

Uputstvo za Samostalni praktični projekat

(OSS Elektronika i telekomunikacije)

Uvod

U ovom dokumentu nalaze se sve potrebne informacije za korišćenje Raspberry Pi uređaja i povezivanje dodatnih komponenti (senzora, aktuatora i kamere) na sam uređaj. Za više informacija o svakom koraku, savetujemo da pogledate dodatnu literaturu na kraju dokumenta. Ovaj dokument je podeljen u pet celine. U prvoj celini nalazi se kratak pregled programiranja Raspberry Pi uređaja u Python programskom jeziku. Nakon toga nalaze se opisi studentskih projekata. Student može sam predložiti novi projekat koji zadovoljava sledeće tri specifikacije:

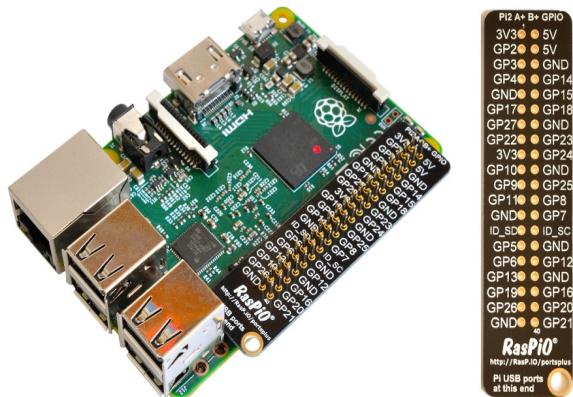
- Korišćenje Raspberry Pi uređaja i Python programskog jezika.
- Povezivanje i interakcija sa dodatnim uređajima (senzori/aktuatori/kamera).
- Bežična komunikacija pomoću integrisanog WiFi-a ili Bluetooth-a.

Novi projekat odobrava predmetni nastavnik, a sam projekat se radi samostalno uz konsultaciju sa predmetnim nastavnikom.

Osnove Raspberry Pi programiranja

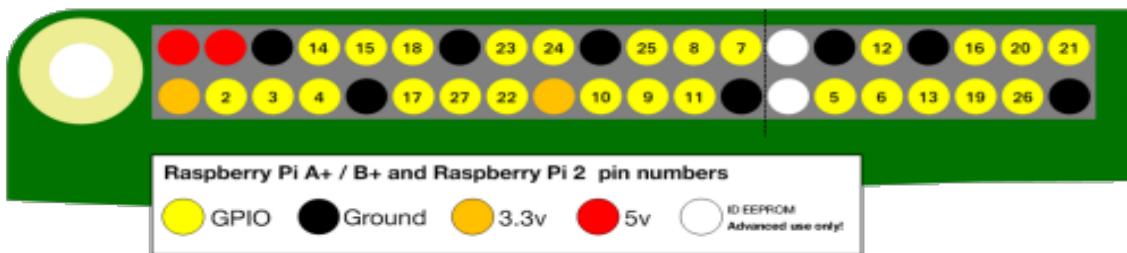
Većina materijala i slika u ovom delu preuzeta je sa sajta: <https://projects.raspberrypi.org>. Na ovom sajtu možete naći mnoge druge zanimljive projekte i korisne informacije.

Raspberry-Pi (u nastavku teksta rPi) je mali integrисани računar niske cene koji se korisit za hobi projekte iz oblasti računarstva i elektronike. Jedna od najmoćnijih osobina rPi uređaja jeste red integrisanih pinova (Sl. 1) opšte namene (eng. General Purpose Input/Output - GPIO) koji predstavljaju interfejs za interakciju sa stvarnim svetom.



Slika 1 - rPi GPIO pinovi [1]

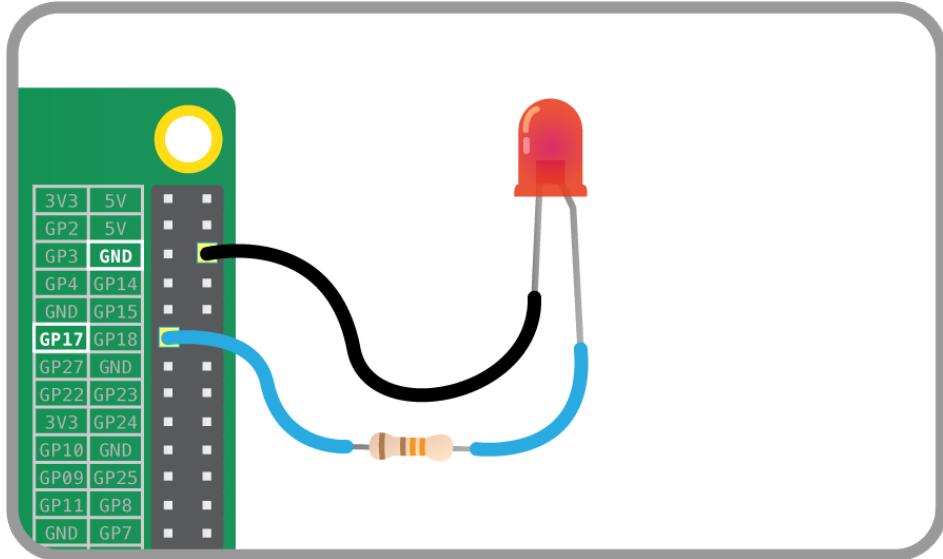
Detaljan raspored rPi pinova prikazan je na Sl. 2. Kao što se može videti, rPi ima pinove za napajanje od 5V i 3.3V, GND pinove koji predstavljaju uzemljenje i opšte pinove koji služe za povezivanje senzora i aktuatora. Može doći do promene rasporeda pinova na različitim serijama rPi ploča, tako da pre početka izrade projekta obavezno proverite seriju vaše ploče i raspored pinova za nju.



Slika 2 - Detaljan pregled pinova na rPi ploči [1]

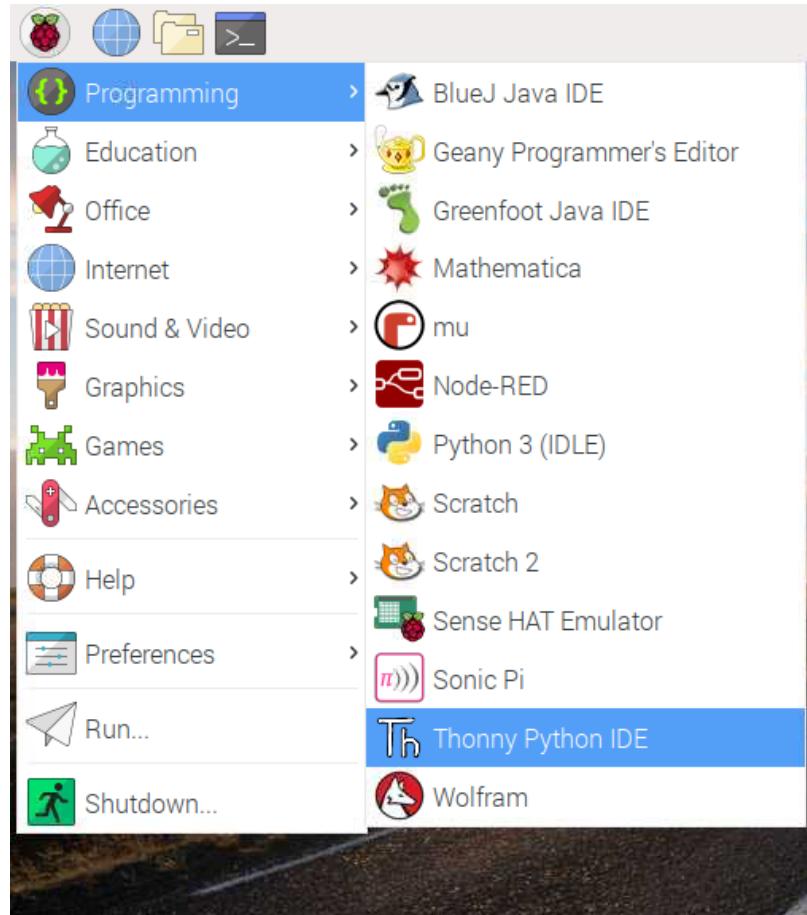
Povezivanje LED diode i zujalice na rPi

Da bi kontrolisali LED diodu, potrebno je povezati dužu nogicu na neki GPIO pin (npr. pin 17), a kraću nogicu na uzemljenje (sva uzemljenja su jednaka), kao što je prikazano na Sl. 3. Otpornik od $500\ \Omega$, služi za zaštitu GPIO pinova.



Slika 3 - Povezivanje LED diode na pin 17 [1]

Za interakciju sa rPi pinovima i povezanim senzorima i aktuatorima koristićemo python biblioteku `gpiozero`. Za uključivanje i isključivanje LED diode imamo na raspolaganju funkcije `LED.on()` i `LED.off()`, dok za pauziranje programa možemo koristiti funkciju `sleep(t_in_s)` iz modula `time`, koja zaustavlja izvršavanje python programa $t_{in\ s}$ sekundi. rPi ima instalirani Python editor - Thonny Python IDE. Da biste pokrenuli Thonny editor kliknite na *Start* (rPi ikonica) i iz *Programming* menija odaberite *Thonny Python IDE* (Sl. 4).



Slika 4 - Pokretanje Thonny editora na rPy uređaju [2]

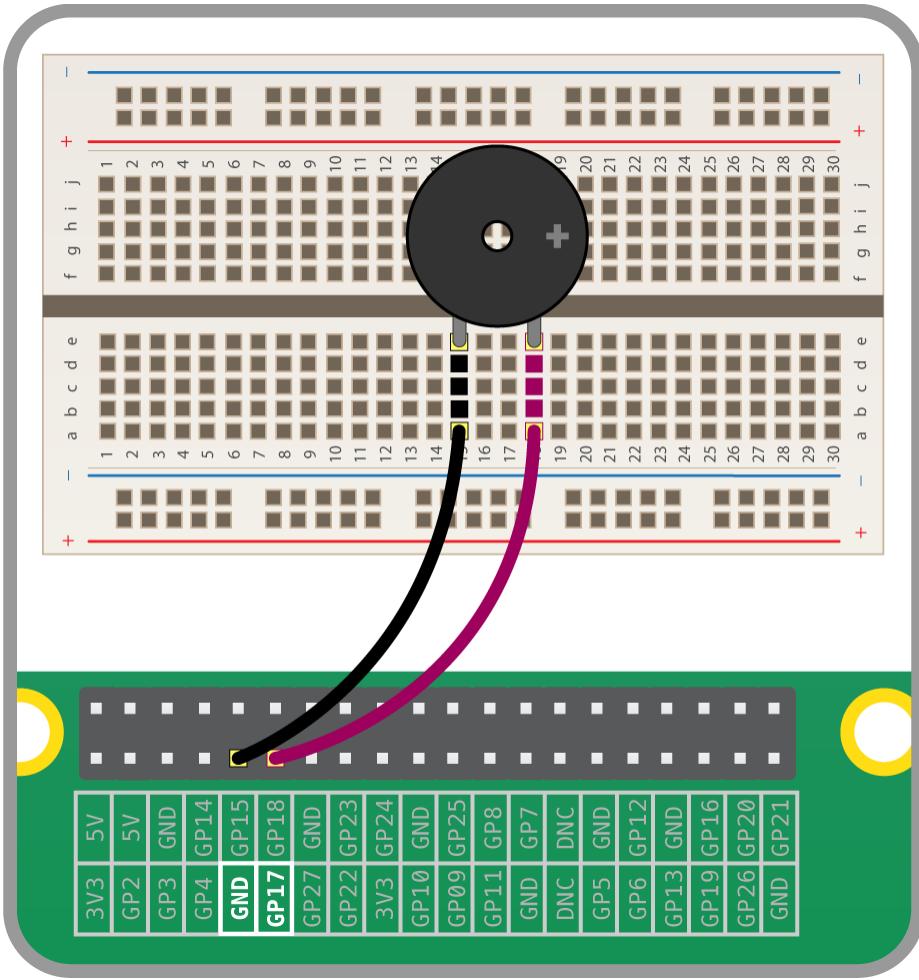
Kod za uključivanje i isključivanje LED diode na pinu 17, sa periodom od 1s:

```
from gpiozero import LED
from time import sleep

led = LED(17)

while True:
    led.on()
    sleep(1)
    led.off()
    sleep(1)
```

Zujalica radi isto kao LED dioda, samo što umesto sveta proizvodi zvuk. Zujalicu je potrebno povezati isto kao i LED diodu (Sl. 5), a kod za kontrolisanje zujalice je dat u nastavku.



Slika 5 - Povezivanje zujalice na GPIO 17 [1]

```
from gpiozero import Buzzer
from time import sleep

buzzer = Buzzer(17)

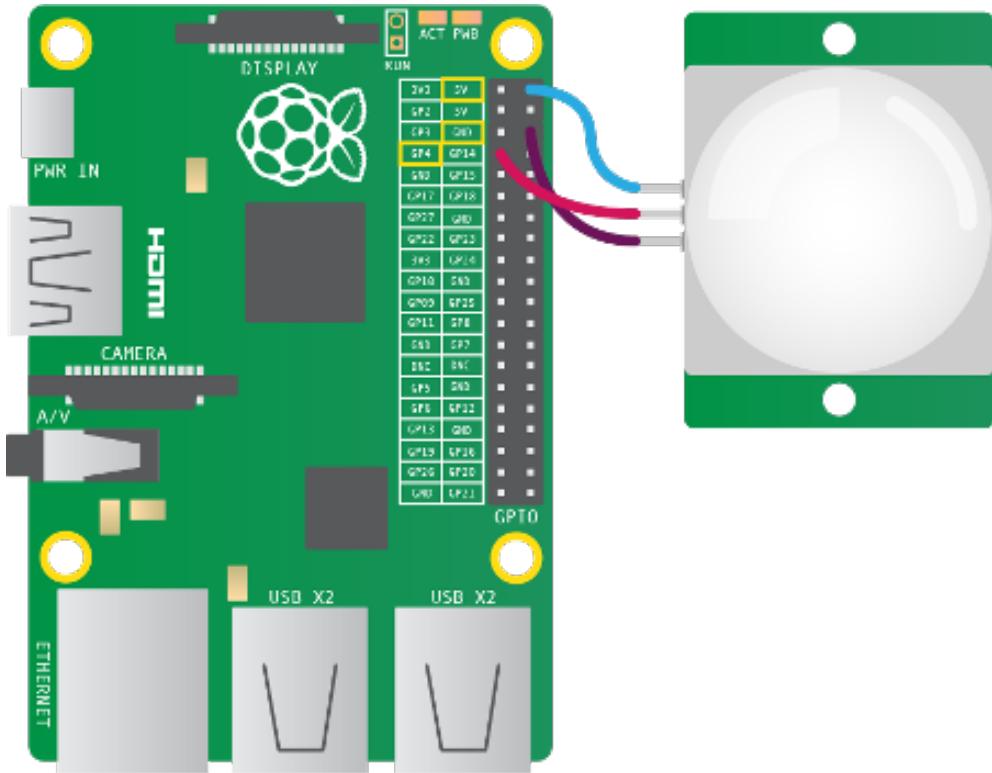
while True:
    buzzer.on()
    sleep(1)
    buzzer.off()
    sleep(1)
```

Zujalica ima posebnu metodu za zujanje `beep()`, tako da je moguće zameniti telo `while` petlje samo sa jednim redom:

```
while True:
    buzzer.beep()
```

Povezivanje senzora na rPi

U ovom primeru ćemo videti kako povezati PIR senzor, koji služi za detekciju pokreta, na rPi uređaj. PIR senzor ima tri terminala, jedan terminal je potrebno povezati na napajanje (5V), jedan terminal na GND, a jedan terminal na GPIO (npr. pin 4). Na donjoj strani senzora je data legenda koji terminal kom pinu odgovara. Povezivanje PIR senzora je prikazano na Sl. 6.



Slika 6 - Povezivanje PIR senzora na rPi uređaj [1]

Za detekciju pokreta imamo funkcije `MotionSensor.wait_for_motion()` i `MotionSensor.wait_for_no_motion()`, koje detektuju kad počinje i kad se završava kretanje, respektivno. Kad se detekuje pokret, ispisaćemo poruku na standardni izlaz pomoću python funkcije `print("")`.

```
from gpiozero import MotionSensor

pir = MotionSensor(4)

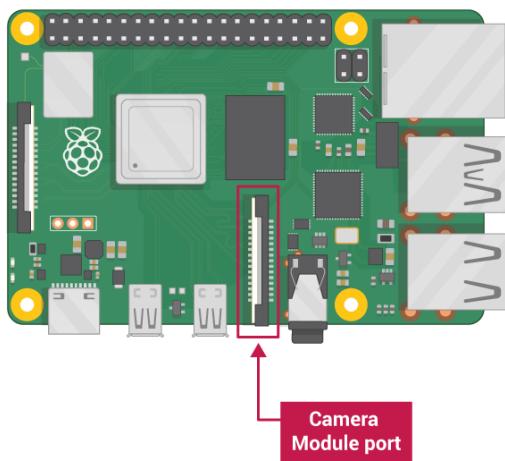
while True:
    pir.wait_for_motion()
    print("You moved")
    pir.wait_for_no_motion()
```

Za povezivanje različitih senzora i aktuatora, pogledajte dokumentaciju `gpiozero` biblioteke [3].

Ova biblioteka postoji već više godina i ima odličnu podršku za sve popularne senzore i aktuatora.

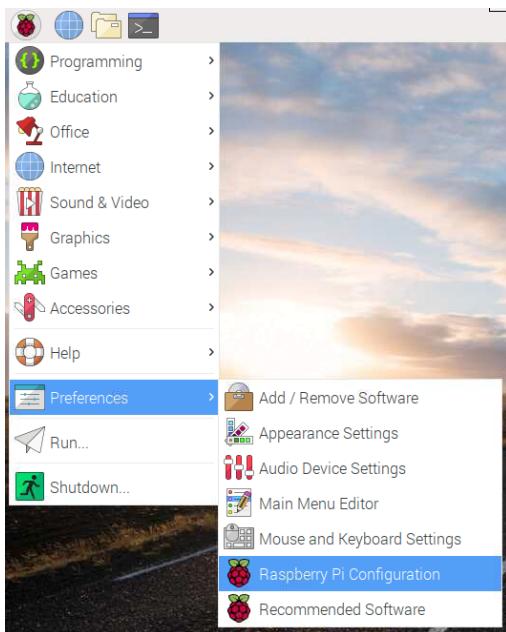
Povezivanje kamere na rPi

Moguće je koristiti ili standardnu USB kameru ili posebnu rPi kameru. rPi kamera ima bolju programsku integraciju (postoje posebne biblioteke za komunikaciju sa rPi kamera modulom), dok je za USB kameru potrebno koristiti neku drugu biblioteku, kao što je OpenCV. rPi kamera se povezuje na poseban port na samoj ploči (Sl. 7.).

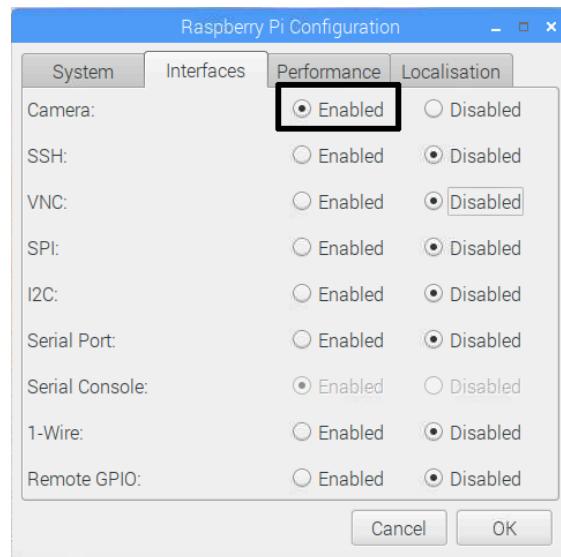


Slika 7 - Port za povezivanje rPi kamere [2]

Pre programiranja kamere, potrebno je proveriti da li je omogućen interfejs sa kamerom. Ovo se radi klikom na *Start->Preferences* i izborom opcije *Raspberry Pi Configurations* (Sl. 8. i Sl. 9).



Slika 8 - Pokretanje Raspberry Pi Configuration prozora [2]



Slika 9 - Omogućivanje interfejsa sa rPi kamerom [2]

Za kontrolu rPi kamere koristićemo posebnu biblioteku picamera. Za prikaz videa možemo koristiti funkcije `PiCamera.start_preview()` i `PiCamera.stop_preview()`.

```
from picamera import PiCamera
```

```
from time import sleep  
  
camera = PiCamera()  
  
camera.start_preview()  
sleep(5)  
camera.stop_preview()
```

Za snimanje videa možemo koristiti funkcije `PiCamera.start_recording()` i

```
PiCamera.stop_recording().  
  
from picamera import PiCamera  
from time import sleep  
  
camera = PiCamera()  
  
camera.start_preview()  
camera.start_recording('/home/pi/Desktop/video.h264')  
sleep(5)  
camera.stop_recording()  
camera.stop_preview()
```

Za više informacija o `picamera` biblioteci pogledajte [4]. Za primer interakcije sa USB kamerom pogledajte [5].

Pravljenje i pokretanje web sajta na Raspberry Pi uređaju

Za povezivanje rPi uređaja na internet pomoću ugrađenog WiFi modula, kao i više informacija o naprednim WiFi podešavanjima možete naći na stranici [6]. Da bi ste pronašli IP adresu vašeg rPi uređaja ukucajte komandu `ifconfig` u terminalu, i pod sekcijom `wlan0` možete videti vašu IP adresu. Za pravljenje i pokretanje web sajtova koristićemo Python biblioteku `Flask`. `Flask` dolazi sa ugrađenim web serverom koji ćemo koristiti za pokretanje našeg sajta. Za instaliranje `flaska` potrebno je pokrenuti komandu `pip install Flask`.

Napravite novi python fajl (npr. `hello.py`) i ukucajte sledeći kod:

```
from flask import Flask  
  
app = Flask(__name__)  
  
@app.route("/")  
def hello_world():  
    return "<p>Hello, World!</p>"  
  
if __name__ == '__main__':  
    app.run(debug=True, host='0.0.0.0', port=80)
```

Da biste pokrenuli program potrebno je u terminalu ukucati:

```
python3 hello.py
```

Funkcija `hello_world` vraća sadržaj html stranice, dok dekorator `@app.route("/")` definiše putanju koju kucamo u browseru. Ukoliko na rPi uređaju u browseru ukucamo adresu <http://127.0.0.1> otvorice nam se naš web sajt. Ovom sajtu možemo pristupiti i iz lokalne mreže odlaskom na IP adresu koju smo pronašli pomoću `ifconfig` komande. Za više informacija o pravljenju web sajta i pokretanju istog na rPi uređaju, pogledajte [7]. Za Flask napredne funkcionalnosti, kao što je dodavanje CSS-a, administratorskog panela i mnoge druge opcije pogledajte [8].

Opis projekta 1.

Napisati program za rukovanje alarmom u poznatom tržnom centru "Mega/Giga" i pokrenuti ga na rPi ploči. Pored rPi uređaja, potrebno je koristiti i PIR senzor, svetlosni senzor i 3 LED diode.

Odmah po pokretanju programa, radi se inicijalizacija sistema (da bi se proverila ispravnost signalizacionih komponenti), pa se Zelena dioda upali i ugasi 5 puta, Žuta dioda se upali i ugasi 4 puta, a Crvena dioda se upali i ugasi 3 puta. Sve diode se pale i gase sa periodom od 1s i ovo se radi samo jednom (samo pri pokretanju programa).

Alarm je aktivan samo ako se pritisne dugme. Ukoliko se detektuju neke promene na svetlosnom i PIR senzoru (senzoru pokreta) dok dugme nije pritisnuto, program ništa ne radi.

Dok god je alarm aktivan program čita vrednosti sa svetlosnog i PIR senzora. Smatramo da je svetlosni senzor aktivan ukoliko je noć, tj. nivo svetlosti je ispod 420. Ako je senzor pokreta aktivan, ali svetlosni senzor nije aktivan, tada se pali samo zelena dioda i šalje se poruka "Opomena" na standardni izlaz (jer se smatra da je osoba zadužena za obezbeđenje izašla na nedozvoljenu pauzu). Kad se svetlosni senzor aktivira (nivo svetla padne ispod 420), pali se žuta dioda kao znak da je alarm aktiviran. Ukoliko se senzor pokreta aktivira dok je svetlosni senzor aktivan, pali se crvena dioda i šalje se poruka "UZBUNA!" na standardni izlaz. Napraviti web sajt koji prikazuje stanje sistema.

Opis projekta 2.

Napraviti radar za detekciju neželjenih prepreka pomoću rPi uređaja, ultrazvučnog senzora i servo motora. Servo motor se konstantno pomera u polukrugu, dok ultrazvuci senzor detektuje prisustvo neželjenih prepreka u datom pravcu. Podići web sajt koji prikazuje očitavanje radara.

Opis projekta 3.

Napraviti sistem za proveru identita prilikom pristupa službenim prostorijama u poznatom tržnom centru "Mega/Giga". Sistem implementirati tako da rPi uređaj predstavlja integrisani deo sigurnosne brave za pristup službenim prostorijama, i prepoznavanjem određenog zaposlenog lica šalje signal za otključavanje brave. Otključavanje brave u ovom primeru ilustrovati pomoću dve LED diode, gde će crvena boja označavati da je brava zaključana, a zelena da je brava otključana. Snimanje službenog lica će se vršiti pomoću rPi kamere, a sistem treba da se aktivira nakon što službeno lice pritisne dugme koje pokreće sistem preoznavanja i

otključavanja brave.

Opis projekta 4.

Napraviti sistem za proveru prisutnosti zaštitne maske na licu u cilju sprečavanja Covid-19 infekcije u poznatom tržnom centru "Mega/Giga". Sistem bi bio postavljen na svakom ulazu u objekat, gde bi, uz pomoć rPi uređaja i rPi kamere vršio proveru da li osoba nosi masku. Nakon donošenja zaključka o prisutnosti maske, sistem bi upravljao trokrakom barijerom, gde bi u slučaju prisutnosti zaštitne maske signalizirao dopuštanje jednog prolaza kroz barijeru. U ovom primeru će to biti simulirano uz pomoć dve LED diode, gde dozvola, odnosno, zabrana prolaza, će biti simulirana zelenom i crvenom LED diodom, respektivno.

Opis projekta 5.

Napraviti sistem za bezbednosni nadzor objekta, koji se sastoji od rPi uređaja (sa integrisanim WiFi modulom), rPi kamere i PIR senzora za detekciju pokreta. Napisati program u programskom jeziku Python koji kad detektuje pokret kreće da snima video sa kamere i pomoću WiFi komunikacije streamuje isti na server. Ukoliko nema pokreta (PIR senzor je neaktivovan), rPi ne snima i ne šalje video, već je neaktivovan.

Opis projekta 6.

Napraviti sistem za praćenje temperature u server sali poznatog tržnog centra "Mega /Giga", koji osim merenja temperature, omogućava i upis trenutnih vrednosti temperature u .csv fajl, kao i iscrtavanje datih vrednosti na grafiku (pomoću biblioteke matplotlib). Da bi serveri normalno funkcionsali, temperatura u sali ne sme da pređe 20°C, pa je neophodno implementirati alarm koji će zaposlene upozoriti da je temperatura prešla maksimalnu dozvoljenu vrednost, kako bi se problem otklonio na vreme. Sistem se sastoji od rPi uređaja, senzora za merenje temperature i crvene LED diode, koja predstavlja alarm i pali se samo kada trenutno očitana vrednost temperature pređe 20°C, dok je u ostalim slučajevima neaktivna, tj. ugašena. Kada dođe do paljenja crvene LED diode, tj. do alarma, neophodno je poslati poruku "OPOMENA" na standardni izlaz ako je temperatura između 21°C i 25°C, odnosno poruku "UZBUNA" ako je temperatura preko 25°C.

Opis projekta 7.

Pomoću rPi uređaja napraviti personalizovanog social network bota, koji simulira Smart Home koncept. Socijalnu mrežu student bira sam, a u obzir dolaze Twitter (uz pomoć Twython paketa) ili WhatsApp.

Ako se odabere Twitter, cilj projekta je da se napravi sistem koji automatski objavljuje nove postove na odgovarajući Twitter nalog (i na taj način simulira detekciju odgovarajućih pojava u

Smart Home okruženjima).

Sa druge strane, ako se student odluči za WhatsApp, cilj projekta je da se napravi komunikacija sa WhatsApp bot-om, u okviru koje će biti omogućeno paljenje i gašenje LED diode pomoću WhatsApp poruke upućene od strane korisnika ka bot-u, i na taj način će se simulirati paljenje i gašenje svetla u Smart Home okruženju. Bot bi se u WhatsApp slučaju aktivirao slanjem odgovarajuće poruke, i imao bi predefinisane odgovore, u zavisnosti od komande koja mu se zadaje. Osim rPi uređaja, u ovom projektu bi se koristila dodatna LED dioda, koja bi simulirala svetlo.

Reference

1. Raspberry Pi projekat - "Physical Computing with Python" (<https://projects.raspberrypi.org/en/projects/physical-computing>)
2. Raspberry Pi projekat - "Getting started with the Camera Module" (<https://projects.raspberrypi.org/en/projects/getting-started-with-picamera>)
3. Dokumentacija za biblioteku `gpiozero` - "GPIO Zero" (<https://gpiozero.readthedocs.io/en/stable/>)
4. Dokumentacija za biblioteku `picamera` - "picamera library" (<https://picamera.readthedocs.io/en/release-1.13/>)
5. Korišćenje USB kamere sa rPi uređajem - "Raspberry Pi Tutorial on using a USB Camera to display and record videos with Python" "Raspberry Pi Tutorial on using a USB Camera to display and record videos with Python" (<https://medium.com/propelland/raspberry-pi-tutorial-on-using-a-usb-camera-to-display-and-record-videos-with-python-a41c6938f89f>)
6. Raspberry Pi dokumentacija - WiFi veza (<https://www.raspberrypi.org/documentation/configuration/wireless/>)
7. Raspberry Pi projekat - "Build a Python Web Server with Flask" (<https://projects.raspberrypi.org/en/projects/python-web-server-with-flask>)
8. Flask sajt (<https://palletsprojects.com/p/flask/>)
9. [Twython · PyPI](#)
10. [Tweet on Raspberry Pi via Twython! \(Part 1\) \(deviceplus.com\)](#)