VISVESVARAYA TECHNOLOGICAL UNIVERSITY

Jnana Sangama, Belagavi – 590014.

Internship Report
On

"Home Automation and Security System"

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

BACHELOR OF ENGINEERING
In
COMPUTER SCIENCE & ENGINEERING
&
INFORMATION SCIENCE & ENGINEERING

By

Kusuma S	**1AP21IS015**
Gnanesh V S	**1AP21CS019**
Syed Awaiz Ahmed	**1AP21CS048**

Under the guidance of:

AKHIL SAI

2024 - 2025

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
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: Home Securiy and Automation System

Name of the Students: Kusuma S,

Gnanesh V S

Syed Awaiz Ahemd

External Supervisor:

____

Internal Supervisor:

____

____

Date:

Place:

Project Completion Certificate

I, **Kusuma S** (USN:1AP21IS015), hereby declare that the material presented in the Project Report titled **"Home Securiy and Automation System"** represents original work carried out by me in the **Department of Information Science & Engineering** at the **APS college of Engineering, Bengaluru** 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.E. Internship Work.

Advisor's Name: **Dr.Shivamurthaiah** Guide Name: **Akhil Sai**

Advisor's Signature **Guide Signature**

Project Completion Certificate

I, **Gnanesh V S** (USN:1AP21CS019), hereby declare that the material presented in the Project Report titled **"Home Securiy and Automation System"** represents original work carried out by me in the **Department of Computer Science & Engineering** at the **APS college of Engineering, Bengaluru** 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.E. Internship Work.

Advisor's Name: **Dr.Shivamurthaiah** Guide Name: **Akhil Sai**

Advisor's Signature **Guide Signature**

Project Completion Certificate

I, **Syed Awaiz Ahmed** (USN:1AP21CS048), hereby declare that the material presented in the Project Report titled **"Home Securiy and Automation System"** represents original work carried out by me in the **Department of Computer Science & Engineering** at the **APS college of Engineering, Bengaluru** 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.E. Internship Work.

Advisor's Name: **Dr.Shivamurthaiah** Guide Name: **Akhil Sai**

Advisor's Signature **Guide Signature**

TABLE OF CONTENTS

CHAPTER 8

LIST OF FIGURE

Chapter 1

ABSTRACT

This project introduces a smart home monitoring system designed using a Raspberry Pi and built with the Flask web framework. The system integrates multiple sensors and hardware components to monitor environmental conditions and provide security features. A DHT11 sensor measures temperature and humidity, displaying these readings on a web dashboard for user monitoring. An ultrasonic sensor is used to detect motion within a 10 cm range, and if motion is detected, the system activates a buzzer for an audible alert. Additionally, a request is sent to a laptop server to trigger its webcam, offering potential security surveillance.

The system allows users to interact with it via a web interface, providing control over the system's functionality. Users can switch the monitoring system on or off, and they can toggle specific LEDs to indicate the system’s operational status. The LEDs also provide visual feedback, while the buzzer acts as an alert during motion detection. The Flask app communicates seamlessly with the Raspberry Pi’s GPIO pins to manage these hardware elements.

Session management within the Flask app tracks the system's status and user actions, ensuring smooth operation. The user-friendly web interface makes it easy for users to monitor the environment and control the system remotely. This project merges environmental monitoring with security features, offering a simple and scalable solution for home automation. Future enhancements could include integrating cloud storage for data logging, developing mobile app interfaces, enhancing motion detection with machine learning, and extending compatibility with other IoT devices for broader smart home integration.

Chapter 2

INTRODUCTION

2.1 Objective:

The provided Python Flask code sets up a Raspberry Pi-based IoT monitoring and control system. It integrates an ultrasonic sensor for motion detection, a buzzer for audible alerts, LEDs for visual feedback, and a DHT11 sensor to monitor temperature and humidity. Upon detecting motion, the system triggers a buzzer and sends a request to a remote laptop server to activate a webcam for surveillance. Users can control the system through a web interface, enabling or disabling the monitoring system and controlling individual LEDs. The system ensures real-time feedback and efficient monitoring, making it ideal for home automation and security applications.

2.2 Problem Statement:

1. **Affordability**: Existing home security systems are expensive and inaccessible for many users.
2. **Integration Issues**: Lack of a unified system for motion detection, environmental monitoring, and alerts.
3. **Limited User Control**: Existing solutions often lack an intuitive, user-friendly interface.
4. **Real-Time Monitoring**: Difficulty in enabling remote surveillance features like triggering a webcam.
5. **Proposed Solution**:
    - Raspberry Pi-based system with motion detection, temperature, and humidity sensors.
    - Includes LEDs and a buzzer for alerts.
    - Offers a web interface for control and monitoring.
    - Enables remote webcam activation for surveillance.

Chapter 3

Applications

Residential Security:

- Detect motion near doors or windows to identify potential intrusions.
- Trigger an audible alert via the buzzer and send a request to a connected laptop for webcam activation, allowing real-time video surveillance.
- Monitor remotely through the web interface for added convenience.

Environmental Monitoring:

- Measure temperature and humidity using the DHT11 sensor to ensure comfortable indoor conditions.
- Helpful in areas prone to extreme weather changes to maintain safety and comfort.

Smart Automation:

- Enable or disable LED lights remotely for energy conservation and convenience.
- Integrate with additional IoT devices like smart plugs or HVAC systems for complete automation.

Pet Care:

- Detect unusual movement, ensuring pets remain in designated areas.
- Maintain optimal environmental conditions to provide a safe and comfortable environment for pets.

Educational Demonstration:

- Use as an IoT project to teach students and hobbyists about sensor integration, web interfacing, and automation.
- Demonstrates real-world applications of Python programming and Raspberry Pi in IoT projects.

Greenhouse Management:

- Monitor temperature and humidity in greenhouses to maintain ideal growing conditions for plants.
- Automate additional controls, like irrigation systems or ventilation, based on sensor data.

Remote Property Monitoring:

- Monitor unoccupied vacation homes or rental properties for unauthorized access or environmental changes.
- Notify users of any unusual activity, allowing them to take immediate action remotely.

Chapter 4

Components

1.AIOT kit:

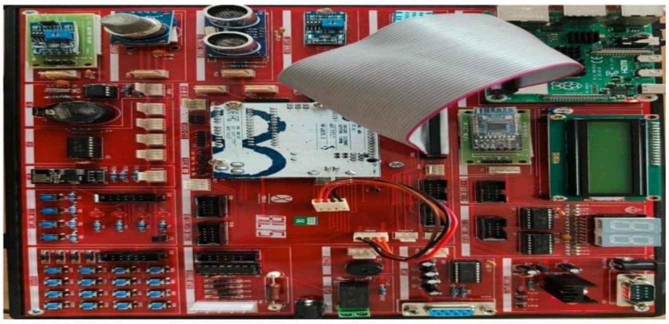


Fig 4.1

i.Raspberry Pi(Model 4):

1. The Raspberry Pi 4 is a powerful, low-cost single-board computer with a 1.5 GHz quad-core processor, up to 8 GB RAM, and multiple connectivity options like USB, HDMI, and Ethernet.
2. It features General Purpose Input/Output (GPIO) pins, making it ideal for DIY projects, IoT applications, and prototyping, offering flexibility for users to interface with sensors, actuators, and other electronic components.

Fig 4.2

ii.DTH11 (Temperature and Humidity) Sensor

1. The DHT11 is a low-cost digital temperature and humidity sensor commonly used in DIY electronics projects, providing reliable readings for temperature (range: 0-50°C) and humidity (range: 20-80% RH).

2. It uses a single-wire digital interface, making it easy to integrate with microcontrollers like the Raspberry Pi and Arduino. Though less accurate than other sensors, it's suitable for basic environmental monitoring in applications such as home automation and weather stations.

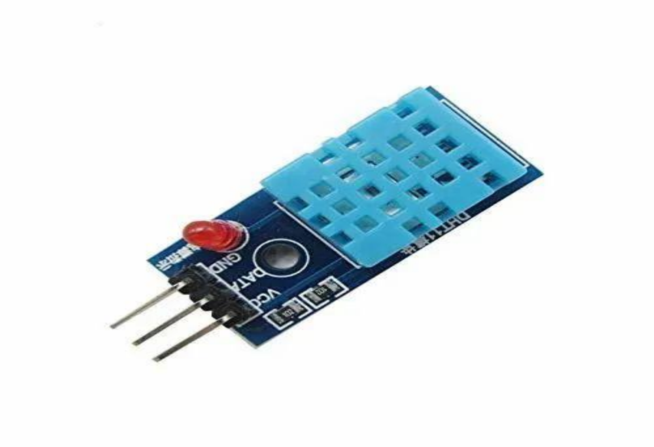


Fig 4.3

iii.Ultrasonic Sensor

1. An ultrasonic sensor measures distance by emitting sound waves and calculating the time it takes for the waves to bounce back after hitting an object, offering high accuracy in distance measurement.
2. Commonly used in robotics, automation, and object detection applications, ultrasonic sensors are typically integrated with microcontrollers like Raspberry Pi and Arduino for projects like motion detection, obstacle avoidance, and range finding due to their simple interface and reliable performance.

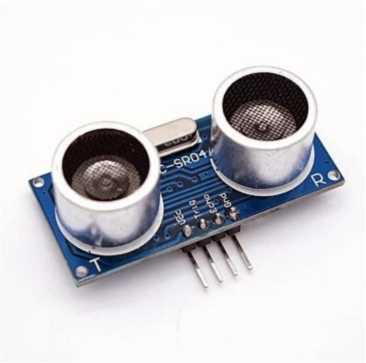


Fig 4.5

iv.LED

1. An LED (Light Emitting Diode) is a semiconductor device that emits light when an electrical current passes through it, known for its energy efficiency, longevity, and fast response time.

Fig 4.6

v.Buzzer

1. A buzzer is an electronic component that produces sound when powered, commonly used for alerts, alarms, or signaling purposes in various applications.
2. In DIY electronics, buzzers are often integrated with microcontrollers like Raspberry Pi or Arduino to provide audible feedback or warnings, such as in security systems, motion detectors, and interactive projects, offering a simple and effective way to alert users.

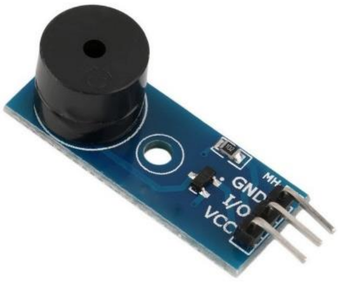


Fig 4.7

Chapter 5

Flow Diagram

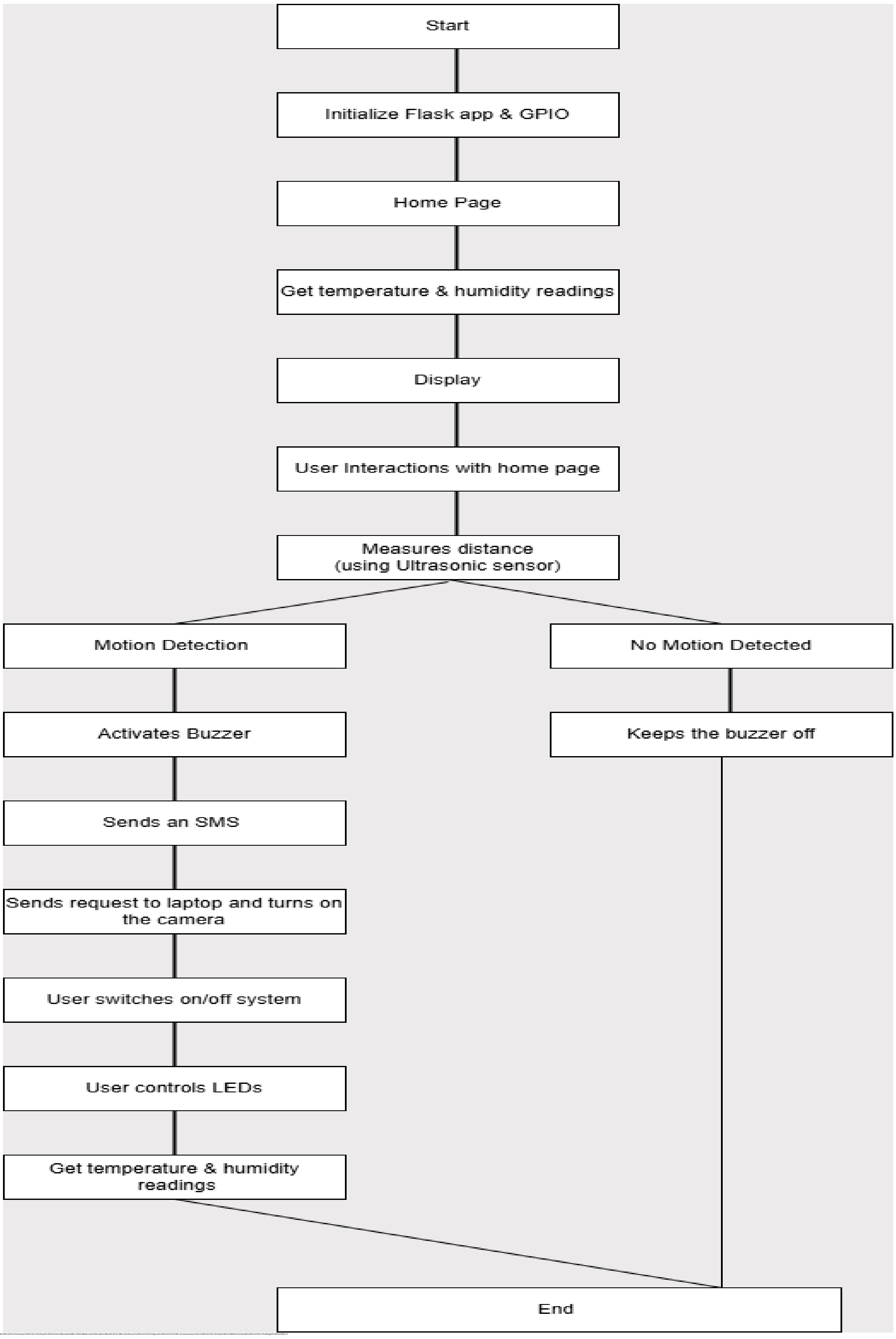
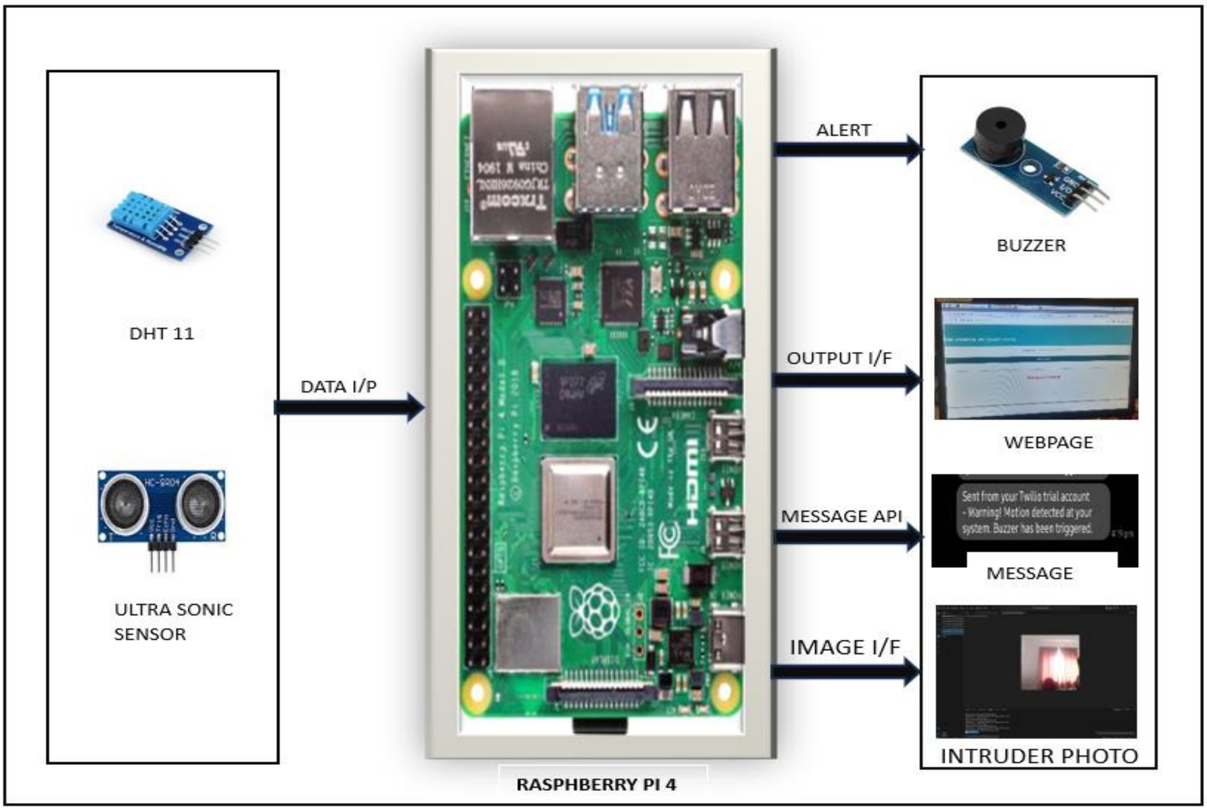

Start
Initialize Flask app & GPIO
Home Page
Get temperature & humidity readings
Display
User Interactions with home page
Measures distance
(using Ultrasonic sensor)
Motion Detection
No Motion Detected
Activates Buzzer
Keeps the buzzer off
Sends an SMS
Sends request to laptop and turns on the camera
User switches on/off system
User controls LEDs
Get temperature & humidity readings
End

Fig5.1



Sent from your Twilio trial account - Warning! Motion detected at your system. Buzzer has been triggered.

Fig5.2

Chapter 6

Conclusion

This Flask-based system is a smart environmental and security monitoring solution, combining multiple sensors and hardware components with a web-based interface. It utilizes a DHT11 sensor to monitor temperature and humidity, which is displayed to the user on the homepage. The system also features an ultrasonic sensor for motion detection; if an object or person is detected within 10 cm, the system triggers a buzzer and sends a request to a laptop server to activate its webcam. Users can control the system remotely, turning the monitoring system on or off, and toggling LEDs that indicate the system’s status.

The Flask web app offers a user-friendly interface, where users can check environmental readings, view motion detection status, and control system features. LEDs are used to indicate the operational status of the system and the sensors, while a buzzer provides an alert when motion is detected. Additionally, the system can be customized to control specific lights, providing further flexibility.

The combination of real-time environmental data and security features such as motion detection makes this system suitable for a variety of applications, from simple home automation to security monitoring. The modular nature allows users to add or modify features based on their requirements, offering a scalable solution for different use cases.

Chapter 7

Future Work

Here are potential future works for enhancing and expanding the current system:

1. **Integration with Cloud Services**:
   - Implement cloud storage or databases (e.g., AWS, Firebase) to log temperature, humidity, and motion detection data. This would allow users to monitor trends over time and remotely access historical data from anywhere.
2. **Mobile App Interface**:
   - Develop a mobile app to interact with the system, offering more convenient control and notifications. The app could also provide push alerts for motion detection, system status, or environmental changes.
3. **Enhanced Motion Detection**:
   - Integrate additional sensors like PIR (Passive Infrared) sensors for more accurate motion detection, reducing false positives from static objects. Additionally, adding AI-based motion analysis could improve the system's ability to distinguish between human and non-human motion.
4. **Voice Control Integration**:
   - Integrate voice-controlled assistants like Amazon Alexa, Google Assistant, or custom voice commands to control the system’s features, such as turning on/off lights, arming/disarming security, or fetching environmental data.
5. **Energy Efficiency Features**:
   - Introduce features to optimize energy consumption, such as automatic power-saving modes or integrating solar panels for powering sensors and LEDs. This would make the system more eco-friendly and cost-efficient in the long term.
6. **Integration with Smart Home Devices**:
   - Expand the system’s compatibility with other smart home devices, like smart thermostats, cameras, or alarms, creating a more comprehensive and automated home security and environmental control solution.

Chapter 8

Appendix

1. **Libraries Used**: Flask, gpiozero, os, time, requests, Adafruit_DHT, RPi.GPIO.
2. **GPIO Configuration**: LEDs (GPIO 5, 3, 12), Ultrasonic sensor (GPIO 19, 26), Buzzer (GPIO 20).
3. **Functions**: distance() for ultrasonic distance measurement, get_hum_temp_reading() for temperature and humidity.
4. **Routes**: / displays temperature and humidity, /home monitors motion, /switch_on_system//switch_off_system toggle the system.
5. **Web Interface**: HTML templates options.html and home.html for system control and data display.
6. **System Behavior**: Motion detection triggers buzzer and laptop webcam activation.
7. **Error Handling**: Catches failed requests or GPIO errors, prints error messages.
8. **Security**: Session management and secret key (os.urandom(12)) for encryption.
9. **Hardware**: Raspberry Pi, DHT11 sensor, ultrasonic sensor, LEDs, and buzzer.
10. **Future Enhancements**: Cloud integration, mobile app, advanced motion detection, smart home IoT support.

8.1Pseudo Code

```
START
1. Import required libraries (Flask, gpiozero, os, time, requests, Adafruit_DHT, RPi.GPIO)
2. Initialize Flask app and set a secret key for session management
3. Setup GPIO for system LEDs (LED1, LED2), ultrasonic sensor, and buzzer
   - Set GPIO pin modes (output for LED and trigger, input for echo)
   - Initialize buzzer pin to LOW
4. Define the Laptop server URL for capturing webcam (replace with actual IP)
5. Define a function `distance()`:
   - Send a pulse from the ultrasonic sensor
   - Measure the echo time to calculate distance in cm

6. Define a route `/` for the main dashboard:

   - Get temperature and humidity readings using `get_hum_temp_reading()`

   - Render the `options.html` template with the readings

7. Define a route `/home` for motion detection:

   - If the system is turned on (from session):

     - Call `distance()` to get the distance from the ultrasonic sensor

     - If distance < 10 cm, consider it motion detected:

       - Turn on buzzer for 1 second

       - Send HTTP request to the laptop server to trigger webcam capture

     - If no motion, ensure the buzzer is off

   - Render `home.html` template with the current distance and motion status

8. Define a route `/switch_on_system` to turn on the system:

   - Set session['system'] = True

   - Turn on the system LED

   - Redirect to `/home` page

9. Define a route `/switch_off_system` to turn off the system:

   - Set session['system'] = False

   - Turn off the system LED

   - Redirect to `/` page

10. Define routes `/switch_on_light/<pos>` and `/switch_off_light/<pos>` to control specific LEDs:

    - If LED is not already on/off, switch it on/off accordingly and update session

11. Define a route `/hum_temp` to get temperature and humidity readings:

    - Use Adafruit_DHT sensor to get readings

    - Return the readings as a list of [humidity, temperature]

12. Set up the Flask app to run on host `0.0.0.0`, port `5000`
```

13. Handle keyboard interrupt to clean up GPIO settings and stop the application

END