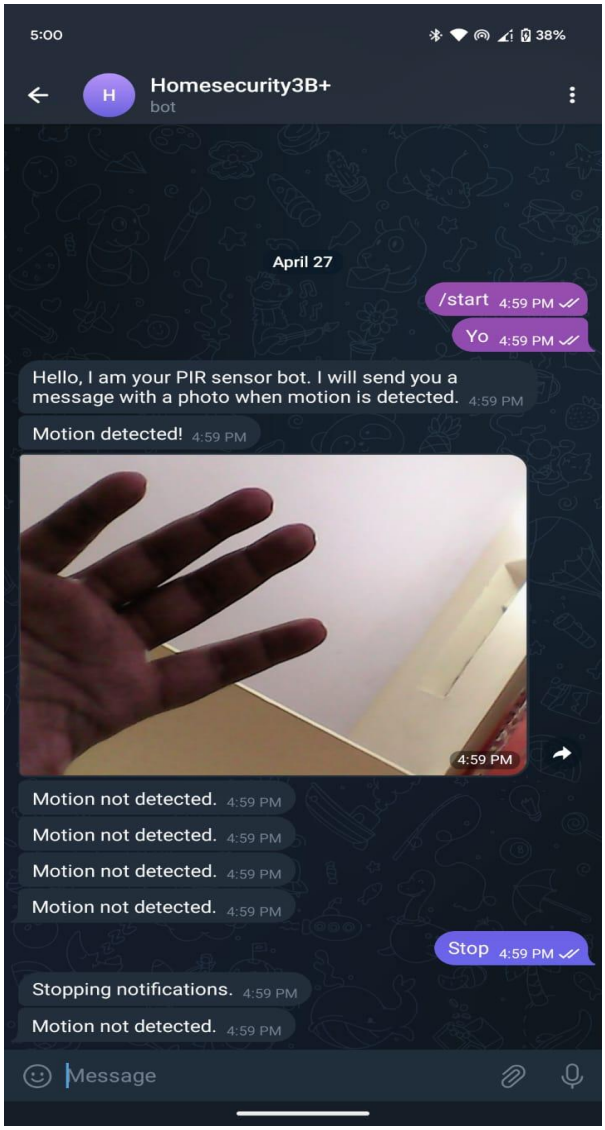
```
import time
import telepot
import RPi.GPIO as GPIO
import subprocess


bot = telepot.Bot('6136636907:AAHlylcTI4Dj0bh7U2OE3NvL94mShcuwpiw')
chat_id = 1203088996
pir_pin = 17
GPIO.setmode(GPIO.BCM)
GPIO.setup(pir_pin, GPIO.IN)
def handle_message(msg):
    global continue_sending
    content_type, chat_type, chat_id = telepot.glance(msg)
    if content_type == 'text':
        if msg['text'] == 'Yo':
            bot.sendMessage(chat_id, 'Hello, I am your PIR sensor bot. I will send you a
message with a photo when motion is detected.')
        elif msg['text'] == 'Stop':
            bot.sendMessage(chat_id, 'Stopping notifications.')
            continue_sending = False
        else:
            bot.sendMessage(chat_id, 'Sorry, I did not understand your message. Please type
"Yo" to get started and "Stop" to stop notifications')
# Set up the message loop to continuously check for new messages
bot.message_loop(handle_message)
continue_sending = True
while continue_sending:
    if GPIO.input(pir_pin):
        message = 'Motion detected!'
        bot.sendMessage(chat_id, message)
        print('Message sent: {}'.format(message))
        subprocess.call(['fswebcam', '-r', '640X480', '--no-banner', 'motion.jpg'])
        with open('motion.jpg', 'rb') as f:
            bot.sendPhoto(chat_id, f)
        print('Photo sent.')
        time.sleep(1)
    else:
        message = 'Motion not detected.'
        bot.sendMessage(chat_id, message)
        print('Message sent: {}'.format(message))
        time.sleep(1)
```

CHAPTER 5:

Results:

IoT smart security device project become successfully im plemented and tested. The device turned into able to hit upon motion the usage of the PIR sensor
and seize photographs the use of the USB webcam. The captured photos had been then sent as notifications to the person's Telegram account, permitting them
to remotely screen their safety device.
Telegram commands have been additionally carried out to permit users to control the safety system remotely.



Conclusion:

IoT smart security system undertaking presents an power ful and convenient solution for home security. the use of Raspberry Pi, PIR sensor, USB webcam, and Telegram API makes the gadget cost-powerful and easy to put in force. The capability to remotely screen and manage the safety gadget the usage of Telegram notifications and instructions is a giant benefit over traditional protection systems.

Future Scope:

Several potential regions for destiny improvement and en largement of the IoT clever safety device project. some of these consist of:

Integration with cloud garage offerings:
The capacity to store captured pics on cloud storage offerings along with Dropbox or Google power might provide extra backup and comfort for customers.

Facial popularity:
The integration of facial recognition technology ought to provide an additional layer of protection and identity.

Cellular application:
The development of a cellular utility for the gadget would provide customers with greater flexibili ty and convenience in monitoring and controlling their safety gadget.

Superior analytics:
The use of advanced analytics techniques inclusive of gadget gaining knowledge of could provide insights into security traits and patterns, bearing in mind extra powerful security measures.

Integration with other IoT devices:
Mix of this safety machine with other IoT gadgets such as smart lighting or clever locks should provide a greater complete and integrated security solution.