HACKINFINITY

PROBLEM STATEMENT TITLE: Vigileye Intrusion Alert System

PROTOTYPE NAME: Intruguard

THEME: Home Security

TEAM NAME: Apex Innovators

TEAM LEADER: Leeshanth.N

TEAM MEMBERS: Naresh Kumar.R

Nivethalakshmi.C.B

Harini. R

HACKINFINITY

Hardware components used:

Component (General Name)	Model / Specification
Microcontroller	Glyph H2
Ultrasonic Sensor	HC-SR04
IR Sensor	Generic IR Sensor Module
CAP1203 Touch Sensor	CAP1203
DHT22 (Temperature & Humidity)	DHT22
SW-420 Vibration Sensor	SW-420
Connecting wires	Jumper wires / standard hookup wires
Breadboard	Standard solderless breadboard
Power supply	3.3 V regulated supply
Resistors	10 kΩ, 4.7 kΩ, 220 Ω (optional for LEDs)

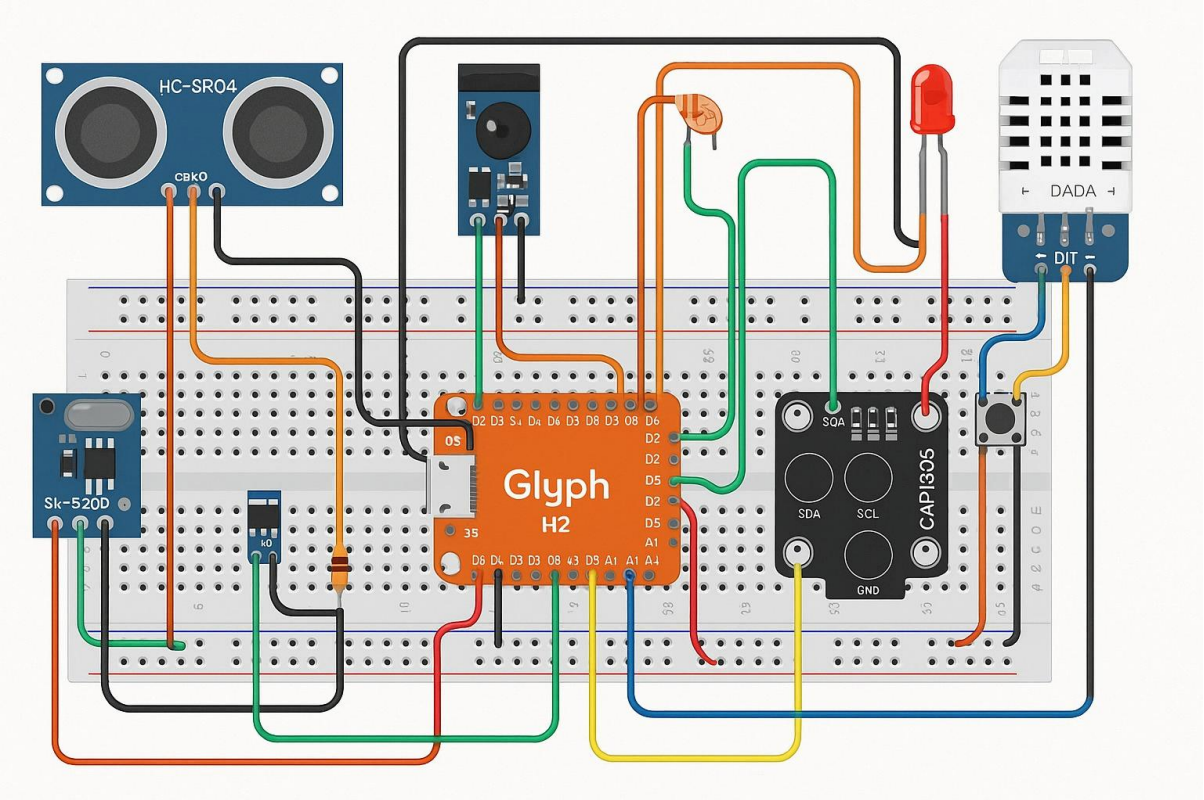
HACKINFINITY

Pin configuration and circuit connections:

Component	Pin on Component	Pin on Glyph H2	Additional Notes / Resistor
Ultrasonic Sensor (TRIG)	TRIG	GPIO5	—
Ultrasonic Sensor (ECHO)	ECHO	GPIO4	10 kΩ + 20 kΩ voltage divider to reduce 5 V → 3.3 V
IR Sensor	OUT	GPIO12	—
CAP1203 Touch Sensor	SDA	GPIO2	I^2C SDA
CAP1203 Touch Sensor	SCL	GPIO3	I^2C SCL
DHT22 Temperature & Humidity	DATA	GPIO14	4.7 kΩ pull-up to 3.3 V
SW-420 Vibration Sensor	OUT	GPIO13	Optional 10 kΩ pull-down
SW-420 Vibration Sensor	VCC	3.3 V	—
SW-420 Vibration Sensor	GND	GND	—
Power Supply	3.3 V / GND	3.3 V / GND	—

HACKINFINITY

Circuit design/schematic diagrams:



Code:

```
#include <DHT.h>
#include <Wire.h>
#include "SparkFun_CAP1203.h"

// ====== DHT22 setup ======
#define DHTPIN 2       // GPIO2
#define DHTTYPE DHT22
DHT dht(DHTPIN, DHTTYPE);

// ====== IR sensor ======
#define IR_PIN 12      // GPIO12 (active-low)
```

HACKINFINITY

```
// ====== Ultrasonic sensor ======
#define TRIG_PIN 5     // GPIO5
#define ECHO_PIN 4     // GPIO4

// ====== CAP1203 touch sensor ======
#define SDA_PIN 8
#define SCL_PIN 9
CAP1203 capSensor;

// ====== Tilt Sensor ======
#define TILT_PIN 13   // GPIO13 connected to SW-520D

// ====== LEDs ======
#define RED_LED   23   // GPIO23 → 220Ω → Red LED → GND
#define GREEN_LED 24   // GPIO24 → 220Ω → Green LED → GND

void setup() {
  Serial.begin(115200);
  // IR
  pinMode(IR_PIN, INPUT);
  // Ultrasonic
  pinMode(TRIG_PIN, OUTPUT);
  pinMode(ECHO_PIN, INPUT);

  // DHT22
  dht.begin();
  dht.readHumidity(); // dummy read
```

```
  // CAP1203
  Wire.begin(SDA_PIN, SCL_PIN);
  if (!capSensor.begin()) {
    Serial.println("CAP1203 not connected. Check wiring.");
    while (1);
  }

  // Tilt Sensor
  pinMode(TILT_PIN, INPUT_PULLUP);

  // LEDs
  pinMode(RED_LED, OUTPUT);
  pinMode(GREEN_LED, OUTPUT);

  Serial.println("Glyph H2 - Sensors + LED (Ultrasonic Ignored) Test Started");
  Serial.println("------------------------------");
}
void loop() {
  bool alert = false;  // assume safe

  // --- IR Sensor ---
  int irState = digitalRead(IR_PIN);
  if (irState == LOW) {   // active-low
    Serial.println("IR: Object detected ");
    alert = true;
  } else {
    Serial.println("IR: No object ");
  }
```

HACKINFINITY

```
// --- Ultrasonic Sensor (ignored for LED logic) ---
digitalWrite(TRIG_PIN, LOW);
delayMicroseconds(2);
digitalWrite(TRIG_PIN, HIGH);
delayMicroseconds(10);
digitalWrite(TRIG_PIN, LOW);

long duration = pulseIn(ECHO_PIN, HIGH, 30000); // 30ms timeout
float distance = duration * 0.034 / 2;          // in cm
if (duration == 0) {
  Serial.println("Ultrasonic: Out of range");
} else {
  Serial.print("Ultrasonic: "); Serial.print(distance); Serial.println(" cm");
}


// --- DHT22 Sensor ---
float h = dht.readHumidity();
float t = dht.readTemperature();
if (isnan(h) || isnan(t)) {
  Serial.println("DHT22: Failed to read ");
  alert = true;
} else {
  Serial.print("DHT22: Temp: "); Serial.print(t);
  Serial.print("°C, Humidity: "); Serial.print(h); Serial.println("%");
}
// --- CAP1203 Sensor ---
bool touchState = capSensor.isTouched();
if (touchState) {
```

```
  Serial.println("CAP1203: Touch detected ");
  alert = true;
 } else {
  Serial.println("CAP1203: No touch");
 }
 // --- Tilt Sensor ---
 int tiltState = digitalRead(TILT_PIN);
 if (tiltState == HIGH) {   // HIGH = Tilt detected
  Serial.println("Tilt: Detected ");
  alert = true;
 } else {
  Serial.println("Tilt: Not tilted");
 }

 // --- LED Control ---
 if (alert) {
  digitalWrite(RED_LED, HIGH);
  digitalWrite(GREEN_LED, LOW);
 } else {
  digitalWrite(RED_LED, LOW);
  digitalWrite(GREEN_LED, HIGH);
 }
 Serial.println("-----------------------------");
 delay(500); // delay for stability
}
```

HACKINFINITY

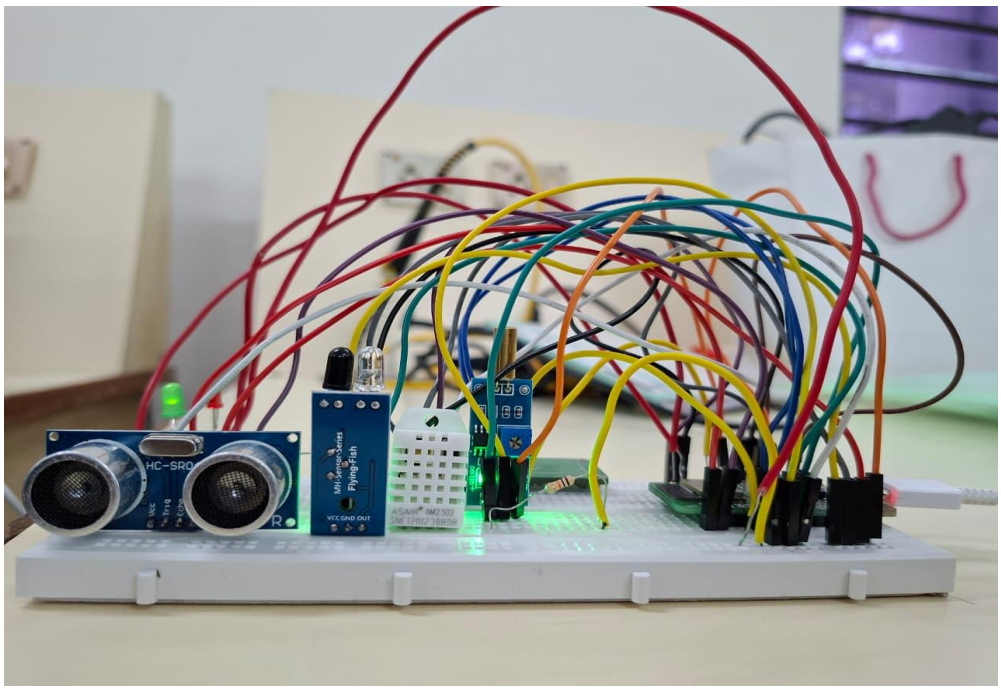
Software / Tools Used:

Arduino IDE – for programming the Glyph H2 microcontroller

Visual Studio Code – for writing dashboard code

HTML, CSS, JavaScript – for creating the web dashboard interface

Pictures of the setup/prototype:



Working explanation and output:

Overview:
The Vigil Eye Intrusion Alert System is a multi-sensor security solution designed to detect unauthorized entry, tampering, and environmental changes in real-time. It integrates an ultrasonic sensor and IR sensor for accurate entry detection, a vibration sensor (SW-420) for tamper alerts, and a CAP12r03 touch sensor for touch detection. The system also monitors temperature and humidity using a DHT22 sensor.

Data can be viewed through a web dashboard created using HTML, CSS, and JavaScript, providing real-time alerts and environmental monitoring. This compact, low-cost system is suitable for homes, offices, and public spaces, combining safety, hygiene, and convenience in one platform.

HACKINFINITY

Entry Detection:

- The Ultrasonic Sensor (HC-SR04) measures the distance of approaching objects.
- The IR Sensor confirms the presence of a person.
- When IR detects an object (regardless of ultrasonic reading), the system triggers the Red LED alert.
- If no detection occurs, the Green LED remains ON, indicating a safe/idle state.

Vibration / Tamper Detection:

- The SW-420 Vibration Sensor detects any vibration or tampering at the door or device.
- When vibration is detected, the system immediately triggers the Red LED alert.
- If no vibration is detected, the system continues to show Green LED, indicating normal status.

Touch Reset:

- The **CAP1203 Touch Sensor** allows a touch-based input.
- When touched, it triggers the **Red LED alert**, simulating an alert action.
- When not touched, it reports **No touch detected**.
- Touch can be used to reset alert counters or acknowledgment in some designs.

.

Environmental Monitoring:

- The DHT22 Sensor continuously reads temperature and humidity.
- If readings are successful, they are printed to the Serial Monitor.
- If reading fails, an alert is triggered.

Environmental data can be logged or displayed on a dashboard for remote monitoring.

HACKINFINITY

Output:

Applications and other relevant details:

Smart Home:

- Sensor data can be sent to a dashboard or mobile app.
- Supports automation rules like turning on fans, alarms, or notifications when certain conditions are detected.
- Forms the basis of a home security and monitoring system.

- **Security and Tamper Detection:**
- SW-420 Vibration Sensor detects tampering or unauthorized access attempts.
- CAP1203 Touch Sensor can be used to trigger manual alerts or reset systems.
- Provides an early warning system for doors, cabinets, or sensitive devices.
-

- **Educational / Experimental Use:**
- Demonstrates integration of multiple sensors in a microcontroller environment.
- Helps students and hobbyists learn about sensor interfacing, real-time monitoring, and alert systems