

Smart Alarm System

Autor: [Ion Teodora-Andreea, 334CB](#)

Introducere

Scopul proiectului este de a construi un sistem de securitate inteligent, util in detectia, semnalarea si identificarea prezentei nepermise intr-o anumita arie de actiune.

Atunci cand identifica miscare, acesta declanseaza o alarma si lumina rosie intermitenta pentru a semnala un intrus. Sistemul urmareste intrusul si capteaza un numar predefinit de poze pe care le va salva pe calculatorul utilizatorului. Alarma poate fi oprita doar prin introducerea parolei presetate.

Un astfel de sistem de securitate cu alarma este util pentru monitorizarea accesului intr-o zona prestabilita, retinand informatii (poze) despre intrus doar in momentul cand a fost detectata prezenta. Prin ignorarea momentelor de inactivitate in aria de actiune sistemul devine unul eficient, fiind necesar un consum minimal de spatiu de stocare.

Descriere generală



Un senzor detecteaza miscarea, transmite informatia la placuta arduino si, cu ajutorul unui servomotor, senzorul este rotit dupa pozitia unde a detectat activitate. La detectia miscarii placuta Arduino actioneaza un buzzer ce poate fi oprit prin introducerea unei parole ce va fi afisata in timp real pe un ecran LCD.

Servomotorul roteste totodata si o camera web care va face un numar predefinit de poze pe care le va salva intr-un fisier pe PC-ul utilizatorului, doar camera fiind gestionata de un Raspberry Pi.

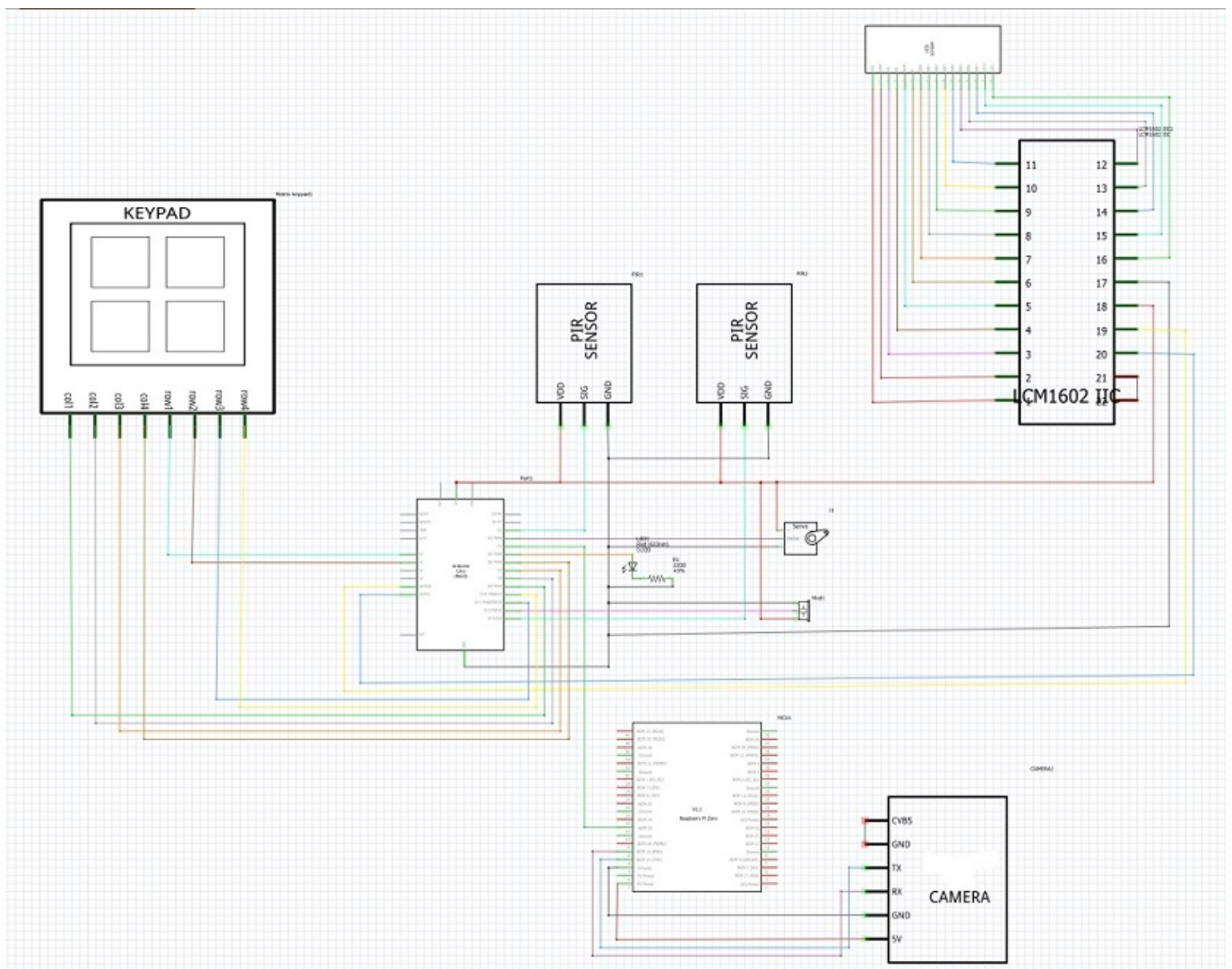
Totodata, de la detectia miscarii si pana cand alarma va fi oprita prin introducerea parolei corecte, un led RGB va lumina rosu intermitent indicand prezenta nepermisa.

Hardware Design

Lista componente

- Arduino Uno
- Raspberry Pi Zero
- Breadboard
- 2 x HC-SR501 PIR Sensor
- Modul buzzer pasiv
- 1602 LCD cu interfata I2C
- SG90 Micro Servo Motor
- Modul Keypad 4x4 (membrana)
- LED
- Camera Web USB
- Rezistente

Schema electrica



Software Design

Parte de software este compusa din 3 fisiere:

1. **smart_alarm_sys.ino**

- cuprinde cod C++ specific placutei Arduino care ii descrie acesteia maniera in care interactioneaza cu componentele si le controleaza

2. **smart_alarm_sys.py**

- fisier ce contine cod python utilizat pentru a realiza functionalitatea placutei Raspberry Pi Zero in cadrul proiectului; pentru punerea in actiune a placutei, se realizeaza o conexiune ssh catre aceasta din spatial local

Aceasta ruleaza scriptul, prin intermediul caruia asteapta input pe pinul 16 conectat la placuta Arduino, dupa primirea semnalului care indica ca a fost detectata miscare, placuta Raspberry Pi actionand camera care va face o poza in directia intrusului.

Dupa ce poza a fost facuta, din interiorul scriptului de python se apeleaza un alt script bash `smart_alarm_sys.sh` care va trimite poza pe laptopul utilizatorului cu ajutorul utilitarului `scp` (poza va fi salvata by default in folderul Pictures din partitia C a utilizatorului curent). Tot in cadrul scriptului bash se face autentificarea accesului in mod automat.

3. **smart_alarm_sys.sh**

- contine codul bash pentru trimiterea imaginii capturate de camera web de pe Raspberry Pi catre calculatorul utilizatorului

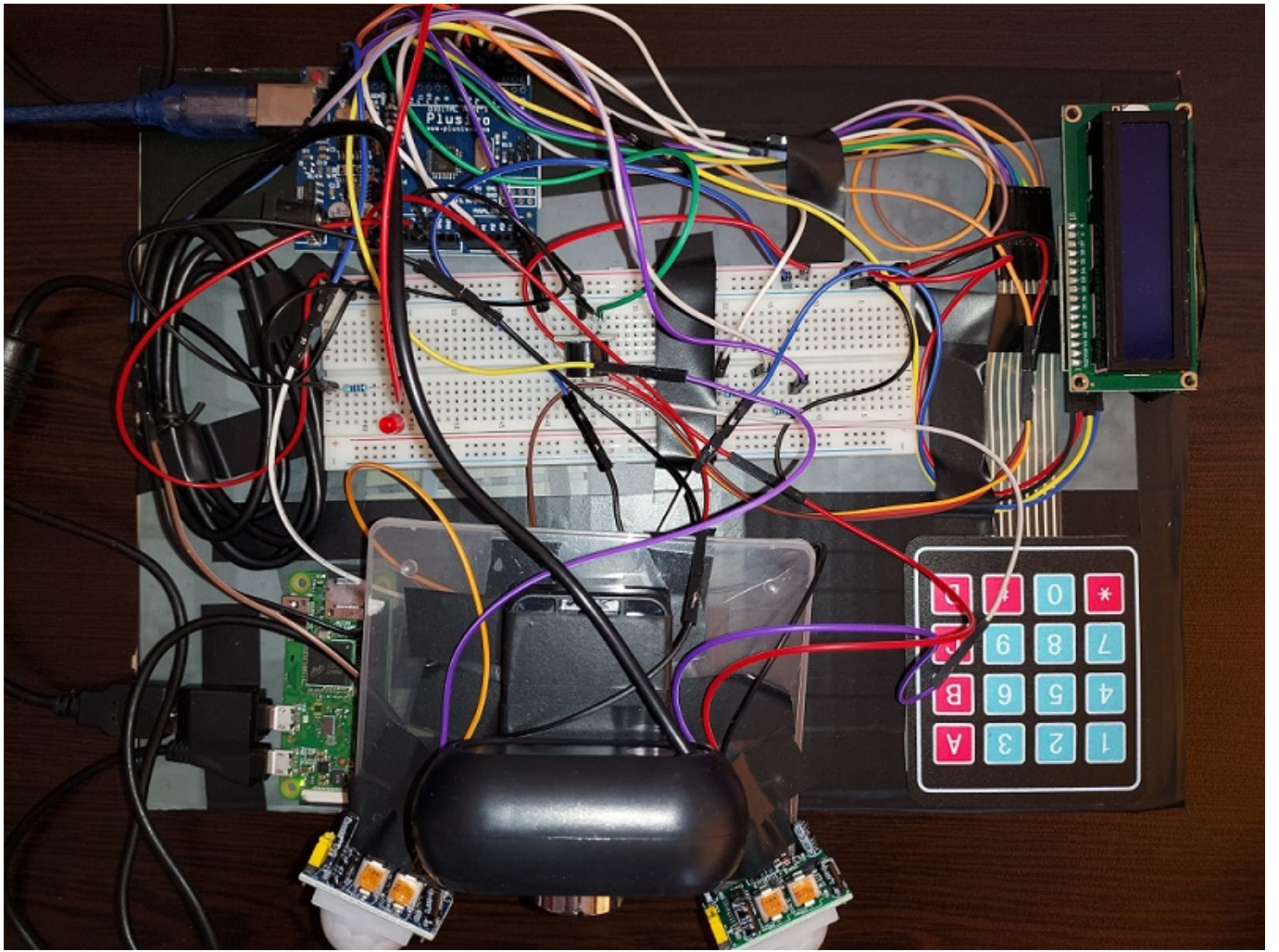
Fisierul `smart_alarm_sys.ino` ce contine cod Arduino se gaseste pe calculatorul utilizatorului.

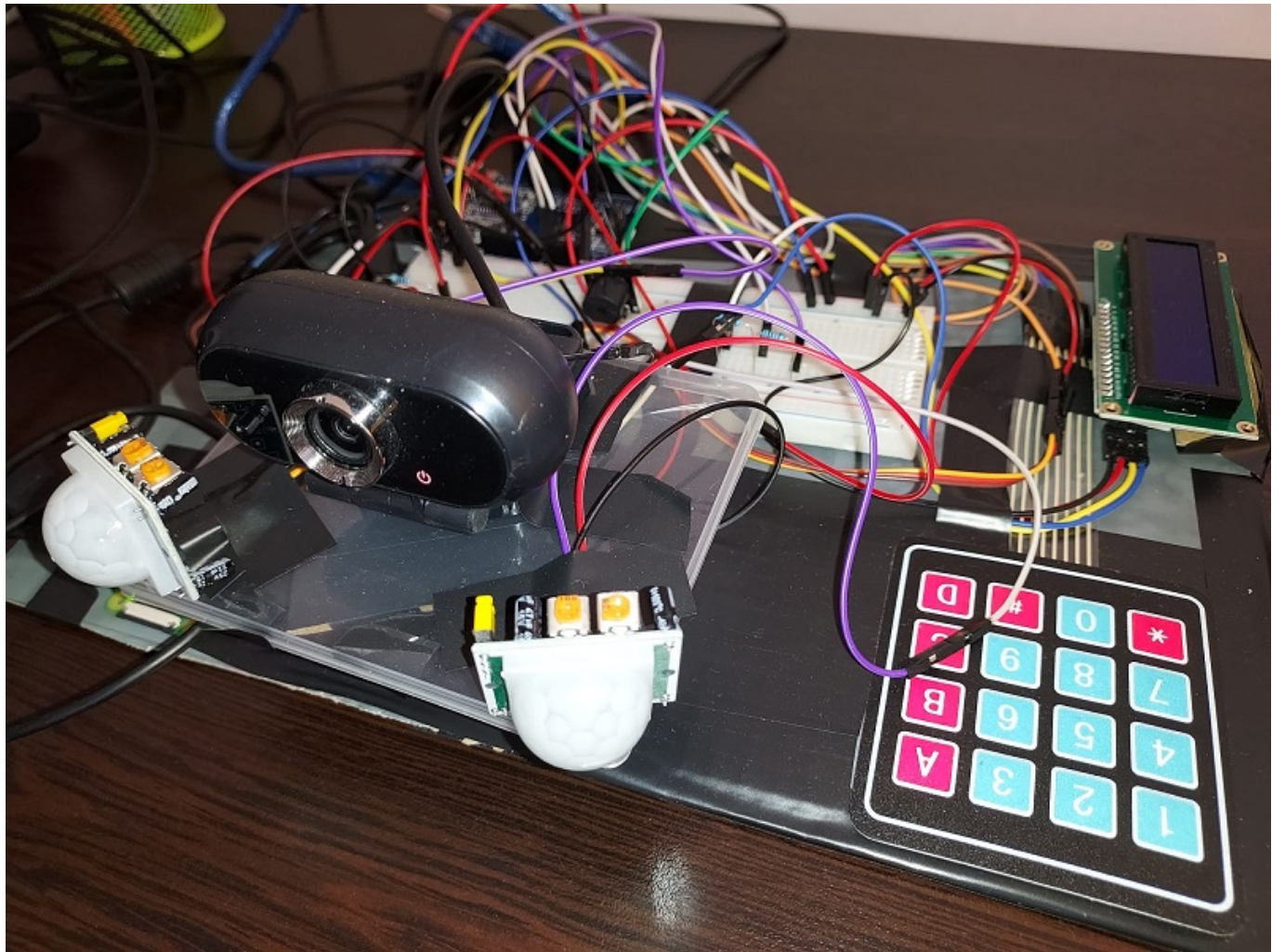
Atat scriptul python, cat si scriptul bash sunt stocate pe placuta Raspberry Pi Zero.

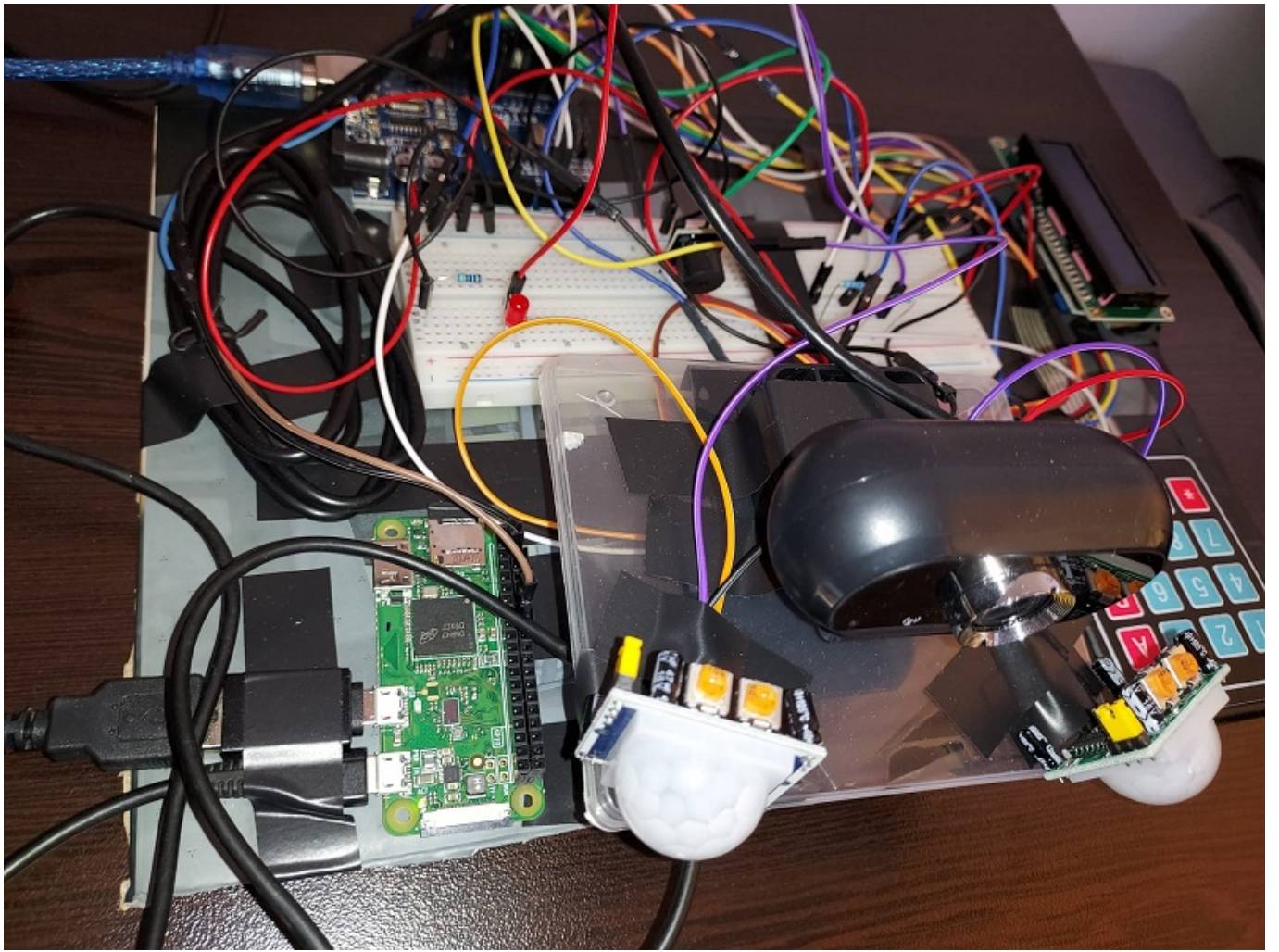
Codul sursa si bibliotecile utilizate pot fi gasite la adresa urmatoare:

https://github.com/theo0131/Smart_alarm_system

Rezultate Obținute









La finalul implementarii proiectului in format fizic, acesta indeplineste toate functiile descrise. Cu ajutorul celor 2 senzori PIR de miscare asezati astfel incat unghiurile lor de actiune sa nu se intrepatrunda decat intr-o portiune bine stabilita, intrusul este detectat, camera este rotita catre directia miscarii receptate, iar buzzerul impreuna cu ledul care lumineaza intermitent activate.

Sistemul format din camera si senzori este montat pe bratul unui micro servo motor care se roteste intr-o directie cu un numar stabilit de grade doar atunci cand unul din senzori detecteaza miscare.

De exemplu, daca doar sensorul din dreapta detecteaza miscare, servo motorul se va roti cu un numar de grade in directia aceea. Daca subiectul intra in aria sensorului stang, sistemul se va roti catre acesta. Daca ambii senzori detecteaza prezenta intrusului, sistemul va ramane nemiscat (se considera ca se afla pe centru, in raza de actiune a camerei).

Cei 2 senzori pot avea uneori erori de detectie, nedetectand constant la fiecare semnal transmis, putand exista o ordinea aleatoare in semnalele date de senzori, dar asta nu va reprezenta o problema caci servo motorul se va roti stanga, respectiv dreapta dupa numarul de semnale transmise, sistemul pastrandu-si pozitia corespunzatoare.

Dupa pornirea alarmei prin intermediul buzzerului, aceasta poate fi oprita prin introducerea parolei folosind keypadul, parola ce va fi afisata in timp real pe un LCD. Daca parola introdusa este corecta se va afisa mesajul "Correct" iar alarma va fi dezactivata, in caz contrar se va afisa "Incorrect" iar alarma va continua sa functioneze.

In momentul in care alarma este pornita, camera este activata si realizeaza o poza stocata in memoria placutei Raspberry Pi, poza pe care o va trimite in folderul Pictures al utilizatorului curent.

Pentru a realiza acesta functionalitate, scriptul python trebuie sa fie rulat pe placuta Raspberry, iar pc-ul utilizatorului sa fie conectat la aceeaasi retea ca placuta.

Un demo al proiectului se gaseste la adresa: <https://youtu.be/bkMKPZI5QT0>

Concluzii

Proiectul implementeaza la nivel de prototip un sistem de securitate cu alarma care ar putea fi foarte util si in practica prin intermediul facilitatilor pe care le pune la dispozitie.

Cu toate acestea, precizia actiunilor realizate si sincronizarea lor este descrisa de caracteristici destul de slabe din cauza calitatii si posibilitatilor reduce ale pieselor utilizate.

Din punct de vedere al functionalitatii curente a proiectului si datorita modului de abordare structurat al problematicii, acesta poate fi usor extins si modificat pentru realizarea unor sisteme mult mai complexe si eficiente.

Download

[smart_alarm_system.zip](#)

Jurnal

- realizat in Tinkercad partea principala a proiectului ce consta din partea de pornire buzzer la detectia miscarii si oprirea acestuia prin parola
- adaugarea servo motorului si miscarea acestuia in functie de miscarea detectata de cei 2 senzori PIR
- crearea partii de Arduino in format fizic
- adaugarea placutei Raspberry Pi impreuna cu camera web; scrierea scripturilor pentru realizarea si trimiterea catre utilizator a imaginii capturate de camera web
- montarea bratului de miscare a camerei impreuna cu senzorii PIR de catre servo motor in functie de miscarea detectata de senzori
- fixarea componentelor pe o bucata de carton rezistent pentru stabilizarea circuitului

Bibliografie/Resurse

Pentru simularea proiectului s-a folosit platforma [Tinkercad](#).

Pentru crearea schematicului a fost folosita aplicatia [Fritzing](#).

Codul Arduino a fost scris si testat cu ajutorul Arduino IDE.

<https://components101.com/sensors/hc-sr501-pir-sensor>

http://www.ee.ic.ac.uk/pcheung/teaching/DE1_EE/stores/sg90_datasheet.pdf

<https://circuitdigest.com/article/16x2-lcd-display-module-pinout-datasheet>

<https://arduinomodules.info/ky-006-passive-buzzer-module/>

<https://components101.com/misc/4x4-keypad-module-pinout-configuration-features-datasheet>

<https://raspberrypi-guide.github.io/electronics/using-usb-webcams>

Export to PDF

From:

<http://ocw.cs.pub.ro/courses/> - **CS Open CourseWare**

Permanent link:

http://ocw.cs.pub.ro/courses/pm/prj2022/sgherman/smart_alarm_system



Last update: **2022/05/27 21:54**