



Fakultet elektrotehnike i računarstva, Sveučilišta u Zagrebu
Zavod za elektroničke sustave i obradu informacija
Sveučilište u Zagrebu

Pametno zvonilo



- Δ Dio projekta „Pametna kuća“
- Δ Mogućnost programiranja i primjene Raspberry Pi računala
- Δ Korištenje Pushover aplikacije za primanje obavijesti

Sažetak

Svakome se dogodilo da je propustio dostavu ili nenajavljeni dolazak gostiju. Projekt pametnog zvona omogućava ukućanima da u svakom trenutku, neovisno jesu li kod kuće, znaju tko se nalazi na vratima kuće te tako izbjegnju prethodno opisane situacije.

U svijetu postoji nekoliko sličnih projekata koji na različite načine alarmiraju ukućane, od slanja mailova, sms-ova do obavještanja putem aplikacija. Jedna od najčešćih je Pushover aplikacija korištena u implementaciji ovog projekta.

Relativno jeftin sustav izveden pomoću Raspberry Pi računala, kamere i dugmeta primjenjiv je u svakom kućanstvu koje ima internetsku vezu te ga mogu koristiti svi ukućani koji posjeduju pametne telefone.

Najveći nedostatak predstavlja prijelaz projekta iz testnog u onaj koji je spreman za stvarno korištenja prilikom čega se mogu javiti problemi s napajanjem i potreba za korištenjem baterija.

Sadržaj

1. UVOD	3
2. OPIS SUSTAVA	4
2.1. Shema sustava	4
3. PREGLED KORIŠTENIH KOMPONENTI	5
3.1. Raspberry Pi	5
3.2. Kamera	6
3.3. Tipka (eng. PushButton)	7
3.4. Ostale komponente	7
3.5. Izgled sustava	7
4. PROGRAMSKA REALIZACIJA SUSTAVA	9
4.1. Pushover aplikacija	9
5. PROGRAMSKI KOD	10
6. REZULTAT IZVRŠAVANJA SKRIPTE	12
7. ZAKLJUČAK	14
8. LITERATURA	15
9. POJMOVNIK	16

Ovaj seminarski rad je izrađen u okviru predmeta „Sustavi za praćenje i vođenje procesa“ na Zavodu za elektroničke sustave i obradbu informacija, Fakulteta elektrotehnike i računarstva, Sveučilišta u Zagrebu.

Sadržaj ovog rada može se slobodno koristiti, umnožavati i distribuirati djelomično ili u cijelosti, uz uvjet da je uvijek naveden izvor dokumenta i autor, te da se time ne ostvaruje materijalna korist, a rezultirajuće djelo daje na korištenje pod istim ili sličnim ovakvim uvjetima.

1. Uvod

U današnje vrijeme gotovo je nemoguće zamisliti život bez tehnologija koje su se utkale u sve pore modernog društva. Velikim razvojem informatičke industrije dolazi do proširenja ponude čime proizvodi postaju dostupni širem spektru ljudi.

Ideja pametne kuće postoji već više desetljeća. Ovakav spoj tehnologije i stila življenja čini dom ljepšim i ugodnijim mjestom za život. Konačni cilj svakog takvog pametnog rješenja je automatizirati ili eliminirati sve one svakodnevne radnje koje vremenski, prostorno, sigurnosno ili na bilo koji drugi način ograničavaju te tako pozitivno utjecati na način života.

Nadzor kuće u odsustvu naše prisutnosti važan je jer nam omogućuje brzo reagiranje na neželjene događaje. Jednostavnim povezivanjem nekoliko komponenti moguće je osigurati svakom od ukućana uvid u ono što se događa na kućnim vratima.

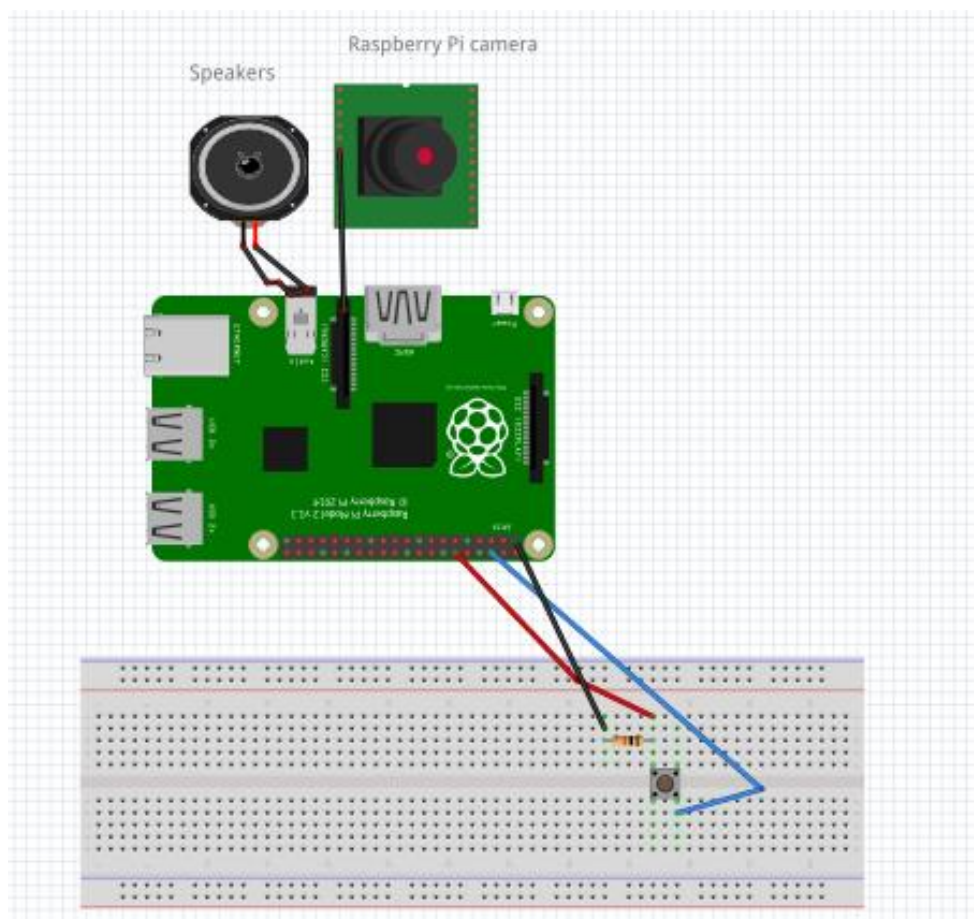
U nastavku su detaljno opisani koraci implementacije sustava pametnog zvona kao i mogućnosti za daljnji razvoj i nadogradnju.

2. Opis sustava

Sustav pametnog zvona sastoji se od Raspberry Pi računala, kamere dugmeta, zvučnika i pull-up otpornika koji su pomoću žica i *protoboarda* spojeni u funkcionalnu cjelinu.

Pritiskom na dugme aktivira se zvono te se pokreće kamera koja vrši slikavanje osobe koja je na vratima. Slika se uz popratnu poruku šalje ukućanima putem Pushover aplikacije. Ukućanima se nudi mogućnost pregledavanja propuštenih dolazaka poštaru, gostiju, susjeda i slično.

2.1. Shema sustava



Slika 1: Povezivanje komponenti sustava pametnog zvona

3. Pregled korištenih komponenti

3.1. Raspberry Pi

Raspberry Pi je računalo veličine kreditne kartice koje je u potpunosti smješteno na jednoj, matičnoj, ploči. Razvijen je u Ujedinjenom Kraljevstvu od strane „Raspberry Pi Foundation“ s ciljem promoviranja učenja osnova računalnih znanosti u školama. Raspberry Pi je sposoban raditi sve što i svako obično stolno računalo, od pregledavanja Interneta i gledanja filmova visoke rezolucije, do pisanja tekstova, tablica i igranja igrica. No što je za nas najvažnije, ima mogućnost interakcije s okolinom te takav predstavlja idealnu platformu za mnoge projekte.

U ovom projektu korišten je model Raspberry Pi 2. Odlikuje se ugrađenim procesorom Broadcom BCM2836 s 4 ARM Cortex-A7 jezgre koje rade na taktu od 900 MHz u pratnji 1 GB SDRAM-a što ga čini 6 puta bržim od prethodnika. Izgled jednog takvog modela prikazan je na slici 2.



Slika 2: Raspberry Pi 2

3.2. Kamera

U radu je korišten službeni model Raspberry Pi kamere koja nudi mogućnost snimanje videa i slika visoke kvalitete. Spaja se na Raspberry *ribbon* kablom te je vrlo malih dimenzija (25 mm x 20 mm x 9 mm) što je čini lako primjenjivom na veliki broj projekata.



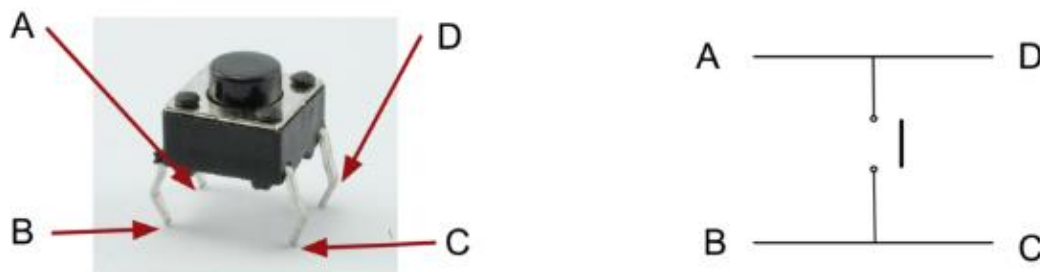
Slika 3: Službena Raspberry Pi kamera

3.3. Tipka (eng. PushButton)

Tipka je komponenta koja povezuje dvije točke u krug nakon što je pritisnuta.

Povezujemo tri žice na Raspberry Pi. Prva ide od jedne etape prekidača kroz pull-up otpornik ($10\text{ k}\Omega$) na 3,3 V. Druga ide iz odgovarajuće noge tipke na GND. Treća se spaja na I/O PIN broj 18 Raspberry Pija.

Kad je tipka otvorena ne postoji veza između dvije noge prekidača - pin spojen na 3,3 V (kroz pull-up otpornik) i čitamo HIGH. Kada je tipka pritisnuta, čini vezu između svoje dvije noge (LOW).



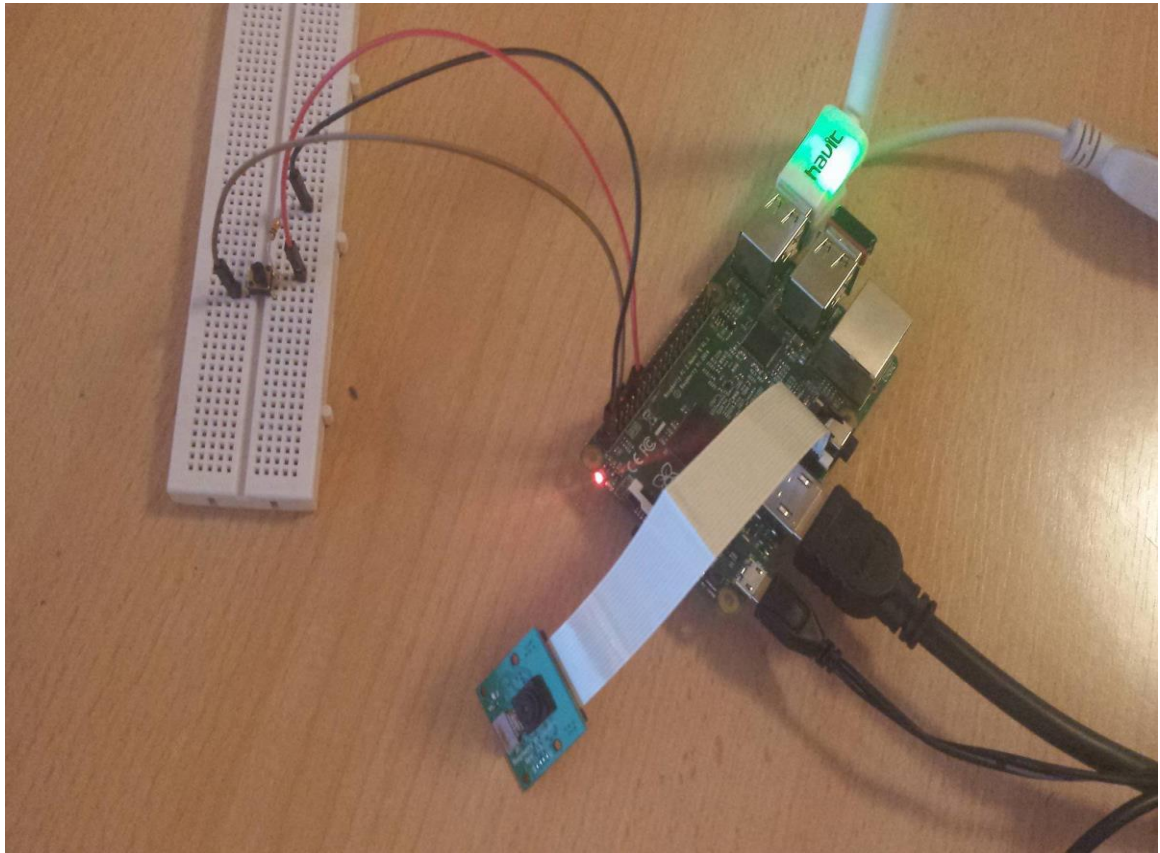
Slika 4: Električna shema tipke

3.4. Ostale komponente

Od ostalih komponenti korištene su: *protoboard*, žice, *pull-up* otpornik od $10\text{ k}\Omega$ i WiFi adapter.

3.5. Izgled sustava

U ovom koraku, potrebno je spojiti komponente prema gornjoj shemi sustava. Konačni izgled sustava je sljedeći:



Slika 5: Sustav pametnog zvona

4. Programska realizacija sustava

Proces programske realizacije sustava se odnosi prvenstveno na programiranje samog Raspberryja i temelji se na komunikaciji sustava s Pushover aplikacijom.

4.1. Pushover aplikacija

Pritiskom na zvono, obrađuju se notifikacije u obliku poruke i slike koje se zatim šalju na Pushover aplikaciju koju je prethodno potrebno instalirati na pametni telefon.

Nakon instalacije slijedi registracija na pushover.net i kreiranje aplikacije. Samom kreacijom, dobivamo *User Key* i *Application Token* kojim ćemo identificirati uređaj kojem šaljemo notifikacije prilikom pisanja python skripte. Nadalje, potrebno je unutar Pushover profila dodati uređaj koji će koristiti aplikaciju (mobitel, računalo,...). Ako je sve pravilno napravljeno, izgled Pushover profila je sljedeći:

The screenshot shows the Pushover user profile page. It is divided into several sections:

- Send a Notification:** Contains a form with fields for 'Send As' (Pushover (default)), 'Device' (All active devices), 'Sound' (Device default sound), 'Title' (optional), 'Message' (text area), and 'URL' (optional). A 'Send Notification' button is at the bottom.
- Your User Key:** Displays the user key 'urwj9aohqfdhf3y2718i1wz8wgqi2u' in a grey box. Below it, it shows the email 't43ogcvkfs@pomail.net' for receiving notifications from e-mails.
- Your Quiet Hours:** Shows 'You do not have any enabled quiet hours.' with an '(Edit)' link.
- Your Devices:** A table listing devices with columns for Name, Status, Last Synced, and Messages Received/Pending.

Name	Status	Last Synced	Messages Received/Pending
gt-I9505	Trial Period (Upgrade Now)	1 minute ago	19 received, 0 pending

Slika 6: izgled Pushover profila

Završivši ovaj dio, spremni smo za uhodavanje sustava i pokretanje skripte.

5. Programski kod

U ovom poglavlju ću proći kroz nekoliko bitnih dijelova koda. Program je pisan u skriptnom jeziku Python. Na samom početku potrebno je definirati koje module ćemo koristiti:

```
#!/usr/bin/python
from time import sleep
import os
import httplib, urllib
import RPi.GPIO as GPIO
import subprocess
import datetime
import MySQLdb
import socket
import fcntl
import struct
```

Import MySQLdb koristimo zbog stvaranja jednostavne baze podataka koja će nam služiti za spremanje slika (nažalost, u trenutku pisanja nije implementirana zbog problema s MySQL-om).

Slijedi vrlo bitno definiranje funkcija pomoću kojih tražimo IP adresu Raspberryja koja nam je potrebna za slanje slika na Pushover aplikaciju:

```
s = socket.socket(socket.AF_INET, socket.SOCK_DGRAM)
s.connect(('8.8.8.8', 0))
address = s.getsockname()[0]
```

Definiramo dvije metode koje će nam služiti za generiranje slika te slanje istih na Pushover:

```

def takepic(imageName):
    print("click")
    command = "sudo raspistill -o /var/www/pidoorbell/img/" + imageName
    print(command)
    os.system(command)

def PushOver(title,message,url):
    application_token = "ai2rxywnm2y1xh38hxxkotytathtywr"
    user_token = "urwj9aohqfdh3y2718i1wz8wgqi2u"
    # Start your connection with the Pushover API server
    conn = httplib.HTTPSConnection("api.pushover.net:443")

    # Send a POST request in urlencoded json
    conn.request("POST", "/1/messages.json",
    urllib.urlencode({
        "token": application_token,
        "user": user_token,
        "title": title,
        "message": message,
        "url": url,
    }), { "Content-type": "application/x-www-form-urlencoded" })

```

Nakon što su metode napisane, slijedi čitanje s GPIO i izvršavanje glavnog dijela skripte u kojem obrađujemo sliku te ju s porukom šaljemo na Pushover:

```

# PushOver('Doorbell','Started','')
print 'Doorbell Server Started\r'

while True:
    if (GPIO.input(18) == False):
        print 'Button Pushed!\r'

        os.system('mpg321 -g 100 /home/pi/doorbell-1.mp3 &amp;')

        now = datetime.datetime.now()
        timeString = now.strftime("%Y-%m-%d_%H_%M_%S")
        print("request received" + timeString)
        filename = timeString + '.jpg'

        print "B-----UTTON DOWN PRESSED"
        takepic(filename)

        #query="INSERT INTO images (id,url,date) VALUES (NULL," + filename + ",NULL)"
        #cur.execute("INSERT INTO images (id,url,date) VALUES (NULL,'" + filename + "',NULL)")
        PushOver('PiDoorBell','Ding Dong! Someone is ringing at your door!','http://' + address + '/pidoorbell/img/' + filename)
        #print query
        up = GPIO.input(18)
        count = count + 1

        sleep(.1)
    #if (GPIO.input(23)==False):
    #restart()

# this is never hit, but should be here to indicate if you plan on leaving the main loop

print "done"
sleep(0.2);

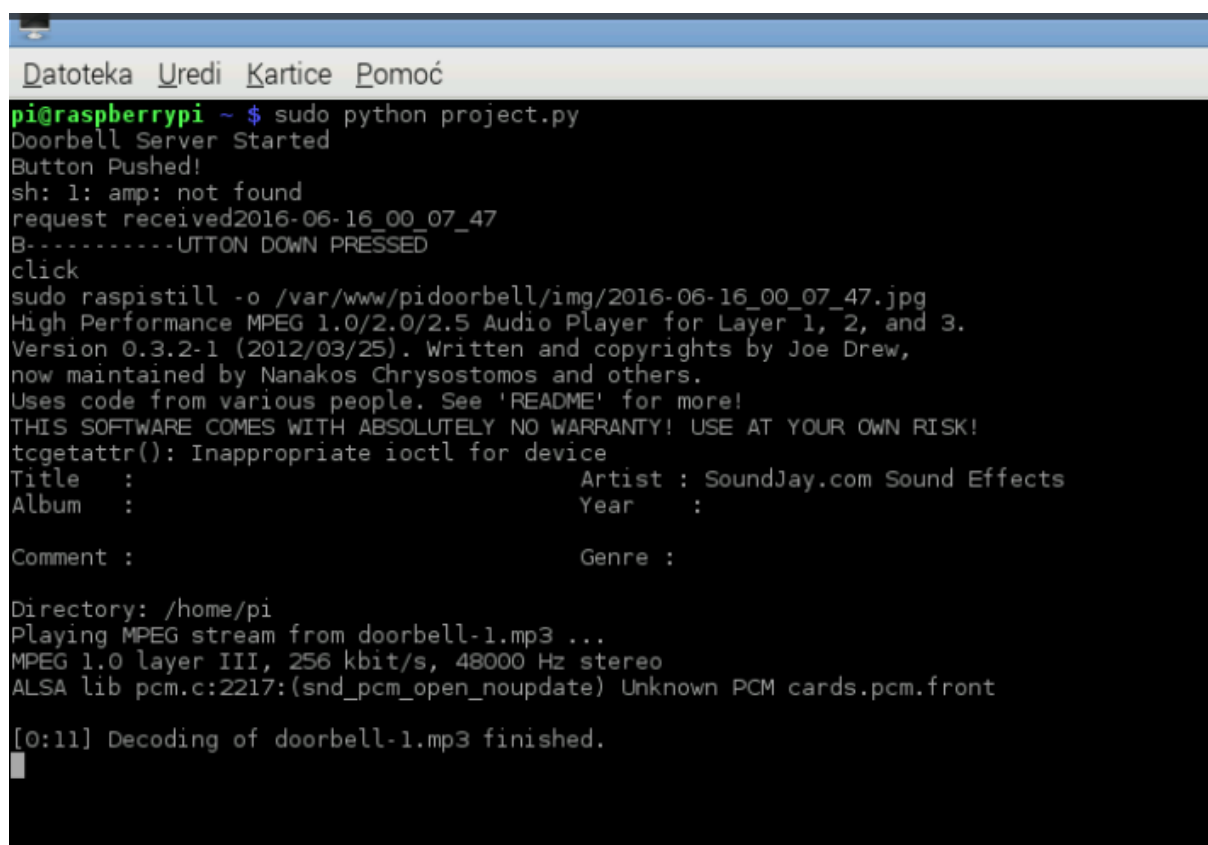
```

6. Rezultat izvršavanja skripte

Skriptu pokrećemo naredbom:

```
sudo python project.py
```

Ako je naredba prošla bez grešaka, izgled terminala je sljedeći:



```
Datoteka Uredi Kartice Pomoć
pi@raspberrypi ~ $ sudo python project.py
Doorbell Server Started
Button Pushed!
sh: 1: amp: not found
request received2016-06-16_00_07_47
B-----UTTON DOWN PRESSED
click
sudo raspistill -o /var/www/pidoorbell/img/2016-06-16_00_07_47.jpg
High Performance MPEG 1.0/2.0/2.5 Audio Player for Layer 1, 2, and 3.
Version 0.3.2-1 (2012/03/25). Written and copyrights by Joe Drew,
now maintained by Nanakos Chrysostomos and others.
Uses code from various people. See 'README' for more!
THIS SOFTWARE COMES WITH ABSOLUTELY NO WARRANTY! USE AT YOUR OWN RISK!
tcgetattr(): Inappropriate ioctl for device
Title : Artist : SoundJay.com Sound Effects
Album : Year :
Comment : Genre :
Directory: /home/pi
Playing MPEG stream from doorbell-1.mp3 ...
MPEG 1.0 layer III, 256 kbit/s, 48000 Hz stereo
ALSA lib pcm.c:2217:(snd_pcm_open_noupdate) Unknown PCM cards.pcm.front
[0:11] Decoding of doorbell-1.mp3 finished.
```

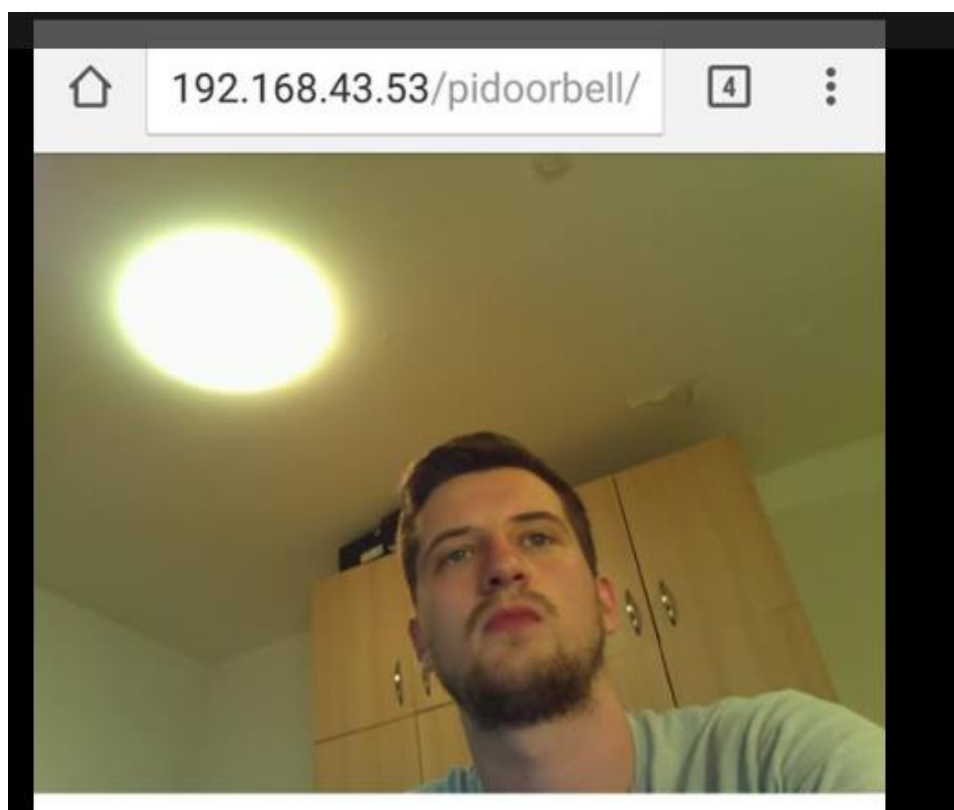
Slika 7: Prevođenje skripte u terminalu

Izgled sučelja Pushover aplikacije po dolasku obavijesti:



Slika 8: Izgled sučelja Pushover aplikacije

Slijedeći prikazani link dobivamo konačni rezultat izvođenja skripte:



Slika 9: Obavijest u obliku slike na Pushover aplikaciji

7. Zaključak

Razvijeni sustav pametnog zvona omogućuje kontrolu nad dolascima gostiju na kućna vrata. Jednostavan je za implementaciju te ga može realizirati svatko s malo znanja o programiranju ugradbenih računalnih sustava i velikom željom za učenjem novih stvari.

Sustav pametnog zvona je primjenjiv u svakom kućanstvu koje ima internetsku vezu te ga je moguće koristiti sa svakog uređaja koji se može povezati na Internet. Korist od sustava ne mogu imati korisnici koji nemaju pristup internetu te oni koji ne znaju koristiti pametni telefon.

U ovom radu je implementiran testni sustav koji nam daje uvid u mnogobrojne mogućnosti pametnog zvona. Za stvarno korištenje ga je ipak potrebno nadograditi postavljanjem komponenti u kućište i izvedbom napajanja koja je prikladnija vanjskim uvjetima.

8. Literatura

- [1] Pushover tutorial URL: <http://www.expertreviews.co.uk/technology/1404017/best-raspberry-pi-projects/page/0/2>
- [2] PushButton tutorial: URL: <http://razzpisampler.oreilly.com/ch07.html>
- [3] PiDoorBell tutorial URL: <http://blog.prototypecreations.net/2015/04/07/raspberry-pi-doorbell-and-camera-part-1/>
- [4] SQL konfiguracija URL: <https://www.stewright.me/2012/09/tutorial-install-phpmyadmin-on-your-raspberry-pi/>

