

# Smart Lock System

Smart Lock System

**Autor:** Bogdan Dumitrescu

**Grupa:** 332CA

**Email:** bogdan.dumitrescu02@stud.acs.upb.ro

## Introducere

Proiectul reprezinta sistem de securitate al casei care consta intr-un sistem inteligent de deschidere al unei usi. El poate fi montat pe orice usa pentru acces cu parola si in cazul incercarii de intrare a unei persoane straine, se va declansa o alarma si vor fi trimise informatii despre momentul in care s-a declansat, catre persoana care detine produsul.

## Descriere generală

### OLD->

Proiectul este format pe de-o parte dintr-o placa de dezvoltare Arduino Mega2560, la care este legat un keypad 4x4, un modul led RGB, un micro servomotor(pentru a deschide usa), un buzzer si un ecran LED cu interfata I2C(pentru a afisa diferite informatii).

Pe de alta parte, proiectul mai contine o alta placa de dezvoltare, ESP32-CAM, folosita atat pentru camera OV2640, cat si pentru modulul Wifi, pentru a trimite date la un server, care notifica userul. Aceasta va schimba informatii cu placa de dezvoltare arduino pentru acces de la distanta, a afisa mesaje pe display-ul led, sau pentru a declansa alarma.

### Update ->

*Citeste partea software pentru varianta finala + software*

### Arduino:

- Exista o parola corecta, daca cineva scrie parola corecta, se activeaza un servo sa deschida zavorul, ledul se face verde pentru 5 secunde, dupa care se inchide zavorul(se invarte invers servo).
- Daca se greseste parola, buzzer-ul scoate un sunet scurt si se face rosu ledul 500ms si se afiseaza mesajul Wrong password

- Daca gresesti parola de 3 ori suna o alarma(buzzerul) cu un sunet de sirena de politie, iar ledul rgb flash-uieste rosu si alabstru sincronizat cu alarma.
- -> aceasta se dezactiveaza fie de la server, fie se scrie parola de urgenta.
- -> Exista implementat modul de schimbare parola direct in arduino sau de pe server

Probabil te intrebi de ce exista 3 componente de feedback(actuatori)(buzzer + led + display). Display-ul il voi folosi sa mai afisez mesaje de la server(acum are mai mult sens).  
exemplu: Vad pe cineva la usa prin camera, afisez prin display mesajul: Nu sunt acasa

Acest arduino comunica cu un ESP32-CAM prin UART.

~~Pe esp voi vrea sa adaug niste senzori (deocamdata am unul de proximitate care nu merge din pacate, vreau sa mai am si unul de distanta si unul de sunet). La Esp voi conecta 2 senzori: unul ultrasonic si unul de detectare sunete.~~

De ce atatia senzori? Vreau sa ii folosesc pentru optimizari, practic la un moment dat vreau sa fie activi doar senzorii de pe esp + tastatura de pe arduino, adica sa fie in sleep procesoarele ca sa nu consume multa baterie.

~~Vreau sa se faca o poza si sa se trimita prin gmail cuiva o alarma, poate accesa pozele de pe un server(alt server nu de pe esp explic ulterior de ce).~~

Totodata pe esp mai am si o ruta de live\_video, care returneaza un video efectiv cu ce se intampla in momentul respectiv.(modulul de wifi l-as dezactiva daca senzorii nu returneaza nimic ca nu are sens sa pornesti camera sa nu vezi nimic).

Am un alt server pentru ca am observat ca se incarca greu pozele pe esp, si daca se apeleaza prin alt server(adica daca am un buton de take\_photo si dupa am view\_photo), se vede mai fluid si mi se pare mai buna logica.

Vreau pe serverul ca pe acest server sa am si un feature de activate alarm/deactivate alarm si allow access de la distanta.

**Pentru comunicarea esp-arduino prin UART, folosesc un [level shifter](#) , deoarece pinii gpio opereaza la 3.3V(maxim 3.6V), iar pinii arduino opereaza 5V**

## Hardware Design

Fiecare senzor/actuator are un link catre datasheet-ul acesteia.

**Pinii sunt alesi astfel incat protocolul folosit pe actuator/senzor sa fie valid pentru fiecare. De exemplu, pentru rgb leds am ales pinii 10, 11, 12 care sunt pini PWM. Alti pini au fost alesi si aleator / am folosit pini analogici undeva unde puteau fi folositi pini digitali, dar nu am mai avut pini digitali pe placa, de exemplu, servomotorul este legat la un pin analogic, in loc de un pin PWM, dar reuseste sa isi indeplineasca sarcina si asa**

| Component                  | Arduino Pin(s)           | Protocol/Communication Type |
|----------------------------|--------------------------|-----------------------------|
| Keypad (Rows)              | 6, 7, 8, 9               | Digital Input               |
| Keypad (Columns)           | 2, 3, 4, 5               | Digital Input               |
| RGB LED (Red)              | 10                       | GPIO                        |
| RGB LED (Green)            | 11                       | GPIO                        |
| RGB LED (Blue)             | 12                       | GPIO                        |
| Active Buzzer              | 13                       | GPIO/Digital Output         |
| LCD (I2C Address)          | I2C (A4 - SDA, A5 - SCL) | I2C                         |
| Servo Motor 360 Continuous | A0                       | GPIO                        |
| HC-SR04 Plus Trigger       | A2                       | GPIO                        |
| HC-SR04 Plus Echo          | A3                       | GPIO                        |
| RX                         | 0                        | UART                        |
| TX                         | 1                        | UART                        |

| Component | ESP32-CAM Pin(s) | Protocol/Communication Type |
|-----------|------------------|-----------------------------|
| RX        | 3                | UART                        |
| TX        | 1                | UART                        |

| Component              | ESP32-CAM Pin(s) | Protocol/Communication Type |
|------------------------|------------------|-----------------------------|
| HC-SR04 Plus Trigger   | 13               | GPIO                        |
| HC-SR04 Plus Echo      | 12               | GPIO                        |
| Sound Sensor KY-037 AO | 2                | ADC                         |
| Sound Sensor KY-037 DO | 4                | GPIO                        |

## Hardware images



# Project Design Update

Prin "design" ma refer la rearanjarea/eliminarea senzorilor pe placi 

Am renuntat la senzorul de sunet deoarece nu era optim pentru distante mari, doar la apropiere detecta bine sunetele, oricat incercam sa il reglez din potentiometru, am ales in schimb sa maresc distanta de la senzorul ultrasonic ca sa detecteze mai din timp daca este cineva la usa

## Software Design

**Github Repo:** <https://github.com/dumibxd26/HouseSecuritySystem/tree/main>

### Explicarea codului de pe github

In repository exista 3 directoare:

- **ArduinoLock** - contine codul pentru placa de dezvoltare Arduino UNO
- **ESPServer** - contine codul pentru placa de dezvoltare ESP32-CAM

- **CloudServer** - contine codul pentru serverul .NET care interactioneaza cu serverul tinut pe ESP32-CAM

## Arduino Lock:

In partea de setup, initializam Seriala cu baud rate-ul 9600 pentru ca acesta este maximul recomandat pe arduino, prin intermediul acesteia vom comunica si cu ESP-ul, unde evident vom seta acelasi baud rate. Dupa aceea initializam senzorii, pentru Keypad si LCD am ales sa folosesc clase pentru a implementa mai multe functii, si chiar initializarea. Initializarea LCD-ului consta in pornirea backlight-ului si afisarea pe ecran a mesajului "Enter password" pe prima linie. Initializarea tasturii setarea pinilor de pe linii ca input si a activa rezistentele de pullup, iar coloanele ca output si high. Un semnal LOW pe o linie si o coloana simultan reprezinta apasarea unei taste. Intra un final, initializam pinii pentru fiecare senzor/actuator. Inital, dezactivam sistemul(il setam in modul idle si asteptam semnal de pe senzorul de proximitate pentru activare). In prima parte din loop, se verifica daca este detectata o persoana la distanta de minim 30 metri, daca a fost detectata si placuta este oprita, aceasta porneste(se asteapta semnale de la tastatura sau de la serverul extern). Daca este pornita de mai mult de 120 de secunde(2 min) si nu este in proces de folosire( alarma nu e pornita, o parola nu este scrisa, sau daca nu se reseteaza parola), placuta este dezactivata.

```
void checkDistance()
{
  digitalWrite(trigPin, LOW);
  delayMicroseconds(2);
  digitalWrite(trigPin, HIGH);
  delayMicroseconds(10);
  digitalWrite(trigPin, LOW);

  duration = pulseIn(echoPin, HIGH);
  distance = duration * 0.034 / 2;

  if (distance > 0 && distance
```