NUST

School of Electrical Engineering and Computer Science (SEECS)

Home Security System

Internet of Things

Instructor: Dr. Rafia Mumtaz

Presenters:

Sadaf Ambreen – 482183

Pakeeza Rasheed - 496433

Warda Bashir - 452455

Motivation

- The advent of the Internet of Things (IoT) has revolutionized various facets of modern living, with home automation standing out as one of the most significant applications

- Home security involves the **integration of various devices and systems** through a centralized control mechanism, enabling homeowners to manage and monitor their living environment efficiently

- By leveraging IoT sensors, home automation systems can **collect real-time data, process it, and execute actions** that enhance convenience, security, and energy efficiency

Background

Rigid & Inflexible: Difficult to customize or scale according to changing security needs

High Installation & Maintenance Costs: Often require professional installation and ongoing costly maintenance

No Integration with Other Systems: Cannot easily integrate with home automation, cameras, or other smart devices

Manual Operation: arming/disarming, alert checking must be performed manually on-site

Limited Data Logging and Analytics: No or minimal logging of security events for analysis or forensic use

Dependency on Physical Infrastructure: Often rely on wired connections and physical sensors, making upgrades difficult

Delayed Emergency Response: Emergency services might be alerted only after an alarm is triggered without further verification

Lack of Real-Time Alerts: Users often receive alerts only after physical alarms sound, limiting proactive responses

Limited Remote Access: Cannot be monitored remotely via smartphones or the internet

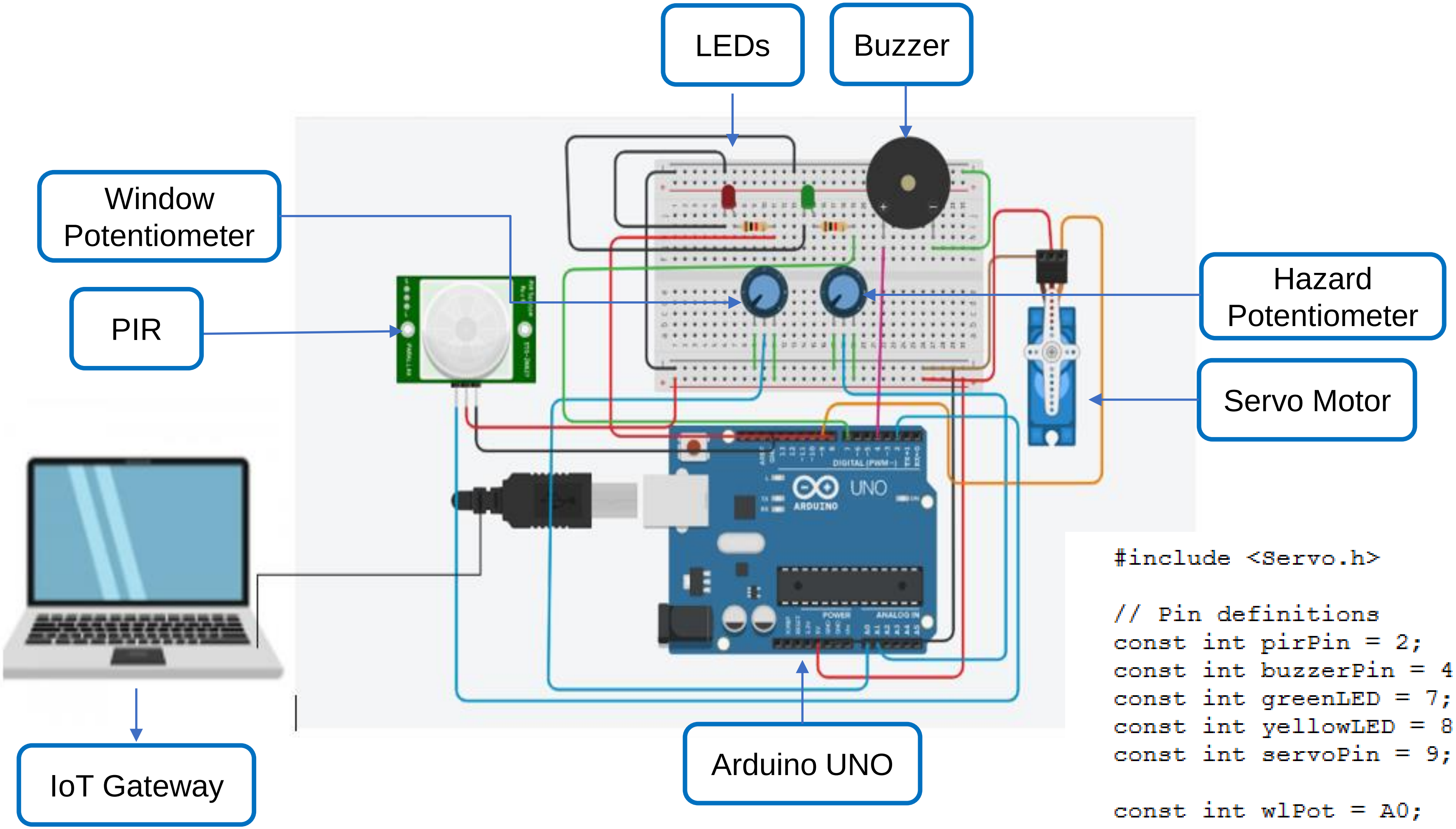
Sensor & Related Equipment

Component	Arduino Connection	Remarks
Hazard Potentiometer	A1 (analog input) Outer pins to 5V and GND, wiper(middle pin) to A1	This provides the necessary voltage for the potentiometer to operate.
Window Potentiometer	A0 (analog input) Outer pins to 5V and GND, wiper to A0	The wiper's voltage here indicates the position of the simulated window lock
PIR Motion Sensor	D2 (digital input) VCC to 5V, GND to GND, signal to D2	The signal pin outputs a digital signal (HIGH or LOW) depending on whether motion is detected
Servo Motor	D9 (PWM digital output) Power to 5V, GND to GND, signal to D9	D9 (this pin must be a PWM pin). PWM is a technique used to control the angle of the servo motor
Buzzer	D4 (digital output)	Through 220-ohm resistor to limit current and prevent damage
LEDs (Green, Yellow)	D7 (Green), D8 (Yellow)	A resistor is crucial here to limit the current and prevent the LED from burning out
USB Cable	Laptop USB port	Serial communication
Laptop Camera	Laptop integrated hardware	openCV and yagmail library to control the camera and capture images when the Arduino sends a signal

Tools and Technology

Software	Description
Arduino IDE	Software used to write, compile, and upload code to the Arduino
Visual Studio Code (VS Code)	VS Code is used to write and run Python scripts that handle image capture and sending alerts
Twilio's API	Allows the Arduino to send SMS alerts by sending HTTP requests to Twilio's API, which then delivers messages to a phone
Programming Languages	C++: Used for Arduino programming Python: Used for processing serial data, handling camera input, sending email/SMS alerts, and integrating APIs

Circuit Diagram

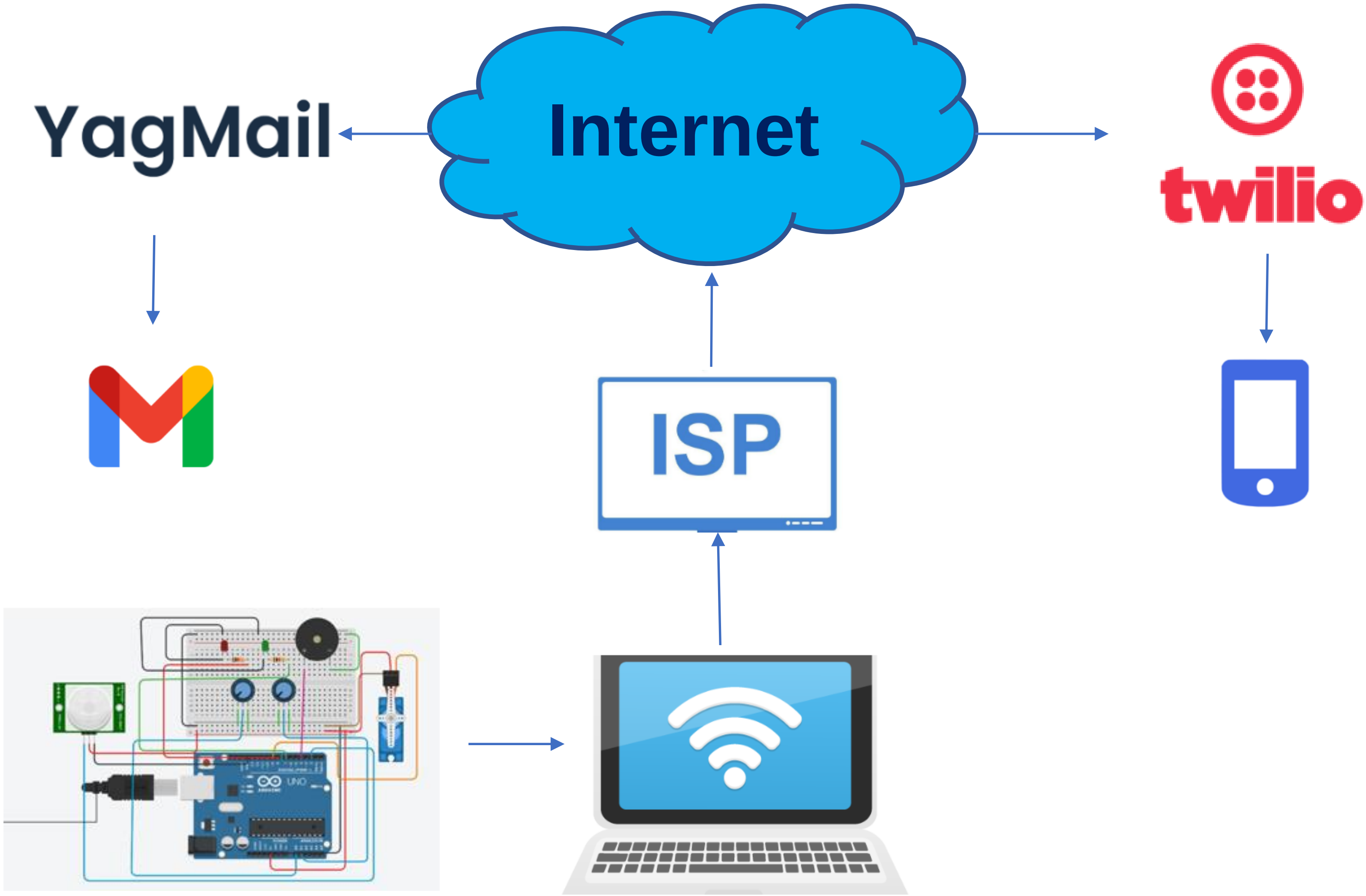


```
#include <Servo.h>

// Pin definitions
const int pirPin = 2;
const int buzzerPin = 4;
const int greenLED = 7;
const int yellowLED = 8;
const int servoPin = 9;

const int wlPot = A0;
const int hlzPot = A1;
```

Communication Technology



YagMail

Internet

twilio

ISP

Communication Technology & Network Topology

Communication	Description	Role in System	Requirements
Wi-Fi Module	Allows Arduino to connect to a Wi-Fi network for remote monitoring and control via the internet	Enables users to access the system globally using internet	Wi-Fi module; Internet connection; Laptop acts as Wi-Fi gateway
SMS API (Twilio)	Third-party web service API for sending SMS alerts triggered by the Arduino	Sends SMS notifications to users' phones during security events	Internet connectivity (Wi-Fi or GSM); API credentials; HTTP requests from Arduino or laptop
USB Connection to Laptop	Wired serial communication between Arduino and laptop	Used for debugging, data logging, sending commands, and triggering camera captures	USB cable; Serial communication software on laptop
Network Topology Star	Network architecture with Arduino as central hub communicating with sensors, actuators, and internet via Wi-Fi router	Provides scalable, manageable, and performant network structure	Arduino or advanced microcontroller; Wi-Fi router; connected devices

Co Relation with OSI Model

Layer	Technology Used	Justification
Physical Layer	USB Serial (Arduino → Laptop)	Reliable and simple connection between microcontroller and host
Internet Layer	Laptop Internet via Wi-Fi	Required for sending alerts to cloud (email/SMS)
Application Layer	HTTP API (Python)	Used to interact with Email (SMTP) and SMS APIs (e.g., Twilio)

Methodology

PIR Motion Sensor Detects human motion near the property

Normal (no motion):

-Green LED ON (safe condition)

-Buzzer OFF.

When Motion Detected:

-Red LED ON.

-Buzzer ON.

-Door is locked via servo motor, unless hazard level is high

-Triggers camera snapshot and alert

Window Lock Status – Potentiometer (A0)

Simulates window lock position (0–100%)

Classified into 6 states:

% Open	Status	Action
0–5	Fully Closed	Safe
6–15	Mostly Closed	Safe
16–30	Slightly Open	Safe
31–70	Partially Open	Yellow LED ON, log "Caution"
71–95	Mostly Open	Yellow LED ON, log "Risk"
96–100	Fully Open	Red LED + Buzzer ON, log "Security Breach

Methodology

Hazard Level – Potentiometer (A1)

Simulates environmental hazard level (0–100%).
5 classified levels:

% Level	Status	Action
0–10	Safe	Green LED ON
11–30	Low Risk	Green LED ON
31–60	Moderate Risk	Yellow LED ON, monitor only
61–85	High Risk	Red LED ON, Buzzer, Door unlocks for emergency
86–100	DANGER!	Red LED ON, Buzzer, Door unlocks, snapshot + alert triggered

Servo Door Lock Mechanism

Locked:
-On motion detection, when hazard is low.
Unlocked:
-Automatically when hazard level is “High Risk” or “DANGER” for emergency exit
Re-locks once danger is cleared and environment returns to safe

Methodology

🔊 Alert System

Buzzer (pin 4)

Activated in any of these cases:

-Motion detected

-Hazard Level ≥ 3

-Window Lock Status ≥ 4

Automatically turns OFF after 2 seconds in moderate cases.

💡 LED Indicators

LED	Color	Condition Trigger
Green	Green	No danger, no motion, window & hazard in safe range
Red	Red	Moderate hazard or partially open window and Motion detected or serious risk (hazard or window)

Python Integration Features

Webcam (OpenCV)

Takes a snapshot when motion is detected

File saved locally with a timestamped name

Methodology

Email Alert (Yagmail) Sent if danger is detected

Includes:

-Detailed log message

-Webcam snapshot (if available)

SMS Alert (Twilio) Sends a brief message to the user

Notifies about the alert and suggests checking email

Conditions That Trigger Alerts

Motion + High Hazard

Window "Security Breach"

Hazard Level “High Risk” or above

Alert Reset

Once danger is cleared or motion stops, system resets and can send new alerts later if needed

Timing and Reporting

Sensor readings every 5 seconds.

Full status report printed to serial monitor every 60 seconds, showing:

-Window Lock %

-Hazard Level %

-Door Lock Status

-

Advantages

Cost-Effective:

Uses inexpensive, readily available components, making it affordable
Suitable alternative for users with limited budgets

Customizable:

Flexible Arduino platform allows easy addition/removal of sensors & Users can modify code as security needs

Easy to Prototype:

Designed for experimentation and rapid testing
Modular components and Arduino IDE simplify development

Real-Time Response:

Quickly detects and responds to potential threats & Minimizes damage by prompt alerts and actions

Integration Capabilities:

Can be connected with other home automation systems & Enables seamless interaction with lighting, climate control, and more

Image Capture:

Laptop camera integration allows capturing evidence & it Helps deter crime and assists in investigations

Educational Value:

Provides hands-on experience with microcontrollers and sensors
Enhances skills in electronics, programming, and security systems

Challenges & Limitations

Arduino's Limited Resources:
Limited processing power and memory restrict system complexity and number of devices
Struggles with complex calculations or managing many sensors simultaneously

Lack of Built-in Wireless:
No native Wi-Fi or Bluetooth on Arduino Uno
Requires external modules, increasing cost and complexity

Power Consumption: Running on batteries requires efficient power management for long-term operation

Scalability: Expanding coverage or features may require more powerful microcontrollers or distributed systems

Security Considerations: Necessitates encryption and authentication to prevent unauthorized access

User Interface:
Basic system lacks an advanced or user-friendly interface
Adding displays or mobile apps would improve usability

Laptop Dependency:
Image capturing relies on laptop connection, limiting portability
Self-contained solutions with onboard storage/display could be preferable

.

Conclusion

Developed a versatile home security system based on Arduino Uno

Integrated multiple sensors and actuators, including a laptop camera for image capture

System detects various threats and responds with multiple alert mechanisms

Despite limitations, provides a solid foundation for advanced, customized solutions

.

Future Work

Advanced Security Features:

Add biometric sensors (fingerprint, facial recognition) for authentication
Develop intelligent algorithms using machine learning to reduce false alarms
Implement active deterrents like sirens, strobe lights, or voice warnings

Improved User Interface:

Design user-friendly mobile app or web interface for control and monitoring
Add local display (e.g., LCD) on Arduino for status updates
Incorporate voice control for hands-free operation

Integration with Smart Home Ecosystems:

Connect with smart locks, lighting, surveillance cameras for automation

Power Optimization:

Use power-saving modes and efficient management to extend battery life

Increased Scalability:

Support more sensors and devices for larger homes or commercial use

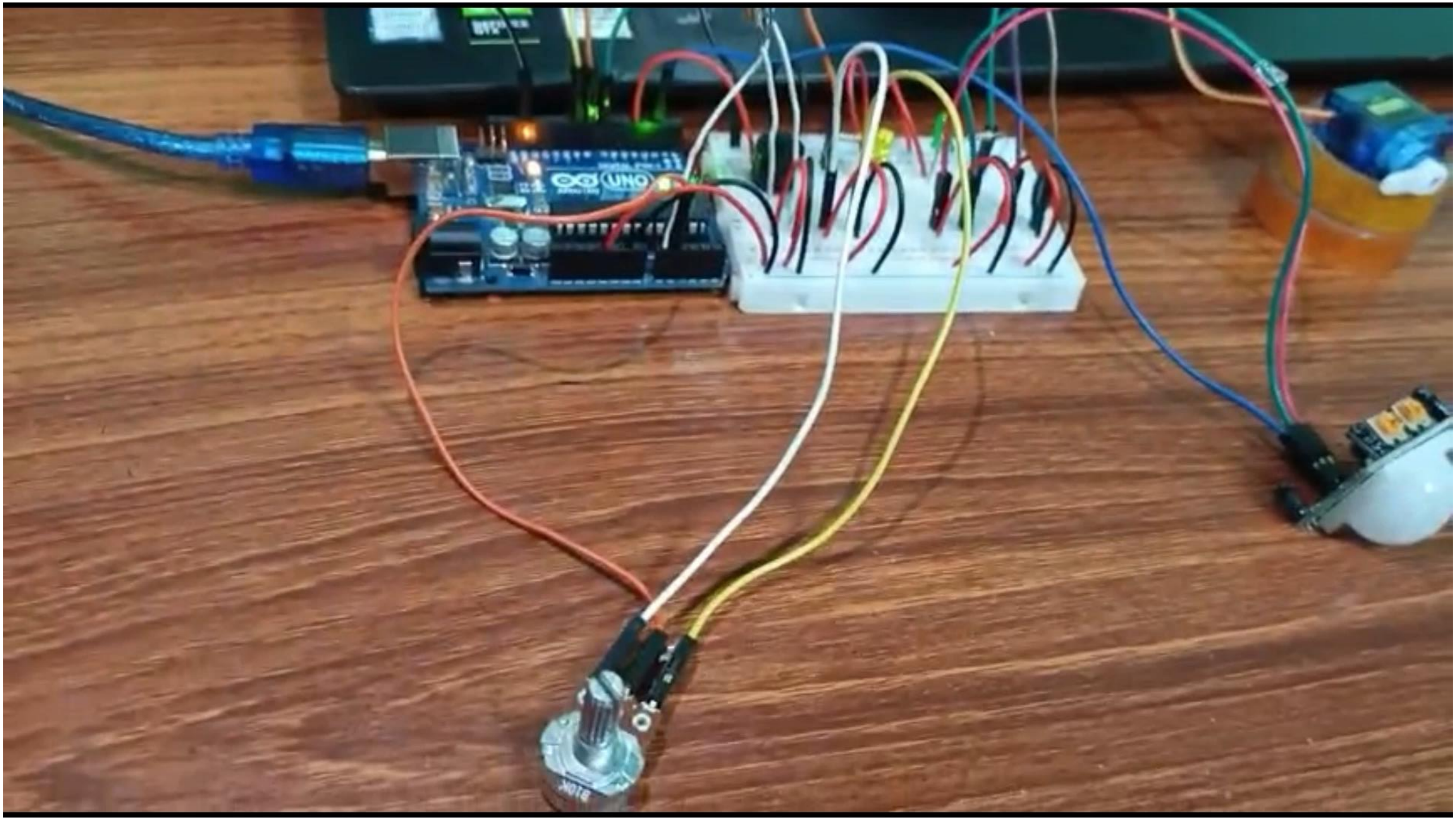
Self-Contained Operation:

Replace laptop dependency with onboard storage (e.g., SD card) and display for images

Cloud Integration:

Store data and logs in the cloud for remote access, analysis, backups, diagnostics, and updates

Demo



THANK YOU