



Titlu = Sistem de supraveghere

**Echipa: Petrica Petru(1306B)(petrica.petru99@gmail.com), Dorcu Andrei(1306B)
(andy.dorcu@yahoo.com), Bucur Raul-Vasile (bucurraul123@yahoo.com)**

Sistem de supraveghere

Elevator Pitch:

Prin acest proiect incercam sa rezolvam una din problemele care apar frecvent, si anume securitatea. Din cauza atacurilor ce pot aparea in zilele noastre am dorit sa implementam un sistem de supraveghere care sa proceseze imaginile pentru a detecta miscare.

Componente:

Hardware:

- 1 x Raspberry Pi
- 2 x Servo(SG-90)
- 2 x Webcam
- 2 x LED for lighting
- 2 x Resistor(current limiting for led)

Software:

- Python 3
- Flask
- OpenCV
- GPIO library

Cos Sursa : https://github.com/bucurraul/SM_Proiect

Video:https://drive.google.com/file/d/1Wg-Vw7jqcVjD2AV89CHx-BAmwxsDvudE/view?fbclid=IwAR1VKmOFCeyHJA5OYqe8_CGc3j_-z63hukrvGKQun01jaUBP0g3659Nfh9Y

Story:

Am creat un Web Service care accepta conexiunea mai multor clienti si transmite live evenimentele filmate de pe 2 camere, care pot fi miscate de catre oricare client cu un numar de grade si directia dorita. Totodata clientul poate alege daca sa aprinda un led aflat deasupra camerelor pentru mai multa luminozitate.

Camerele proceseaza imaginile primite pentru detectia de miscare.

Conectarea hardware:

Pentru inceput am luat un webcam in vederea testarii functionalitatii acestuia. Dupa ce ne-am asigurat ca functioneaza dupa conectarea acesteia pe USB, si anume ca putem citi din /dev/video0 si putem produce o imagine valida cu un utilitar de genul fswecam am trecut la pasul urmator si anume, montarea acesteia pe un servo. Pe servo am atasat camera, fixand-o bine si fiind siguri ca aceasta nu va iesi de pe el, fiind prinsa cu un surub. Servo-ul a fost fixat bine de o placa de lemn avand in vizor o pozitionare ulterioara a acestuia intr-un unghi din care camera va putea acoperi o suprafata cat mare pentru detectia miscarii.

Langa ea am atasat un led colorat in albastru sau verde pe care l-am pozitionat strategic sa ilumineze in fata ei. Am ales aceasta culoare deoarece este deranjanta pentru un intrus care se va uita la camera, dar ii va permite acesteia sa surprinda detaliile necesare in vederea identificarii suspectului. Pentru alimentarea acestuia am folosit o sursa ATX de calculator care este capabila sa furnizeze un curent suficient la o tensiune de 5 volti astfel incat sa nu existe nici o problema cu alimentarea, nici atunci cand vor fi mai multe camere. Pinul de comanda al servo-ului este conectat la un GPIO de pe raspberry pi si acesta va fi comandat din python utilizand biblioteca RPi.GPIO. Vazand ca functioneaza totul bine, am mai adaugat, exact in aceeasi maniera inca o camera. Pentru cea de-a doua camera am repetat procesul, si anume am fixat-o bine pe servo si am conectat-o la USB-ul placutei de dezvoltare Raspberry Pi. Servo-ul acesteia va fi alimentat din aceeasi sursa si pinul de comanda va fi legat la alt pin de GPIO.

Pe langa aceste camere vom adauga cate un actuator care sa produca lumina in eventualitatea in care utilizatorul doreste sa observe mai bine ce se petrece. Pentru aceasta am folosit 2 led-uri, unul albastru si unul verde si pe acestea, dupa ce le-am pozitionat langa camera le-am conectat anodul la plus-ul de la sursa printr-o rezistenta de limitare a curentului pentru a nu arde led-ul si catodul ledului va fi pe GPIO. Cand pinul de pe placuta va fi pe LOW acesta se va aprinde. Am ales aceasta varianta deoarece este superioara celei cand ledul este activ pe HIGH.

Pasii de instalare:

-> Necesita Python3

-> Daca pachetele exista in repository-ul al Raspbian-ului ele pot fi instalate utilizand "apt-get".

-> Intai aveti nevoie sa actualizati pachetele disponibile utilizand comanda

"sudo apt-get update"

-> Dupa care este nevoie de pachetul RPI.GPIO, care poate fi instalat utilizand

"sudo apt-get install rpi.gpio"

-> Urmatoarele pachete sunt module Python3 independente de Module Raspbian:Flask,OpenCV

-> Pentru a instala cele 2 module este necesar Modulul "PIP" care vine la pachet cu orice versiune de Python

->Comenzile sunt urmatoarele:

"pip install flask" sau "python3 -m pip install flask" (daca sunt pe masina mai multe module python)

"pip install opencv-python" sau "python3 -m pip install opencv-python" (daca sunt pe masina mai multe module python)

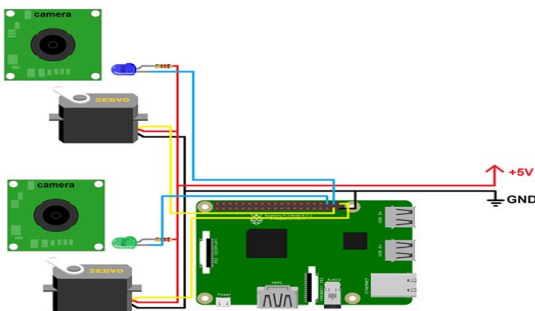
Rularea:

Pentru a rula script-ul se va scrie in linia de comanda:

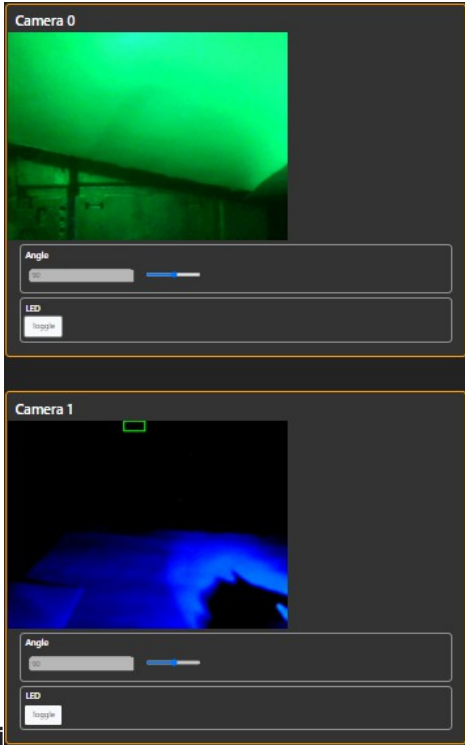
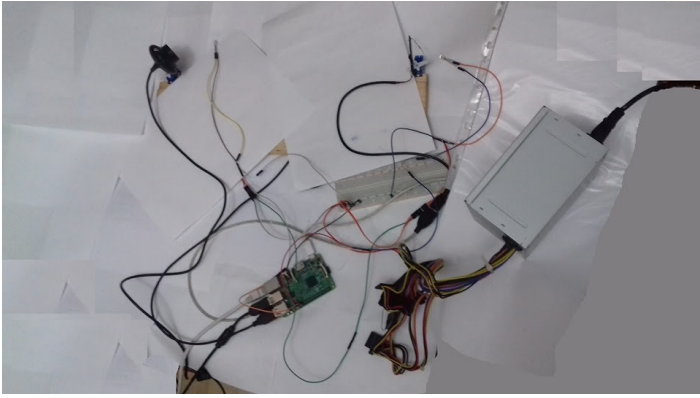
-> Daca sunt toate piese montate cum a fost explicat atunci pe rețeaua locală va rula un server pe portul "6969" (default introdus).

-> Nu mai rămâne decât să accesați pagina pe orice device conectat la rețea printr-un browser astfel: http://<IP_RASPBERRY>:6969/ pentru a putea vedea frame-urile prinse de camera și mișcarea pe care o detectează.

Schema proiectului:



Circuit fizic:



Pagina WEB: