



FACULTATEA DE
**AUTOMATICĂ ȘI
CALCULATOARE**



Proiect MAȘINI ELECTRICE ȘI ACȚIONĂRI

**Sistem inteligent de supraveghere bazat
pe senzor de mișcare**

Studenti:

- *Iurea Andreea-Irina*
- *Tomescu Adriana-Marinela*
- *Smărăndescu Antonio-Ștefan*

Grupa: 324 AC

1. Introducere

În era digitală, tehnologiile de automatizare și supraveghere sunt din ce în ce mai utilizate în diverse domenii, de la securitate la monitorizarea mediului. Proiectul de față explorează integrarea acestor tehnologii într-un sistem accesibil și eficient, utilizând o placă Arduino și alte componente esențiale pentru a dezvolta un sistem de supraveghere inteligent. Acest sistem funcționează prin detectarea mișcării și răspunde prin activarea unui buzzer, aprinderea unui LED și mișcarea servomotoarelor care susțin o cameră ESP32, oferind o privire de ansamblu asupra încăperii. Camera este conectată la telefonul mobil, permițând vizualizarea de la distanță și asigurând monitorizarea constantă a spațiului pentru a preveni evenimentele neprevăzute.

Obiectiv: Scopul principal al proiectului este de a crea un sistem automatizat capabil să detecteze și să urmărească deplasarea unei persoane utilizând un modul ESP32-CAM pentru capturarea imaginii și un Arduino Uno pentru controlul servomotoarelor responsabile de mișcarea camerei.

2. Contribuțiile Fiecărei Persoane din Proiect

- **Tomescu Adriana-Marinela:**
 - Dezvoltarea și implementarea codului pentru Arduino și ESP32-CAM.
 - Configurarea și programarea servomotoarelor astfel încât să se miște atunci când este detectată mișcarea.
 - Coordonarea echipei și gestionarea resurselor proiectului.
 - Pregătirea prezentării și a materialelor vizuale pentru susținerea proiectului.
- **Iurea Andreea-Irina:**
 - Montarea hardware-ului și asamblarea componentelor pe breadboard.
 - Testarea și calibrarea servomotoarelor pentru mișcări precise.
 - Punerea în funcțiune a modului ESP32-CAM și lipirea acestuia pe servomotoare.
 - Pregătirea prezentării și a materialelor vizuale pentru susținerea proiectului.
- **Smărăndescu Antonio-Ștefan:**
 - Documentarea proiectului și realizarea referatului.
 - Pregătirea prezentării și a materialelor vizuale pentru susținerea proiectului.

- Realizarea conexiunilor electrice între componente și verificarea lor.
- Montarea hardware-ului și asamblarea componentelor pe breadboard.

3. Obiectivele Proiectului Propus

- **Obiective principale:**

- Dezvoltarea unui sistem capabil să detecteze și să urmeze orice tip de mișcare folosind tehnologii accesibile.
 - Asigurarea unei soluții stabile și precise pentru detectarea și urmărirea unui posibil eveniment neplăcut .
 - Crearea unui algoritm de control pentru mișcarea fluidă și precisă a servomotoarelor.
 - Supravegherea unei suprafețe mai mari datorită servomotoarelor pe care este poziționată camera.
 - Supravegherea precisă a mișcării persoanelor datorită senzorilor de atenționare (led și buzzer).

- **Obiective secundare**

- Învățarea și Aplicarea Tehnicilor de Control al Servomotoarelor
 - Proiectul oferă oportunitatea de a învăța și aplica cunoștințe despre controlul precis al servomotoarelor, integrarea acestora cu un microcontroller și programarea acestuia.
- Dezvoltarea Abilităților de Proiectare și Construcție Hardware
 - Proiectarea și construirea unui dispozitiv hardware complex, care combină electronică, mecanică și programare.
- Optimizarea proiectului astfel încât să fie util, ușor de folosit și să poată fi urmărit de la distanță cu ajutorul telefonului
 - Conectarea camerei la telefonul mobil.

4. Descrierea domeniului ales (SMART HOUSE) și a soluțiilor similare

Domeniul "Smart House" se referă la integrarea tehnologiei avansate în locuințe pentru a automatiza, monitoriza și controla diverse sisteme și dispozitive. Acest domeniu acoperă o gamă largă de aplicații care contribuie la creșterea eficienței, confortului și securității locuințelor moderne.

Aplicațiile de comandă și control din domeniul "Smart House" permit utilizatorilor să gestioneze și să monitorizeze diverse aspecte ale locuinței prin intermediul dispozitivelor mobile, tablete sau asistenți vocali.

În domeniul supravegherii, al identificării și al verificării persoanelor, inovația este continuă și diversificată cu numeroase soluții care își propun să ducă acest domeniu la un alt nivel. Câteva dintre aceste soluții sunt:

Automatizarea Iluminatului în Funcție de Prezență:

- **Descriere:** Sistemele de iluminat inteligent care folosesc senzori de mișcare și recunoaștere facială pentru a aprinde și a ajusta luminile în funcție de prezența și preferințele utilizatorilor.

Asistenți personali:

- **Descriere:** Dispozitive inteligente care folosesc recunoașterea facială și alte tehnologii de detecție pentru a urmări mișcările utilizatorilor și pentru a oferi servicii personalizate, cum ar fi controlul vocal al dispozitivelor casnice sau gestionarea activităților zilnice.

Termostate Inteligente:

- **Descriere:** Termostatele inteligente folosesc recunoașterea facială pentru a detecta prezența utilizatorilor și pentru a ajusta temperatura în funcție de preferințele acestora și de programul zilnic.

Oglinzi Inteligente:

- **Descriere:** Oglinzile inteligente sunt echipate cu recunoaștere facială și pot oferi informații personalizate, cum ar fi prognoza meteo, calendarul zilnic și știri.

În acest context, proiectul realizat de echipa noastră propune un mod diferit de supraveghere care îmbină elemente mecanice, digitale și electronice, fiind diferit de sistemele clasice prin posibilitatea de a roti camera pentru a avea o vizibilitate crescută, dar și prin automatizarea motoarelor pentru a putea urmări activitatea din toate unghiurile unei încăperi, dacă în acea încăpere este detectată mișcare.

5. Descrierea soluției propuse

Proiectul pe care l-am dezvoltat îmbină mai multe tehnologii pentru a crea un sistem de securitate inteligent, eficient și ușor de utilizat. Soluția constă într-un ansamblu de senzori și dispozitive montate pe o platformă Arduino, având ca scop monitorizarea și supravegherea unei încăperi. Iată cum funcționează acest sistem:

1. Senzorul de Mișcare:

- În centrul acestui proiect se află un senzor de mișcare care detectează prezența oricărei mișcări în încăpere. Acesta este esențial pentru a iniția reacția sistemului de securitate.

2. Activarea Buzzerului și LED-ului:

- Imediat ce senzorul detectează mișcare, un buzzer se activează pentru a alerta în mod sonor despre prezența unei activități neautorizate. În același timp, un LED se aprinde pentru a oferi o semnalizare vizuală a detecției.

3. Servomotoare și Camera ESP32:

- O pereche de servomotoare intră în acțiune, mișcând camera ESP32 montată pe ele. Aceste mișcări permit camerei să ofere o privire panoramică asupra încăperii, capturând imagini din diverse unghiuri.

4. Conectivitatea și Vizualizarea Distantă:

- Camera ESP32 este conectată la un telefon mobil prin intermediul adresei IP. Această conexiune permite utilizatorului să acceseze și să vizualizeze în timp real imaginile transmise de cameră, indiferent de locul în care se află. Astfel, utilizatorul poate monitoriza în permanență încăperea, reducând riscul unor evenimente neplăcute.

Beneficiile Soluției

- **Monitorizare Continuă:** Sistemul asigură o supraveghere continuă a încăperii, oferind utilizatorului un sentiment de siguranță și control.
- **Alerte Imediate:** Activarea imediată a buzzerului și LED-ului la detectarea mișcării permite o reacție rapidă în caz de intruziune.
- **Vizualizare Panoramică:** Mișcările servomotoarelor asigură o acoperire completă a încăperii, eliminând zonele moarte.

- **Accesibilitate de la Distanță:** Conectivitatea cu telefonul mobil permite utilizatorului să monitorizeze încăperea de la distanță, oferind flexibilitate și confort.

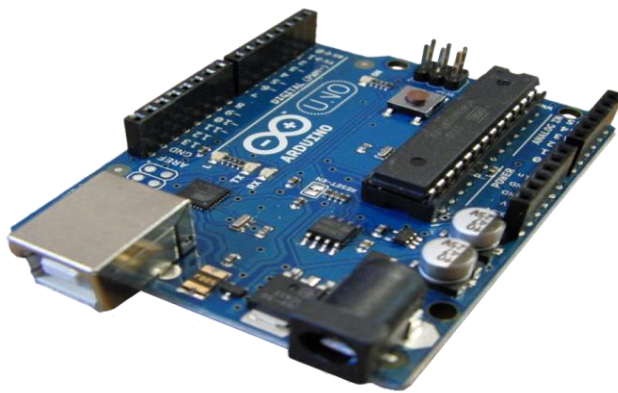
Acest sistem de securitate inteligent este ideal pentru protejarea spațiilor personale sau comerciale, combinând tehnologii avansate într-o soluție simplă și eficientă.

6. Descrierea solutiei implementate cu prezentarea functionalitatilor aferente solutiei

Componente cheie :

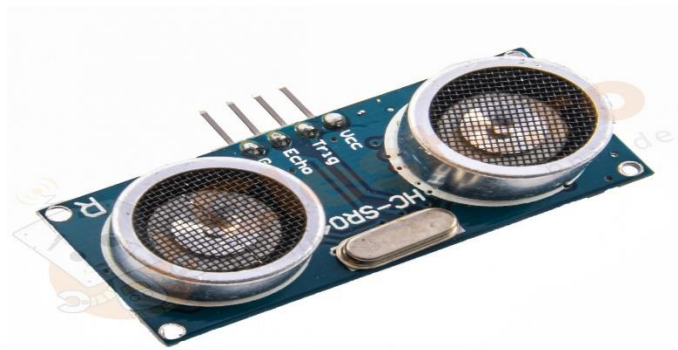
1. Arduino UNO:

- **Rol:** Este microcontrolerul principal care gestionează toate componentele și logica proiectului. Arduino coordonează activarea buzzerului, LED-ului, mișcarea servomotoarelor și controlul camerei ESP32 în funcție de semnalele primite de la senzorul de mișcare.



2. Senzorul Ultrasonic (HC-SR04):

- **Rol:** Detectează prezența mișcării în încăpere. La detectarea mișcării, senzorul trimite un semnal către Arduino, care declanșează restul acțiunilor sistemului de securitate.



3. **Buzzer:**

- **Rol:** Emite un sunet de alertă atunci când senzorul de mișcare detectează activitate. Acest sunet servește ca o avertizare imediată pentru oricine se află în apropiere.



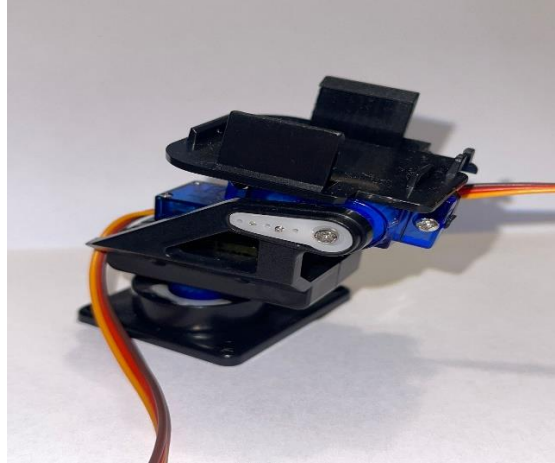
4. **LED:**

- **Rol:** Se aprinde pentru a oferi o indicare vizuală a detectării mișcării. LED-ul ajută la confirmarea vizuală că senzorul a detectat mișcare și că sistemul este activ.



5. **Servomotoare + montură servomotor suport cameră 2 axe:**

- **Rol:** Asigură mișcarea camerei ESP32, permițând acesteia să se rotească și să captureze imagini din diverse unghiuri ale încăperii. Mișcarea servomotoarelor este controlată de Arduino.



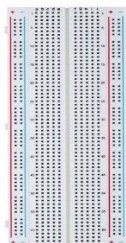
6. Camera ESP32:

- **Rol:** Capturează și transmite imagini în timp real către telefonul mobil. Este montată pe servomotoare pentru a oferi o vizualizare panoramică a încăperii.



7. Breadboard:

- **Rol:** Servește ca platformă de conexiuni pentru componentele electronice. Breadboard-ul facilitează conectarea senzorului, buzzerului, LED-ului, servomotoarelor și camerei la Arduino fără a necesita lipirea componentelor.



8. Baterii:

- **Rol:** Asigură alimentarea întregului sistem. Bateriile furnizează energia necesară pentru funcționarea Arduino-ului, senzorului de mișcare, buzzerului, LED-ului, servomotoarelor și camerei.



9. Cablu USB:

- **Rol:** Conectează Arduino la laptop pentru programare și alimentare. Este esențial pentru încărcarea codului pe Arduino și pentru testarea sistemului.



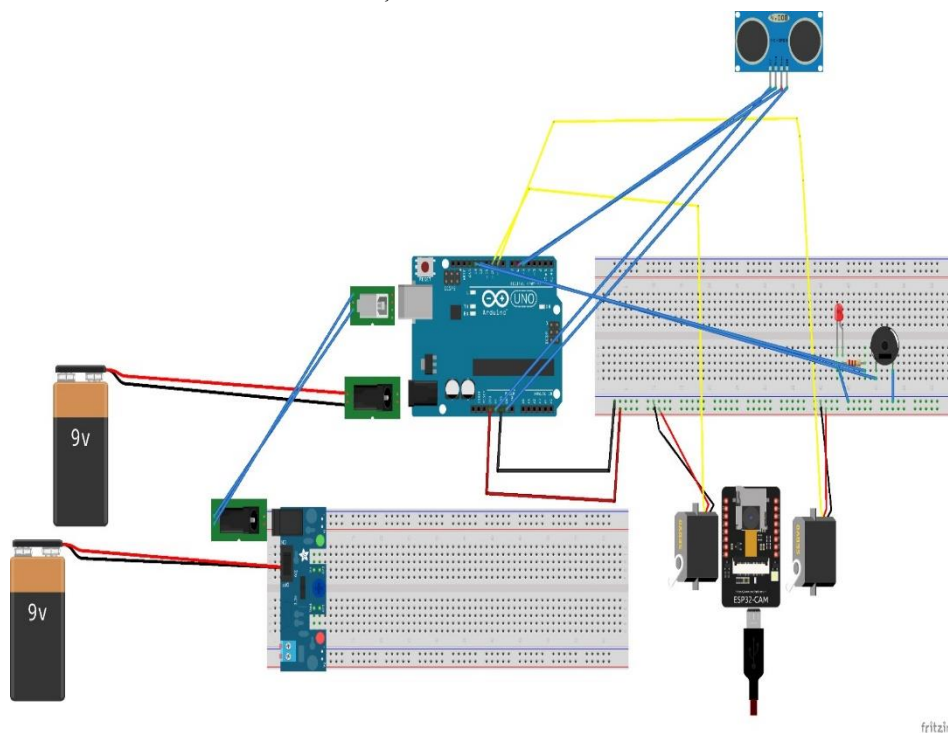
10. Modul Sursa de Alimentare:

- **Rol:** Asigură o alimentare stabilă și reglabilă pentru componentele electronice de pe breadboard. Acesta convertește tensiunea de la o sursă externă (de exemplu, un adaptor de 9V sau 12V) la o tensiune adecvată (5V sau 3.3V) pentru alimentarea circuitelor și componentelor conectate la breadboard.



Configurația și Funcționarea Sistemului

- **Configurare:** Componentele sunt conectate la Arduino folosind breadboard-ul pentru a simplifica realizarea conexiunilor. Sensorul de mișcare este poziționat într-un loc optim pentru a detecta orice intrare în încăpere. Camera ESP32 este montată pe servomotoare pentru a avea libertatea de mișcare necesară. Cablu USB este folosit pentru a programa Arduino și pentru a oferi o sursă de alimentare inițială sau de backup. Modulul sursă de alimentare este utilizat pentru a asigura o tensiune stabilă și corespunzătoare pentru toate componentele conectate la breadboard.
- **Funcționare:**
 1. Sensorul de mișcare detectează mișcarea și trimite un semnal către Arduino.
 2. Arduino activează buzzerul și aprinde LED-ul pentru a alerta despre prezența unei activități.
 3. Simultan, Arduino controlează servomotoarele pentru a mișca camera ESP32, oferind o vizualizare completă a încăperii.
 4. Camera ESP32 transmite imagini în timp real către telefonul mobil, permițând monitorizarea de la distanță.



Componente și Conexiuni:

1. **Arduino:**
 - **Alimentare:**
 - Conectat la sursa de alimentare prin jack-ul de alimentare (alimentat de la baterii de 9V).
 - **GND:** Conectat la linia de împământare a breadboard-ului.

2. **Buzzer:**

- **Positiv (VCC):** Conectat la pinul digital 12 al Arduino.
- **Negativ (GND):** Conectat la linia de împământare a breadboard-ului.

3. **LED:**

- **Anod (lung):** Conectat la un rezistor de aproximativ 220 ohmi, iar celălalt capăt al rezistorului la pinul digital 13 al Arduino.
- **Catod (scurt):** Conectat la linia de împământare a breadboard-ului.

4. **Servomotoare:**

- **Servomotor Vertical:**
 - **VCC:** Conectat la linia de alimentare de 5V a breadboard-ului.
 - **GND:** Conectat la linia de împământare a breadboard-ului.
 - **Semnal:** Conectat la pinul digital 10 al Arduino.
- **Servomotor Orizontal:**
 - **VCC:** Conectat la linia de alimentare de 5V a breadboard-ului.
 - **GND:** Conectat la linia de împământare a breadboard-ului.
 - **Semnal:** Conectat la pinul digital 9 al Arduino.

5. **Modul Sursa de Alimentare:**

- Conectat la breadboard pentru a distribui tensiunea de 5V și GND către toate componentele.

6. **Baterii de 9V:**

- Conectate la modulul sursă de alimentare pentru a furniza energia necesară întregului sistem.

Mici Greutăți Întâmpinate

În timpul realizării și testării proiectului, au fost întâmpinate câteva dificultăți minore, după cum urmează:

1. Formatarea Cardului pentru Cameră:

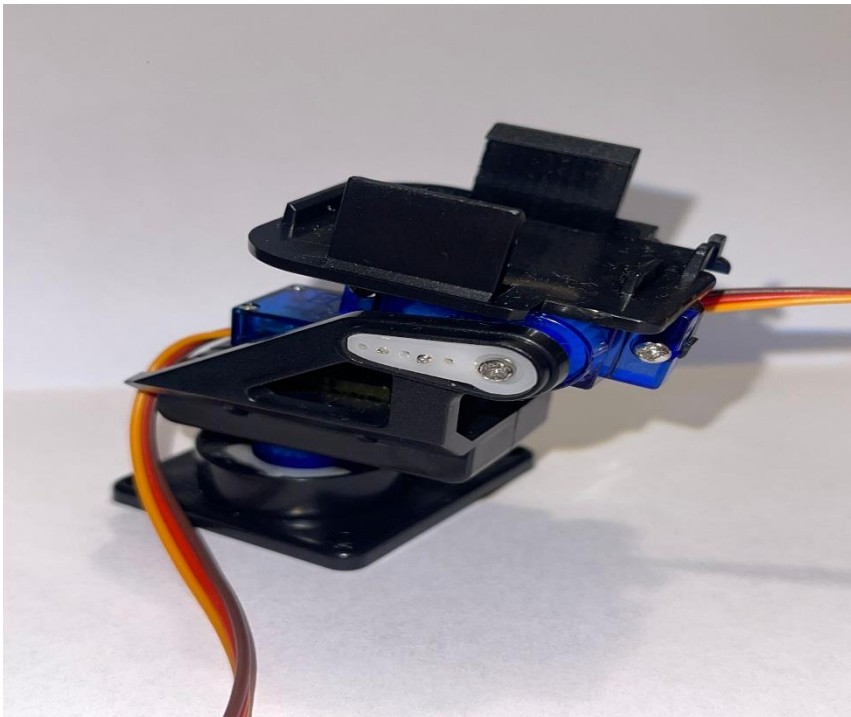
- A fost necesară formatarea corectă a cardului SD pentru camera ESP32, ceea ce a necesitat câteva încercări până când acesta a fost recunoscut și a funcționat corespunzător.

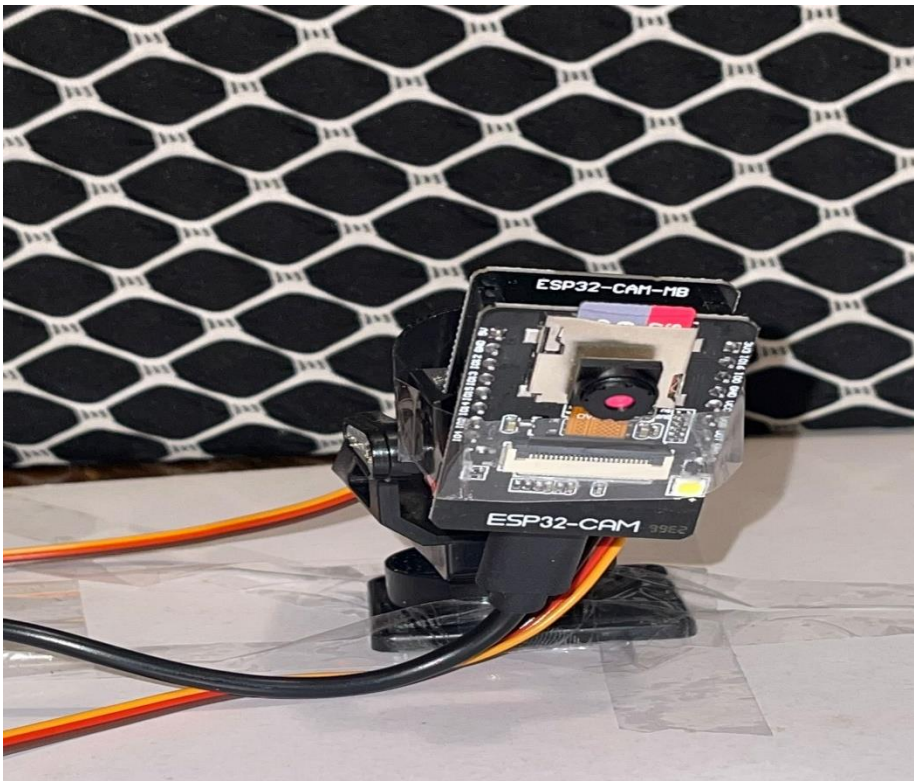
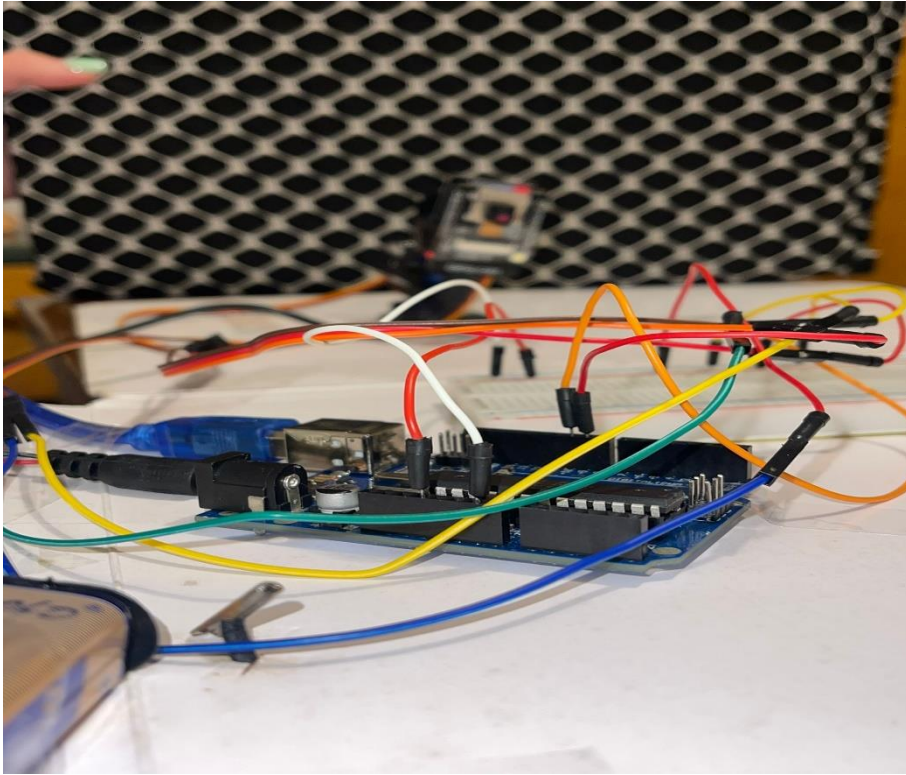
2. Funcționarea Servomotoarelor:

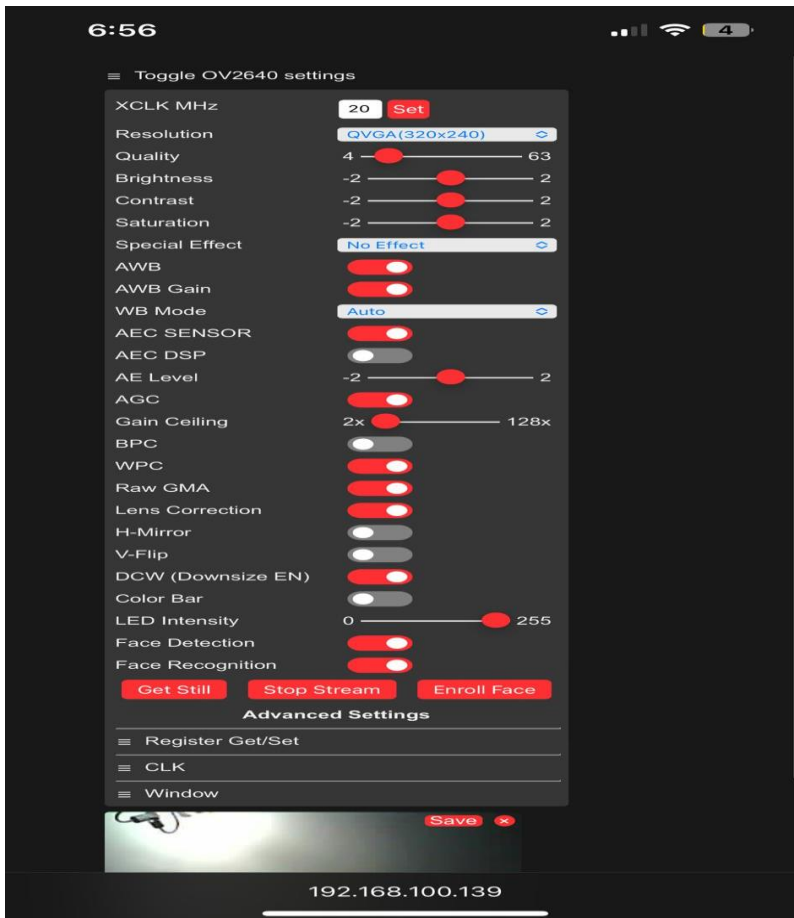
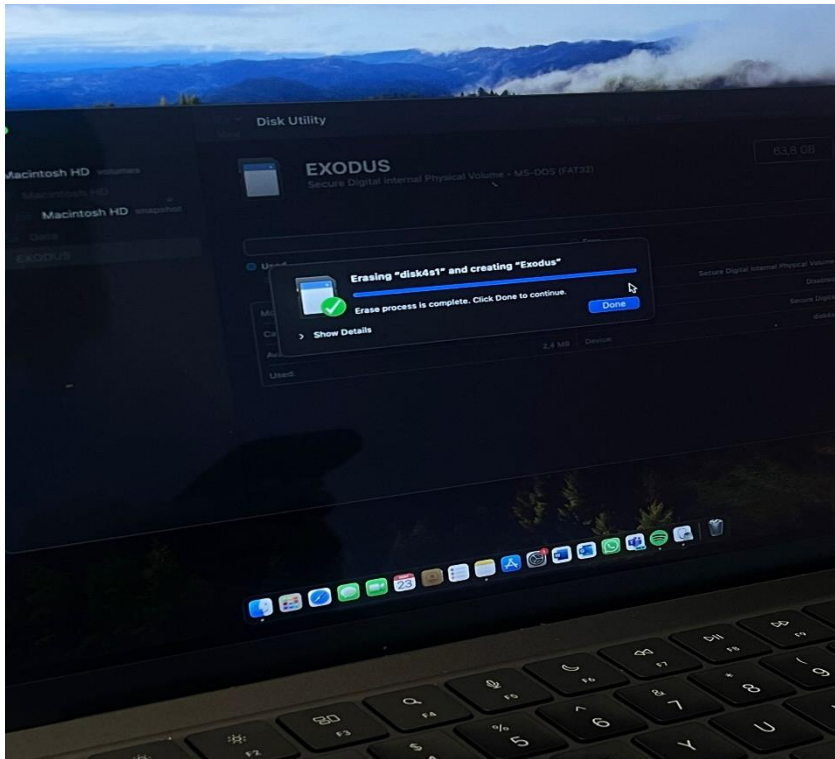
- În unele cazuri, servomotoarele nu au funcționat corespunzător. Acest lucru a fost cauzat fie de conexiuni slabe, fie de alimentarea insuficientă, și a necesitat verificarea și ajustarea conexiunilor și a sursei de alimentare.

3. Conectarea Camerei la Aplicație:

- Configurarea camerei ESP32 pentru a se conecta la aplicația mobilă a fost o altă provocare. A fost necesară verificarea setărilor de rețea și ajustarea parametrilor de conexiune pentru a asigura o comunicare stabilă și eficientă între cameră și aplicație.







Codul Proiectului

❖ Declarațiile Inițiale și Definirea Pinilor

```
#include <Servo.h>

// Definirea pinilor pentru servomotoare
#define VERTICAL_PIN 10
#define HORIZONTAL_PIN 9

// Definirea pinilor pentru senzorul ultrasonic
#define TRIG_PIN 5
#define ECHO_PIN 6

// Definirea pinului pentru buzzer
#define BUZZER_PIN 12

// Definirea pinului pentru LED
#define LED_PIN 13

// Pragul de distanță în cm pentru a declanșa mișcarea servomotoarelor și alarma
#define DISTANCE_THRESHOLD 10

// Declarațiile pentru servomotoare
Servo verticalServo;
Servo horizontalServo;
```

Ce Face:

- Include biblioteca necesară pentru controlul servomotoarelor.
- Defișe pinii pentru controlul servomotoarelor, senzorului ultrasonic, buzzerului și LED-ului.
- Defișe un prag de distanță (în cm) care declanșează mișcarea servomotoarelor și activarea buzzerului/LED-ului.

- Declară obiectele pentru servomotoare.

❖ Funcția pentru Măsurarea Distanței

```
long measureDistance() {  
    // Trimitem un impuls ultrasonic  
    digitalWrite(TRIG_PIN, LOW);  
    delayMicroseconds(2);  
    digitalWrite(TRIG_PIN, HIGH);  
    delayMicroseconds(10);  
    digitalWrite(TRIG_PIN, LOW);  
  
    // Citim timpul de răspuns de la senzor  
    long duration = pulseIn(ECHO_PIN, HIGH);  
  
    // Calculăm distanța în cm  
    long distance = duration * 0.034 / 2;  
    return distance;  
}
```

Ce Face:

- Trimite un impuls ultrasonic prin pinul TRIG al senzorului.
- Măsoară durata de timp până când semnalul ultrasonic revine la senzor (prin pinul ECHO).
- Calculează distanța pe baza duratei măsurate și returnează valoarea distanței în cm.

❖ 3. Funcția setup()

```
void setup() {  
  // Atașarea servomotoarelor la pinii corespunzători  
  verticalServo.attach(VERTICAL_PIN);  
  horizontalServo.attach(HORIZONTAL_PIN);  
  
  // Configurarea pinilor pentru senzorul ultrasonic  
  pinMode(TRIG_PIN, OUTPUT);  
  pinMode(ECHO_PIN, INPUT);  
  
  // Configurarea pinului pentru buzzer  
  pinMode(BUZZER_PIN, OUTPUT);  
  digitalWrite(BUZZER_PIN, LOW);  
  
  // Configurarea pinului pentru LED  
  pinMode(LED_PIN, OUTPUT);  
  digitalWrite(LED_PIN, LOW);  
  
  // Inițializare comunicație serială pentru debugging (opțional)  
  Serial.begin(9600);  
}
```

Ce Face:

- Atașează servomotoarele la pinii specificați.
- Configurează pinii TRIG și ECHO pentru senzorul ultrasonic.
- Configurează pinii pentru buzzer și LED și setează starea inițială a acestora ca LOW (oprit).
- Inițializează comunicația serială pentru debugging (opțional).

❖ 4. Funcția loop()

```
void loop() {  
    // Măsurăm distanța cu senzorul ultrasonic  
    long distance = measureDistance();  
  
    // Verificăm dacă distanța este mai mică decât pragul stabilit  
    if (distance < DISTANCE_THRESHOLD) {  
        // Atașăm servomotoarele (dacă nu sunt deja atașate)  
        if (!verticalServo.attached()) verticalServo.attach(VERTICAL_PIN);  
        if (!horizontalServo.attached()) horizontalServo.attach(HORIZONTAL_PIN);  
  
        // Activăm buzzer-ul și LED-ul  
        digitalWrite(BUZZER_PIN, HIGH);  
        digitalWrite(LED_PIN, HIGH);  
  
        // Mișcare verticală sus maxim și revenire  
        verticalServo.write(180); // Poziția la extrema de sus  
        delay(1000); // Așteptare 1 secundă  
        verticalServo.write(90); // Revenire la poziția de mijloc  
        delay(1000); // Așteptare 1 secundă  
  
        // Mișcare verticală jos maxim și revenire  
        verticalServo.write(0); // Poziția la extrema de jos  
        delay(1000); // Așteptare 1 secundă  
        verticalServo.write(90); // Revenire la poziția de mijloc  
        delay(1000); // Așteptare 1 secundă  
  
        // Mișcare orizontală stânga maxim și revenire  
        horizontalServo.write(0); // Poziția la extrema stângă  
        delay(1000); // Așteptare 1 secundă  
        horizontalServo.write(90); // Revenire la poziția de mijloc  
        delay(1000); // Așteptare 1 secundă  
  
        // Mișcare orizontală dreapta maxim și revenire  
        horizontalServo.write(180); // Poziția la extrema dreaptă  
        delay(1000); // Așteptare 1 secundă  
        horizontalServo.write(90); // Revenire la poziția de mijloc  
        delay(1000); // Așteptare 1 secundă  
    }  
}
```

```

    // Oprirea buzzer-ului și LED-ului după mișcarea servomotoarelor
    digitalWrite(BUZZER_PIN, LOW);
    digitalWrite(LED_PIN, LOW);
} else {
    // Detașăm servomotoarele dacă nu este detectată mișcare
    if (verticalServo.attached()) verticalServo.detach();
    if (horizontalServo.attached()) horizontalServo.detach();

    // Oprim buzzer-ul și LED-ul dacă nu este detectată mișcare
    digitalWrite(BUZZER_PIN, LOW);
    digitalWrite(LED_PIN, LOW);
}

// Debugging (opțional)
Serial.print("Distance: ");
Serial.print(distance);
Serial.println(" cm");

delay(100); // Așteptare între măsurători
}

```

Ce Face:

- Măsoară distanța folosind senzorul ultrasonic.
- Dacă distanța măsurată este mai mică decât pragul stabilit:
 - Atașează servomotoarele dacă nu sunt deja atașate.
 - Activează buzzerul și LED-ul.
 - Mișcă servomotoarele pentru a face o serie de mișcări sus-jos și stânga-dreapta, fiecare urmată de o revenire la poziția de mijloc.
 - Dezactivează buzzerul și LED-ul după ce mișcările sunt complete.
- Dacă distanța măsurată este mai mare decât pragul stabilit:
 - Detașează servomotoarele pentru a economisi energie și a reduce uzura.
 - Dezactivează buzzerul și LED-ul.
- Afișează distanța măsurată pe monitorul serial pentru debugging (opțional).
- Așteaptă 100 ms înainte de a repeta măsurătoarea.

Testarea Soluției pentru Sistemul de Securitate

Pentru a evalua funcționarea și performanța sistemului de securitate bazat pe Arduino, am efectuat o serie de teste riguroase, ținând cont de problemele potențiale cu contactele circuitelor și fiabilitatea componentelor.

Testarea Detecției Mișcării

- **Inițierea Sistemului și Observarea Răspunsului:**
 - Am inițiat sistemul și am monitorizat senzorul de mișcare pentru a verifica dacă detectează corect mișcarea în încăpere.
 - Am simulat diverse scenarii de mișcare pentru a evalua sensibilitatea și precizia senzorului.
 - **Rezultate:** Senzorul a detectat prompt mișcările și a trimis semnale corecte către Arduino.

Testarea Funcționării Buzzerului și LED-ului

- **Activarea Alertei Sonore și Vizuale:**
 - Am verificat funcționarea buzzerului și LED-ului la detectarea mișcării.
 - Am observat dacă buzzerul emite un sunet clar și puternic și dacă LED-ul se aprinde corespunzător.
 - **Rezultate:** Buzzerul și LED-ul s-au activat imediat la detectarea mișcării, oferind alerte sonore și vizuale eficiente.

Testarea Funcționării Servomotoarelor și a Camerei ESP32

- **Verificarea Mișcării și Capturării Imaginilor:**
 - Am testat funcționarea fiecărui servomotor pentru a asigura că mișcă corect camera ESP32.
 - Am observat dacă mișcarea este fluidă și precisă, permițând camerei să captureze imagini din diverse unghiuri.
 - Am verificat transmisia imaginilor în timp real pe telefonul mobil.
 - **Rezultate:** Servomotoarele au funcționat corect, mișcând camera ESP32 pentru a captura imagini clare și complete ale încăperii.

Testarea Conectivității și Sincronizării

- **Verificarea Conexiunii și Sincronizării Camerei:**
 - Am testat conectivitatea camerei ESP32 cu telefonul mobil prin intermediul cardului formatat.
 - Am monitorizat dacă imaginile transmise sunt clare și dacă există întârzieri semnificative în transmisie.
 - **Rezultate:** Conexiunea a fost stabilă, iar imaginile au fost transmise în timp real fără întârzieri semnificative.

Testarea Fiabilității și Stabilității Sistemului

- **Monitorizarea Continuă și Testele de Vibrație:**
 - Am lăsat sistemul să funcționeze continuu pe o perioadă extinsă și am monitorizat dacă apar probleme cu contactele circuitelor sau cu alimentarea.
 - Am efectuat teste de vibrație și șocuri pentru a evalua robustețea sistemului.
 - **Rezultate:** Sistemul a funcționat stabil pe termen lung, iar testele de vibrație și șocuri nu au afectat performanța componentelor.

Înregistrarea Observațiilor și Corectarea Defectelor

- **Identificarea și Remedierea Problemelor:**
 - Am notat orice probleme întâlnite în timpul testelor, cum ar fi eventuale întreruperi ale semnalului sau neconcordanțe în mișcarea servomotoarelor.
 - Am făcut ajustările necesare pentru a corecta defectele identificate, asigurând o funcționare optimă.
 - **Rezultate:** Ajustările au îmbunătățit performanța generală a sistemului, asigurând o operare fiabilă.

Verificarea și Ajustarea Așezării Componentelor

- **Fixarea și Poziționarea Corectă a Componentelor:**
 - Am verificat că toate componentele, inclusiv servomotoarele și camera, sunt bine fixate și poziționate corect pe platformă.
 - Am făcut ajustări pentru a asigura o aliniere perfectă și o fixare sigură a tuturor componentelor.
 - **Rezultate:** Toate componentele au fost aliniate și fixate corect, permițând o funcționare fără probleme a sistemului.

Prin aceste teste, am confirmat funcționalitatea și performanța sistemului de securitate bazat pe Arduino. Identificarea și corectarea problemelor legate de contactele circuitelor și alinierea componentelor au asigurat o funcționare stabilă și fiabilă a dispozitivului în condiții reale de utilizare.