

Mladi za napredek Maribora 2014

31. Srečanje

PAMETNA AVTOMATIZACIJA HIŠE

Raziskovalno področje: elektrotehnika, elektronika

Inovacijski predlog

OECD: KVVVOŠROZ Á ÒÕVŠOE

T ^} q { ! K Á T Š O E Á Q Q

[illegible]

Leto 2014

1 Kazalo vsebine

1	Kazalo vsebine	2
2	Kazalo slik	3
3	Povzetek	4
4	Vsebinski del	5
4.1	Opredelitev problema	5
4.2	Arduino.....	6
4.2.1	Programiranje arduino	6
4.3	Programiranje pametnega telefona	8
4.4	Bluetooth modul hc-06	9
4.5	Servomotor	10
4.6	Vhodna vrata in matrična tipkovnica	11
4.6.1	Vezava elementov z Arduinom	12
4.6.2	Program v programskem okolju Arduino	12
4.7	Komunikacija android-Arduino.....	14
4.7.1	Pošiljanje podatkov iz telefona.....	14
4.7.2	Sprejem podatkov iz telefona	16
4.8	Izdelava aplikacije s programom app inventor.....	19
4.8.1	Ikona aplikacije	19
4.8.2	Kodiran vstop v aplikacijo.....	19
4.8.3	Meni za upravljanje hiše.....	20
4.8.4	Povezava z bluetooth modulom.....	20
4.8.5	Vklop in izklop luči	21
4.8.6	Krmiljenje servomotorjev.....	22
4.8.7	Garažna vrata.....	23
5	Sklep	24
6	Viri	25

2 Kazalo slik

Slika 1: Arduino mega 2560.....	6
Slika 2: Primer programa za utripaločo LED diodo.....	7
Slika 3: Vizuelno oblikovanje aplikacije za Android OS	9
Slika 4: Blokovno programiranje aplikacije za Android OS.....	9
Slika 5: Sprednja stran bluetooth modula HC-06	10
Slika 6: Zadnja stran bluetooth modula HC-06.....	10
Slika 7: Servo motor	10
Slika 8: Vezava elementov z mikrokontrolerjem Arduino	12
Slika 9: Program matrične tipkovnice.....	13
Slika 10: App Inventor pošiljanje podatkov.....	14
Slika 11: Vezava elementov z mikrokontrolerjem Arduino	15
Slika 12: Serijski port za sprejem podatkov.....	16
Slika 13: Prejem podatkov iz telefona	17
Slika 14: Blokovni diagram komunikacije telefona z mikrokontrolerjem.....	18
Slika 15: Ikona aplikacije Avtomatizacija_hiše	19
Slika 16: Vnos gesla za vstop v aplikacijo	20
Slika 17: Vzpostavljanje povezave z bluetooth modulom	21
Slika 19: Tipki za vklop in izklop.....	21
Slika 18: Meni za luči	21
Slika 20: Primer vklopa treh od petih luči.....	22
Slika 21: Meni za rolete	22
Slika 22: Meni za garažna vrata	23

3 Povzetek

Inovacijska naloga prikazuje rešitev in olajšanje vsakdanjika posamezniku, kot je nadziranje, katere luči so prižgane, kateri prostori so zatemnjeni ter odpiranje in zapiranje garažnih vrat. Velikokrat se zgodi, da pozabimo vklopljeno luč, pozabimo zapreti garažna vrata. Da bi to preprečil, sem si izmislil aplikacijo za mobilni pametni telefon Android, preko katerega lahko vklapljam ali izklapljam luč v posameznem prostoru, spustimo in dvignemo lahko rolete v hiši in tako nadziramo osvetlitev prostorov. Aplikacija nam omogoča tudi, da odpremo ali zapremo garažna vrata. Aplikacija je zasnovana tako, da nam pokaže stanja luči, rolet in garažnih vrat. To pomeni, da lahko preprosto vidimo, katere luči so vklopljene, v katerih prostorih so dvignjene oz. spuščene rolete, ter ali so garažna vrata odprta ali zaprta.

Izdelal sem tudi sistem, s katerim lahko vhodna vrata zaklenemo ali odklenemo z matrično tipkovnico. Predhodno si moramo le nastaviti geslo. V tem primeru nikoli več ne bomo potrebovali ključa za vstop v hišo.

4 Vsebinski del

4.1 Opredelitev problema

V Sloveniji in vseh ostalih državah imamo v veliki večini skoraj vsi doma urejeno razsvetljavo, za zatemnitev prostorov uporabljamo rolete ali žaluzije, vhodna vrata odklepamo in zaklepamo s ključem, ki ga je verjetno že prav vsak kdaj kam založil ali izgubil.

Problem se pojavi, kadar opravljamo več del hkrati pa tudi kadar gledamo televizijo ipd. Naenkrat ne moremo biti v več prostorih, zato ne moremo preveriti, če smo kje pustili vklopljeno luč, ali so prostori zvečer zatemnjeni oz. osvetljeni, ko vzide sonce. Da pozabimo ključ, se navadno zgodi takrat, ko to najmanj pričakujemo, morda smo prišli domov na hitro po kakšno stvar, ki je nujna za našo kariero, a takrat ne moramo nič.

Dejstvo je, da število uporabnikov pametnih telefonov narašča, največji porast je po raziskavah opažen pri dijakih in študentih ter pri prebivalcih do 40-ih let starosti. Iz tega lahko sklepamo, da je velika večina prebivalcev, ki se odloča za izgradnjo nove hiše, uporablja pametni mobilni telefon.

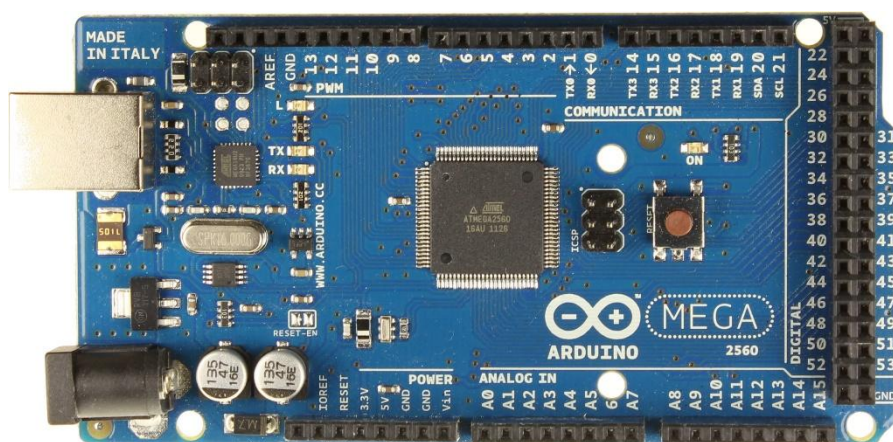
Da bi čim bolj olajšal posameznikov vsakdan, sem izdelal aplikacijo za pametni telefon Android, s katero lahko preko bluetooth povezave vklopimo in izklopimo vse luči, katere imamo povezane na mikrokontroler Arduino. Prav tako nam aplikacija omogoča zatemnitev ali osvetlitev prostorov, odpremo in zapremo pa lahko tudi garažna vrata.

Za nadzor luči sem uporabil LED diode, za rolete in garažna vrata pa servomotorje. Le te sem uporabil zaradi izdelave makete, v realnosti pa bi lahko to storili preko relejskh stikal.

Odklepanje in zaklepanje vhodnih vrat sem izdelal s pomočjo matrične tipkovnice, tako nikoli več ne bomo potrebovali ključa. Program je zaradi izdelave makete ponovno napisan tako, da se ob pravilnem geslu servomotor obrne v eno smer, ob pritisku tipke za zaklepanje, pa v drugo smer.

4.2 Arduino

Mikrokrmilnik Arduino je univerzalni mikrokrmilnik, zasnovan na ATmel tehnologiji in je idealen za področje elektronike in robotike. Isto platformo uporablja tudi znani robotski sesalnik iRobot – Roomba, Scooba in ostali iz družine. Arduino je odprto kodna mikrokontrolna platforma, zasnovana na enostavni razvojni ploščici z vhodno/izhodnimi konektorji in uporabniku prijaznem razvojnem okolju (brezplačna programska oprema). Arduino je lahko samostojen ali pa povezan z vašim računalnikom.



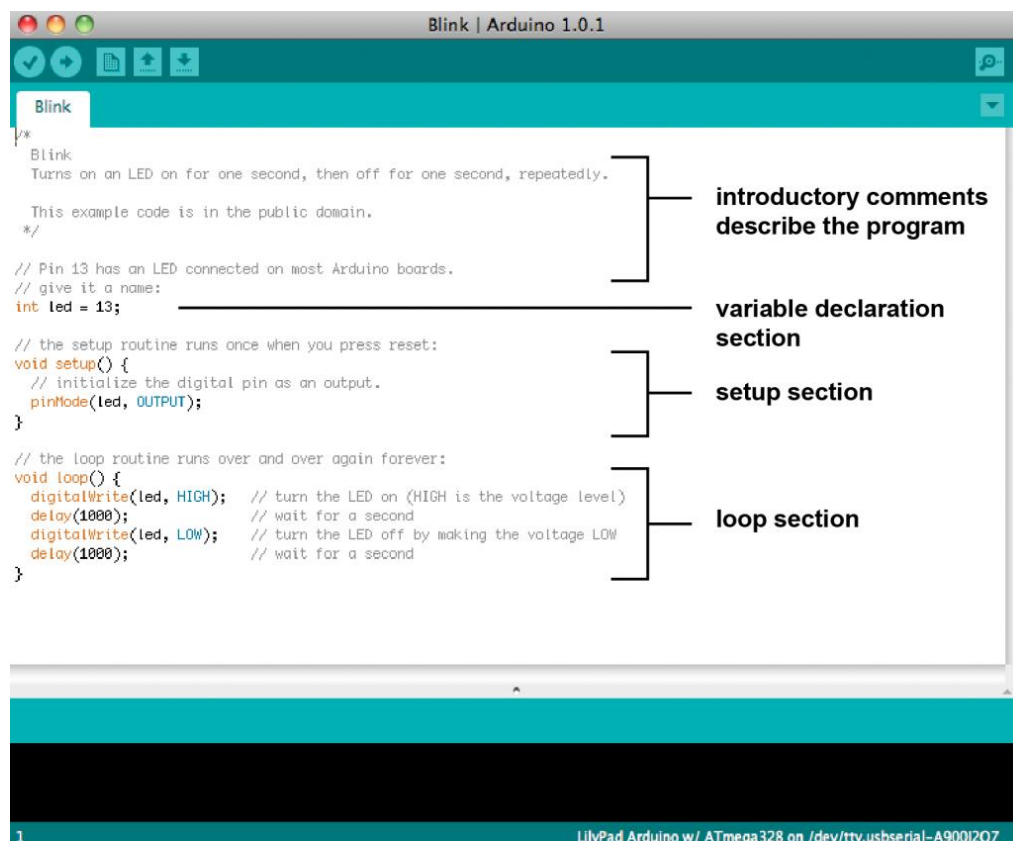
Slika 1: Arduino mega 2560

Sam sem za svojo nalogo uporabil mikrokrmilnik Arduino Mega 2560, ki temelji na ATmega 2560 mikrokontrolerju. Ima 54 digitalnih vhodov/izhodov od katerih jih lahko 14 deluje kot PWM izhodi, 16 analognih vhodov, 4UARTe, USB povezavo, napajalni priključek, ICSP priključek ter gumb za resetiranje. Zanj sem se odločil predvsem zaradi velikega števila digitalnih vhodov/izhodov, kar mi je omogočilo priključitev vseh servomotorjev, LED diod in numerične tipkovnice.

4.2.1 Programiranje arduino

Razvojno okolje Arduino lahko brezplačno prenesemo s spletnega naslova (<http://arduino.cc/en/main/software>). Možno ga je namestiti na več operacijskih sistemov. Sam sem ga uporabljal v kombinaciji z Windows 7. Verzija programa, ki jo imam nameščeno, je 1.5.5, uporabljal pa sem jo v slovenskem jeziku.

Arduino programski jezik temelji na programskem jeziku C, ki so ga razvili v začetku 70-ih let prejšnjega stoletja za uporabo na UNIX operacijskem sistemu. Za tiste, ki na ta jezik niso navajeni, je težko berljiv in razumljiv. Zaradi teh razlogov je Arduino ekipa pripravila standardno Arduino knjižnico, ki vsebuje nabor enostavnih in uporabno naravnanih funkcij. Na ta način je možno programirati Arduino mikrokontroler kolikor se le da enostavno. Arduino knjižnice so same po sebi napisane v programskem jeziku C++. Pri tem se moramo zavedati, da čeprav uporabljamo funkcije iz Arduinovih knjižnic, bo vse funkcioniralo enako kot v jeziku C.



Slika 2: Primer programa za utripaločo LED diodo

Arduino program mora vedno vsebovati metodi *setup* in *loop*, drugače program ne bo deloval.

Metoda *setup* se izvede samo enkrat, in sicer ob zagonu programa, ko priklopimo modul na napajanje ali kadar ga resetiramo. V tej metodi po navadi pripravimo program, preden se začne izvajati. Izvedemo inicializacijo spremenljivk, nastavitve digitalnih in analognih priključkov ter pripravimo uporabljene knjižnice za nadaljno uporabo.

Metoda *loop* se prične izvajati takoj za metodo *setup*. Kot že ime samo pove, se ta metoda ponavlja in sicer v neskončnost. Na tej metodi temelji celoten algoritem programa. Koda mora biti napisana tako, da znotraj te metode preverjamo, do katerih sprememb je prišlo na priključkih Arduino modula in nanje ustrezno odreagiramo. Pri tem se moramo za zavedati, da je mikrokrmilnik pri obdelavi ukazov zelo hiter, tako lahko v 1 sekundi izvede tudi do milijon vrstic kode, kar pomeni, da se lahko metoda *loop* v eni sekundi izvede tudi več tisočkrat.

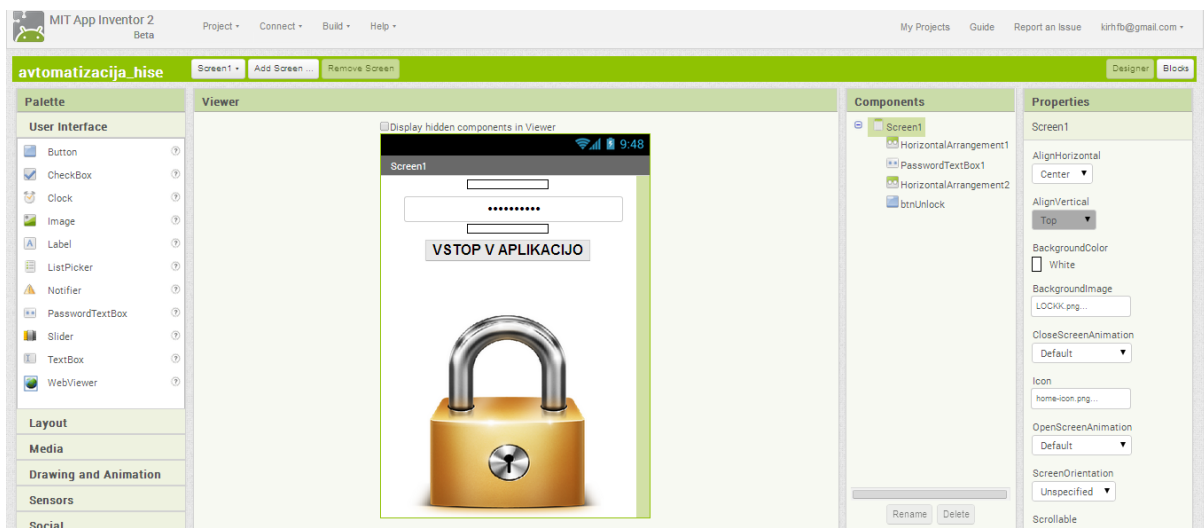
4.3 Programiranje pametnega telefona

AppInventor (<http://www.appinventorbeta.com/>) je izredno perspektivno orodje za preprosto izdelavo programov za Android MIT (Massachusetts Institute of Technology).

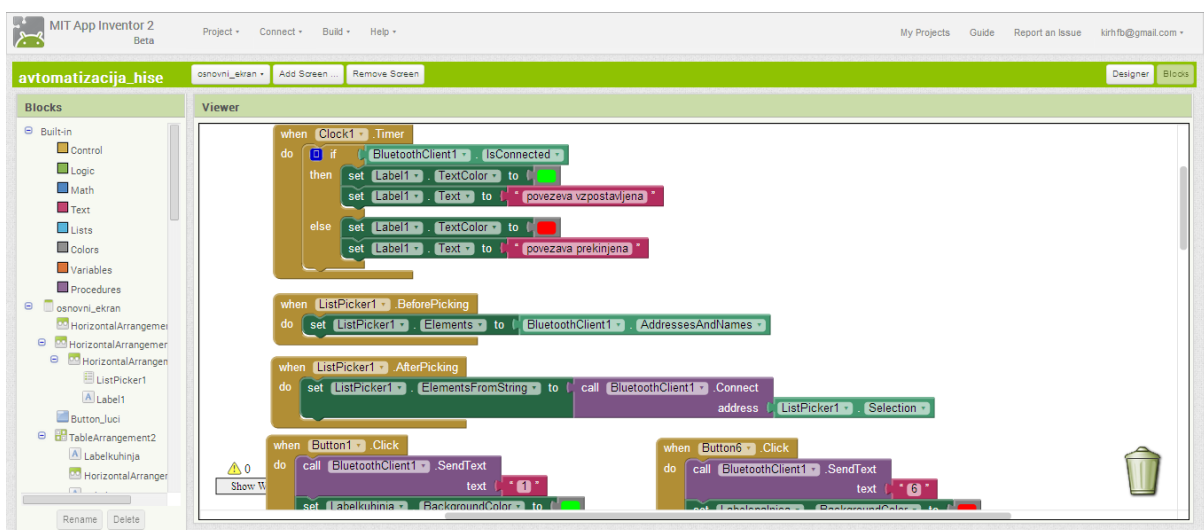
To orodje spada med spletne programe. Razvoj programa in vso logiko delovanja razvijemo s spletnim brskalnikom.

Izdelava programa s tem orodjem traja precej dolgo. Temu primerna je tudi svoboda, ki jo imamo pri tem, saj lahko program popolnoma prilagodimo svojim potrebam. Na voljo imamo gumbke, besedilna polja, slike, video predvajalnike... Dostopamo lahko tudi do senzorjev telefona. Skratka, možnosti za izdelavo programa je ogromno. Tudi videz je v povsem naši domeni. Gre za nekakšno zlaganje kock, od katerih pa ima vsaka lastnosti, ki jo lahko spremenimo. Po končani izdelavi uporabniškega vmesnika sledi tisti težji del, ko moramo vsem tem elementom dodati funkcije. Gumbek sam po sebi ne zna nič narediti, dokler mu ne povemo, kaj želimo od njega. Tukaj pa sledi vstop v programerski svet. Po kliku možnosti BlockEditor se nam odpre stran za zlaganje teh kock. Programiranje v tem primeru ne poteka z vpisovanjem kode, ampak z zlaganjem blokov, ki pa so oblikovani tako, da takoj vidimo, kaj spada skupaj. Če dobro pogledamo, vidimo, da gre pravzaprav za nekakšno grafično pisanje kode. Precej zanimiv pristop k lažjemu programiranju, čeprav roko na srce, osnovno programersko logiko moramo pa v tem primeru le poznati. Edino kar nam je oproščeno, je nameščanje zapletenih programov in znanje programskega jezika.

Program omogoča povezavo z našim pametnim telefonom Android, preko povezave WiFi. To nam omogoča, da sproti spremljamo, kaj se s programom dogaja, kako dejansko izgleda na telefonu. Za to potrebujemo MIT-ovo aplikacijo nameščeno na telefonu, katero prenesemo iz trgovine Play.



Slika 3: Vizuelno oblikovanje aplikacije za Android OS

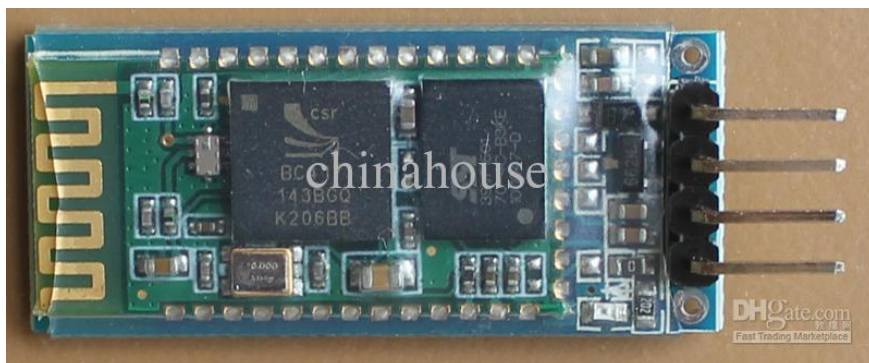


Slika 4: Blokovno programiranje aplikacije za Android OS

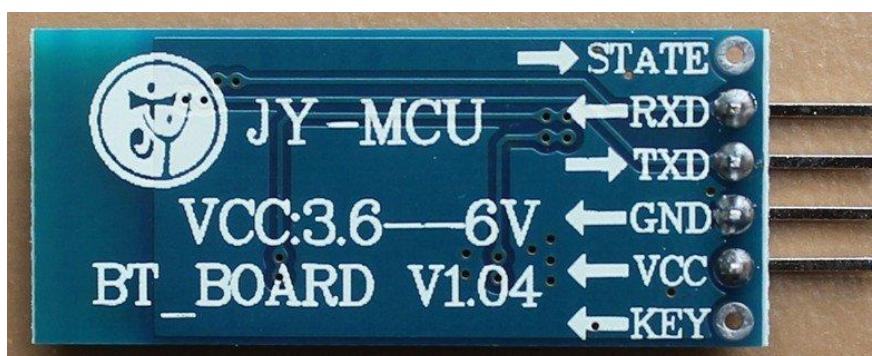
4.4 Bluetooth modul hc-06

Bluetooth modul HC-06 nam omogoča enostavno prejemanje in oddajanje podatkov med napravo, ki ima vgrajen bluetooth in Arduino mikrokontrolerjem, na katerega ga priključimo preko spodaj navedenih povezav:

- VCC: priključek za enosmerno napetost v območju med 3,6 V in 6 V. Sam sem v projektu uporabljal 5 V.
- GND: ozemljitev.
- TXD: serijski izhod modula, ki je povezan z RX vrati na mikrokontrolerju.
- RXD: serijski vhod modula, ki je povezan z TX vrati na mikrokontrolerju.



Slika 5: Sprednja stran bluetooth modula HC-06



Slika 6: Zadnja stran bluetooth modula HC-06

4.5 Servomotor

Prednost teh motorjev je ta, da lahko zelo natančno nastavimo oziroma spremljamo njihovo pozicijo. Značilnost tega motorja je, da končno gred motorja premika le v obsegu 180° (od -90° do $+90^\circ$). Ker se ti motorji tako množično uporabljajo, je njihova cena nizka, poleg



Slika 7: Servo motor

tega pa se odlikujejo po odličnih navorih.

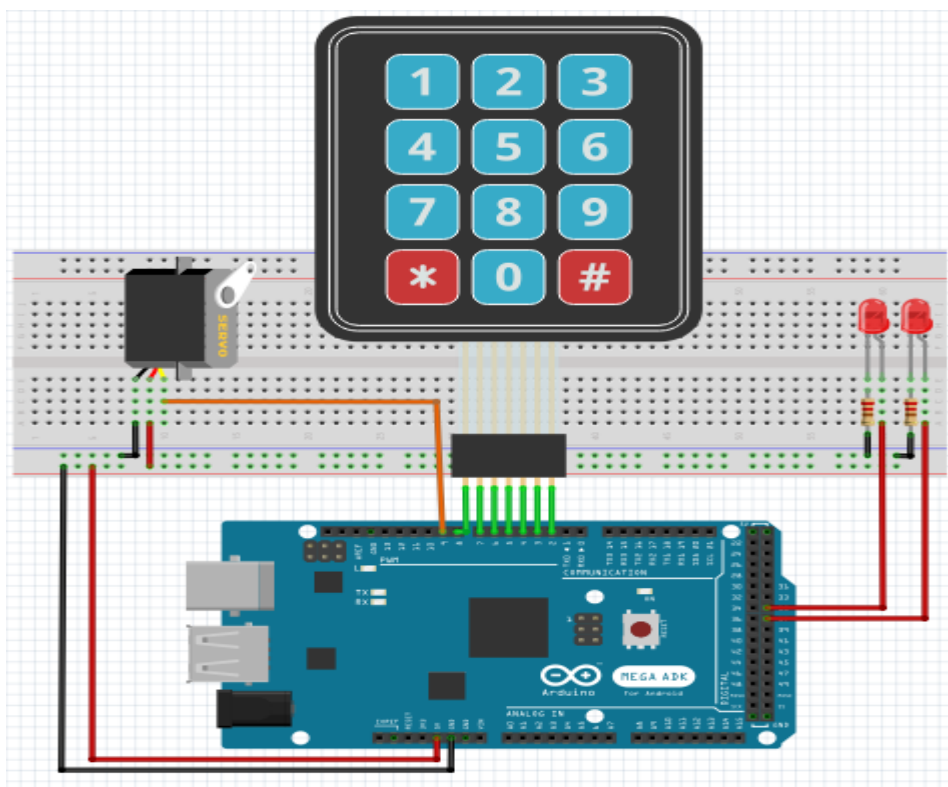
Motor za priklop potrebuje tri priključke, dva za napajanje (+,-) in priključek za servo-signal. Pozicijo servomotorja lahko v programu Arduino točno določimo, določimo čas, ki ga naj motor potrebuje, da pride iz ene lokacije do druge.

4.6 Vhodna vrata in matrična tipkovnica

Kot sem že v uvodu povedal, nam odklepanje vhodnih vrat z matrično tipkovnico olajša vse skrbi s ključem. Vse kar si moramo zapomniti je geslo, ki ga predhodno zapišemo v program mikrokontrolerja.

Odklepanje vhodnih vrat poteka tako, da v tipkovnico vpišemo geslo, nato pa pritisnemo zvezdico (*). Ko se vrata odklenejo, se vklopi signalna LED dioda zelene barve. V tem trenutku se obrne tudi servomotor, ki na maketi deluje kot loputa, katera se v tem trenutku dvigne.

Vhodna vrata zaklenemo s pritiskom na lojtro (#) na tipkovnici, takrat zelena signalna LED dioda izklopi, vklopi pa se LED dioda rdeče barve. Servomotor se za enak kot obrne v obratno smer, tako da se loputa na maketi spusti.



Slika 8: Vezava elementov z mikrokontrolerjem Arduino

4.6.1 Vezava elementov z Arduino

Matrična tipkovnica je na mikrokontroler povezana preko digitalnih vhodov (2, 3, 4, 5, 6, 7, 8). Rdeča in zelena signalna LED dioda, sta priključeni na digitalna vhoda 35 in 37. Zaporedno z uporoma sta vezana še upora, vrednosti $100\ \Omega$. Servomotor je priključen na napetost 5 V, na maso in pa na digitalni vhod 9, preko katerega mu določamo položaj.

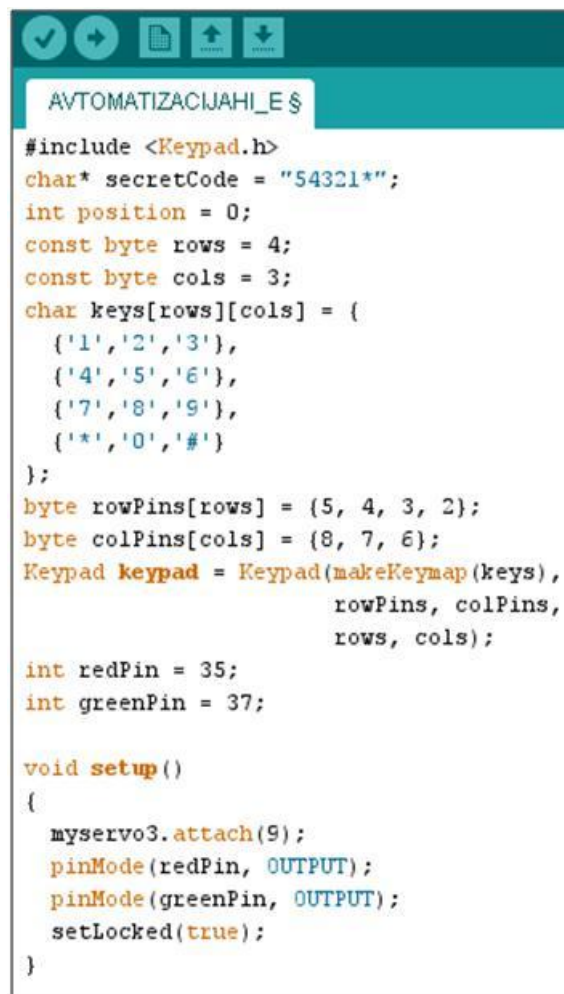
4.6.2 Program v programskem okolju Arduino

V programskem okolju Arduino najprej vstavimo knjižnico, imenovano »Keypad«. Knjižnica je dostopna na uradni spletni strani Arduino.

Določimo geslo, v mojem primeru »54321*«. Nato določimo število vrstic in stolpcev, ki jih vsebuje naša matrična tipkovnica, skiciramo postavitev tipk.

Nato najprej vnesemo digitalne vhode, na katere so povezane vrstice matrične tipkovnice. Enako storimo še za stolpce tipkovnice.

Določimo na katere digitalne vhode priključimo zeleno in rdečo signalno LED diodo. Določimo tudi digitalni izhod za servomotor in nastavimo digitalna priključka za LED diodi kot izhoda.



```
#include <Keypad.h>
char* secretCode = "54321*";
int position = 0;
const byte rows = 4;
const byte cols = 3;
char keys[rows][cols] = {
  {'1', '2', '3'},
  {'4', '5', '6'},
  {'7', '8', '9'},
  {'*', '0', '#'}
};
byte rowPins[rows] = {5, 4, 3, 2};
byte colPins[cols] = {8, 7, 6};
Keypad keypad = Keypad(makeKeymap(keys),
                        rowPins, colPins,
                        rows, cols);

int redPin = 35;
int greenPin = 37;

void setup()
{
  myservo3.attach(9);
  pinMode(redPin, OUTPUT);
  pinMode(greenPin, OUTPUT);
  setLocked(true);
}
```

Slika 9: Program matrične tipkovnice

V nadaljevanju programa sledijo še funkcije, katere se izvedejo v primeru odklepanja oz. zaklepanja vhodnih vrat. Torej, ob vpisu pravilnega gesla se servomotor obrne od 0° do 180° v korakih po 1°. Da servomotor doseže končni položaj, potrebuje 15 ms. Zaklepanje vhodnih vrat poteka popolnoma enako, le da je začetni položaj servomotorja 180°, končni pa 0°. Tudi pri zaklepanju vrat servomotor potuje v korakih po 1°, za doseg končnega položaja pa potrebuje 15 s.

4.7 Komunikacija android-Arduino

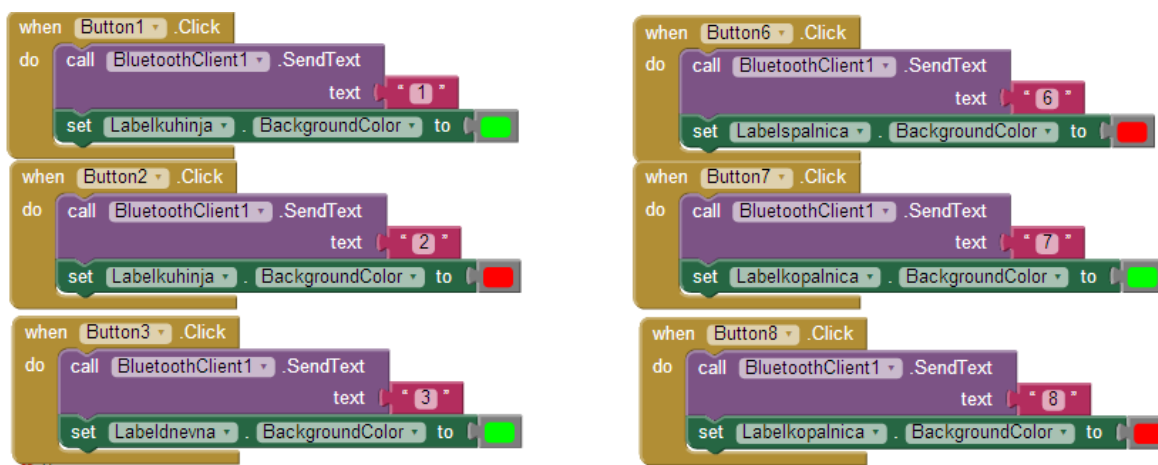
Komunikacija med pametnim mobilnim telefonom Android in mikrokontrolerjem Arduino poteka preko bluetooth modula. Kako je povezan bluetooth modul z Arduino, sem že omenil. Povezava poteka preko bluetooth prenosa podatkov (ang: baud rate), frekvence 9600 Hz.

Pošiljanje podatkov je precej preprosto. Aplikacija pošlje nek podatek, v mojem primeru številko ali črko. Ta podatek mikrokontroler zazna, in glede na napisan program izvede operacijo, ki jo določimo za posamezen podatek.

4.7.1 Pošiljanje podatkov iz telefona

Spletno programsko okolje App Inventor nam omogoča enostavno pošiljanje podatkov. V knjižnici programa poiščemo modul za pošiljanje teksta preko bluetootha.

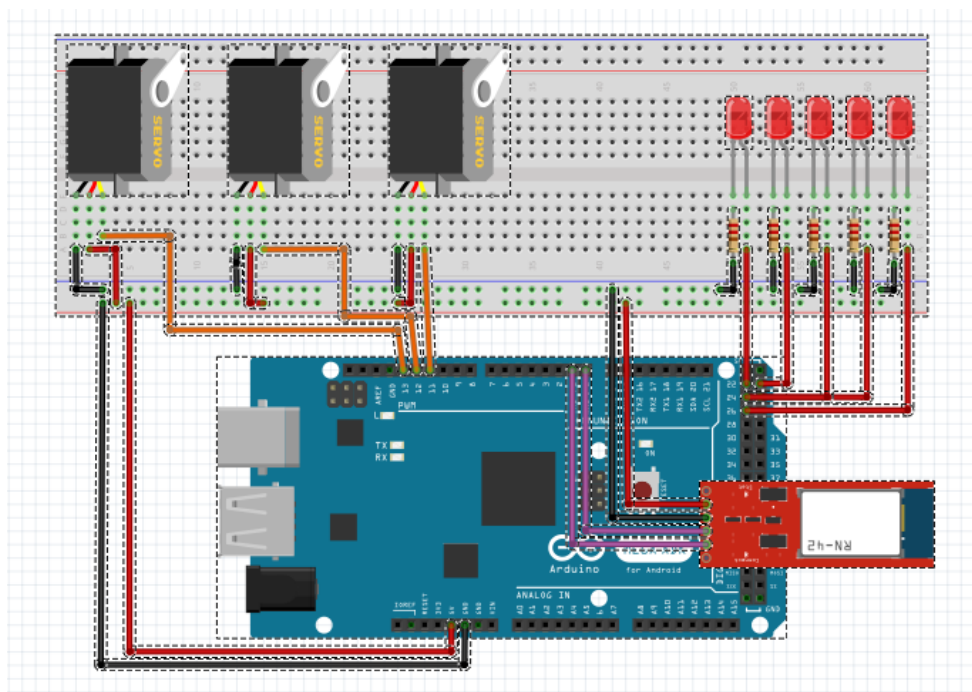
Kot je prikazano spodaj na sliki, najprej izberemo tipko, ob pritisku katere želimo poslati podatek preko bluetootha. Vse poteka v blokovnem programiranju App Inventor. Znotraj bloka za tipko povlečemo funkcijo, ki omogoča pošiljanje podatkov preko bluetootha. V to funkcijo povlečemo še tekstovni blok, v katerega napišemo željen podatek, ki ga bomo poslali ob pritisku na določeno tipko. V mojem primeru tipka 1 predstavlja tipko za vklop luči v kuhinji in ob pritisku le te, pametni telefon Android pošlje število 1 preko bluetootha.



Slika 10: App Inventor pošiljanje podatkov

Tabela prikazuje podatke, ki jih pošljemo s pritiskom na določeno tipko v aplikaciji Avtomatizacija_hiše.

<i>Tipka v programu Avtomatizacija_hiše</i>	<i>Pošlji podatek</i>
<i>Luč kuhinja vklop</i>	1
<i>Luč kuhinja izklop</i>	2
<i>Luč dnevna soba vklop</i>	3
<i>Luč dnevna soba izklop</i>	4
<i>Luč spalnica vklop</i>	5
<i>Luč spalnica izklop</i>	6
<i>Luč kopalnica vklop</i>	7
<i>Luč kopalnica izklop</i>	8
<i>Luč zunanje vklop</i>	9
<i>Luč zunanje izklop</i>	A
<i>Roleta spalnica dvig</i>	B
<i>Roleta spalnica spust</i>	C
<i>Rolera kopalnica dvig</i>	D
<i>Roleta kopalnica spust</i>	E
<i>Garažna vrata odpri</i>	F
<i>Garažna vrata zapri</i>	G



Slika 11: Vezava elementov z mikrokontrolerjem Arduino

4.7.2 Sprejem podatkov iz telefona

```
Serial.begin(9600);  
// serijski port začne prejemati podatke  
}
```

Slika 12: Serijski port za sprejem podatkov

V programskem okolju Arduino najprej določimo vhodne in izhodne spremenljivke, enako kot pri matrični tipkovnici. Nato določimo serijski vhod za sprejem podatkov, kot prikazuje slika zgoraj.

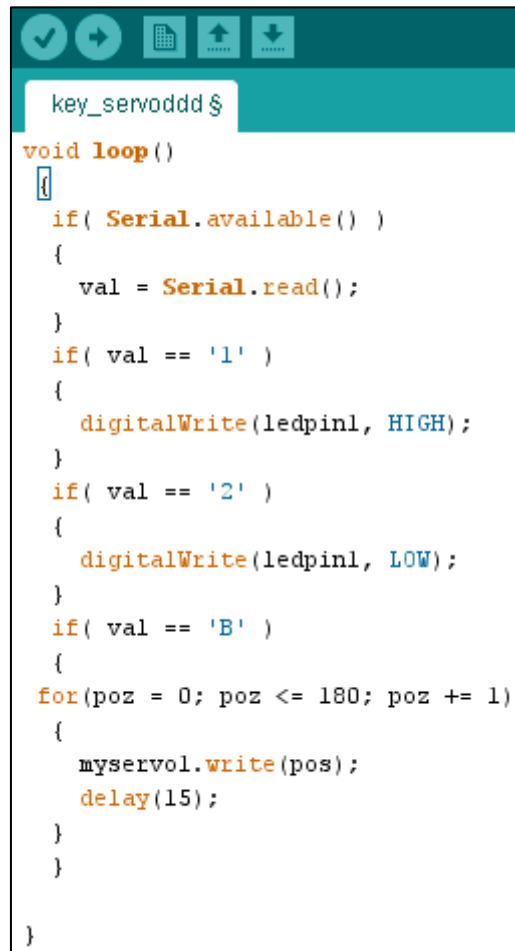
Slika 13 prikazuje program, napisan v programskem okolju Arduino, za sprejem podatkov.

Najprej program preveri, če je prispel kakšen podatek.

V primeru, da podatek prispe, program preveri, ali je za določen prispeli podatek napisan program za izvedbo.

V primeru da prejmemo število 1, LED dioda 1 zasveti, ko pa prejmemo število 2, le ta ugasne.

Enak postopek velja za servomotorje. Kadar prejmemo črko B, se servomotor iz pozicije 0° obrne za 180° v korakih po 1°, za pot pa potrebuje 15 ms.



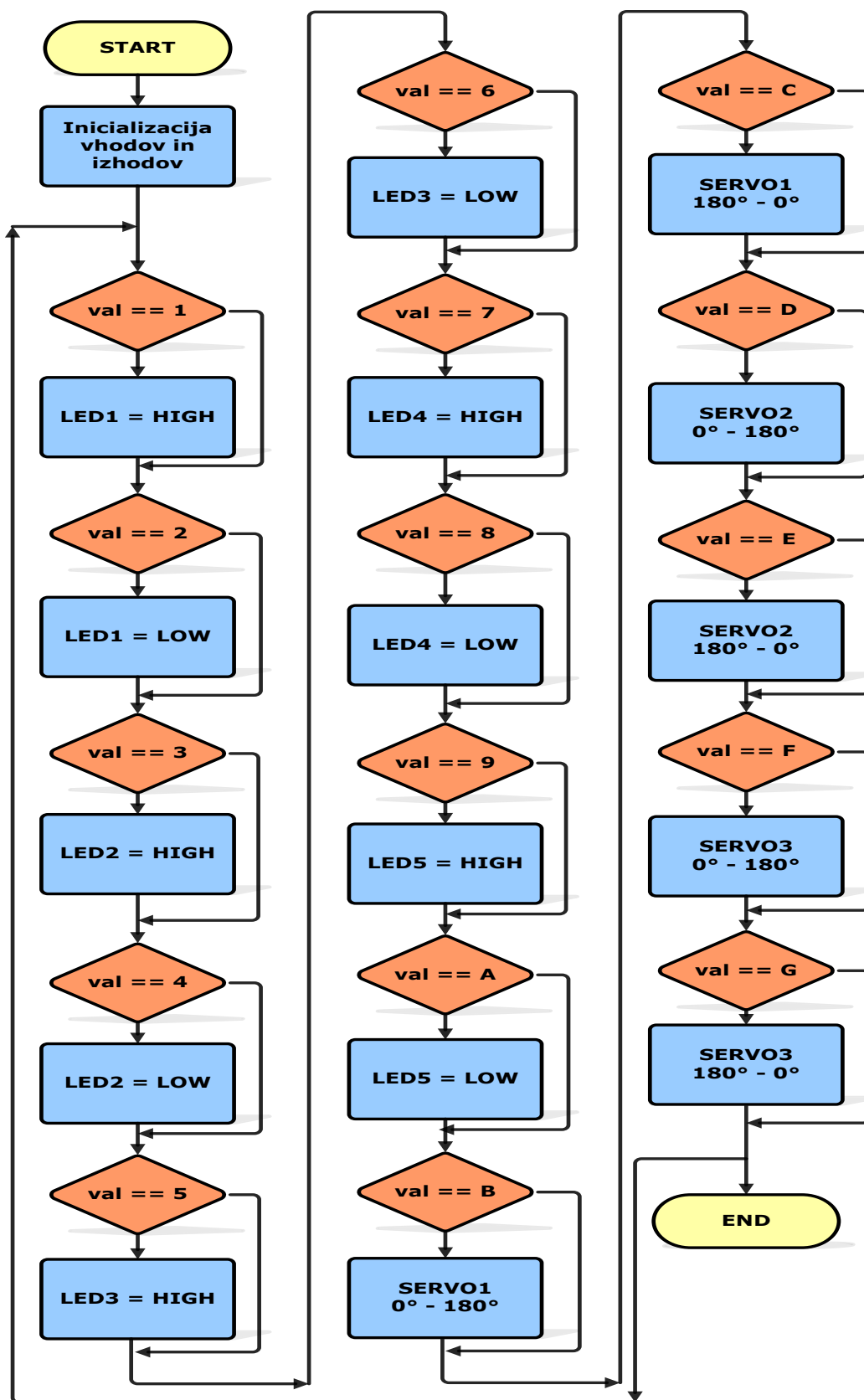
```
key_servoddd$  
void loop()  
{  
  if( Serial.available() )  
  {  
    val = Serial.read();  
  }  
  if( val == '1' )  
  {  
    digitalWrite(ledpin1, HIGH);  
  }  
  if( val == '2' )  
  {  
    digitalWrite(ledpin1, LOW);  
  }  
  if( val == 'B' )  
  {  
    for(poz = 0; poz <= 180; poz += 1)  
    {  
      myservol.write(pos);  
      delay(15);  
    }  
  }  
}
```

Slika 13: Prejem podatkov iz telfona

Diagram poteka (slika 14) sem izdelal v programskem okolju Visio. Diagram prikazuje operacije, ki se izvedejo ob prejemu posameznega podatka preko bluetooth povezave.

V primeru, da prejmemo podatek 8, izklopimo LED diodo 4. Če pogledamo tabelo zgoraj, vidimo, da smo izklopili razsvetljavo v kopalnici.

V primeru, da prejmemo podatek C, se servomotor 1 zavrti iz 180° na 0°. Roleta v spalnici se spusti.



Slika 14: Blokovni diagram komunikacije telefona z mikrokontrolerjem

4.8 Izdelava aplikacije s programom app inventor

Kot sem že omenil, sem aplikacijo za krmiljenje luči, rolet in garažnih vrat, izdelal s spletnim programom App Inventor. Najprej je na vrsto prišlo grafično oblikovanje aplikacije, oz. z drugimi besedami povedano, kako bo aplikacija izgledala, kadar jo bomo pognali na našem Android pametnem telefonu.

4.8.1 Ikona aplikacije

V meniju Android telefonov se nam prikaže mnogo ikon vseh naloženih aplikacij. Da bi hitreje našli mojo aplikacijo, sem za ikono izbral sliko narisane hiše, pod njo pa se nahaja napis Avtomatizacija_hiše, kot prikazuje slika 15.



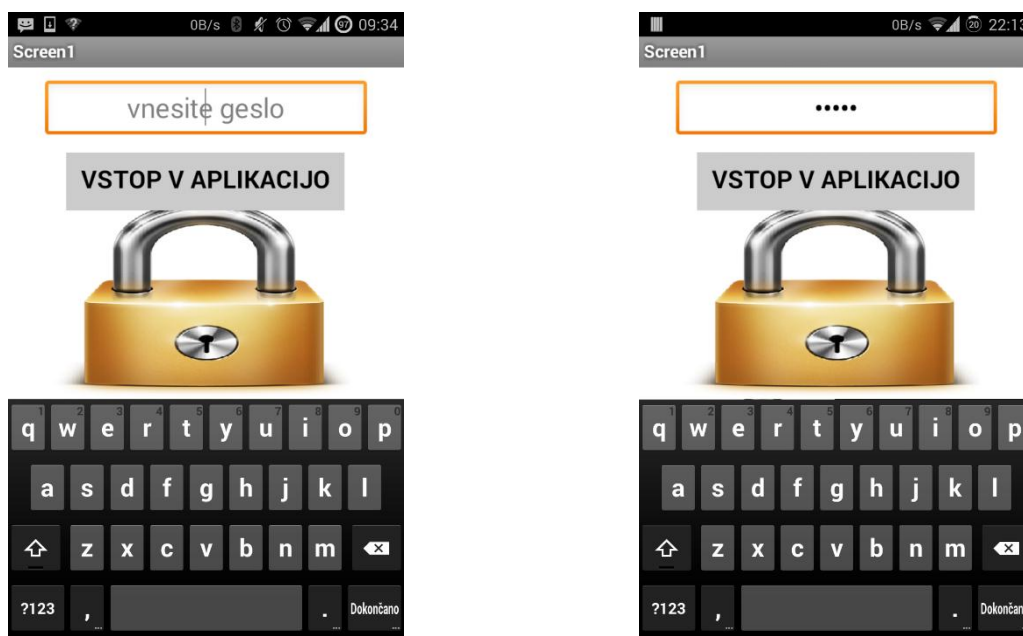
Slika 15: Ikona aplikacije Avtomatizacija_hiše

4.8.2 Kodiran vstop v aplikacijo

Aplikacija se zažene v trenutku, ko pritisnemo na ikono le-te.

Vhod v glavni meni, v katerem lahko vklopljamo luči, ... je možen le preko predhodnega vpisa gesla, katerega si po želji izberemo. To nam omogoča, da lahko naprave v hiši preko

telefona upravljajo le tisti, kateri poznajo geslo. Operacijski sistem Android omogoča, da lahko eno aplikacijo naložijo prav vsi, ki imajo telefone z enakim operacijskim sistemom, kar pomeni, da lahko imajo vsi nameščeno aplikacijo Avtomatizacija_hiše. Ampak v glavni meni lahko vstopijo le tisti, ki poznajo geslo.



Slika 16: Vnos gesla za vstop v aplikacijo

Ob pritisku ikone se nam prikaže polje, na katerem piše vnos gesla. Ob pritisku na polje se pojavi tipkovnica. Vpišemo izbrano geslo, nato pritisnemo tipko VSTOP V APLIKACIJO. V kolikor je geslo pravilno, se prikaže glavni meni, če pa geslo ni pravilno, nas aplikacija enostavno ne spusti v glavni meni. Kot prikazuje druga slika zgoraj, se namesto črk pri vnosu gesla prikažejo pike, da ostali v bližini ne vidijo gesla.

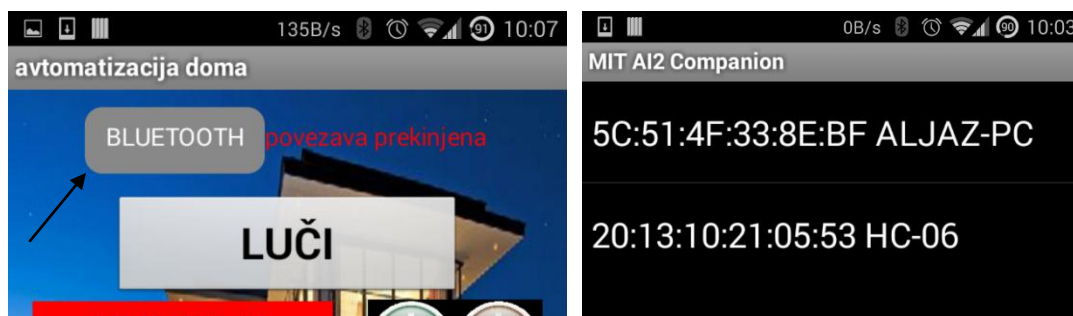
4.8.3 Meni za upravljanje hiše

Po vnosu pravilnega gesla, vstopimo v glavni meni aplikacije.

4.8.4 Povezava z bluetooth modulom

Za praktično uporabo aplikacije moramo najprej vzpostaviti povezavo z bluetooth modulom. To storimo tako, da na vrhu glavnega menija pritisnemo tipko z napisom bluetooth. Zraven tipke se nahaja še napis, ki nam pove ali je povezava vzpostavljena, ali ne.

Ob pritisku tipke se nam prikažejo vse bluetooth naprave, ki so nam na razpolago in jih imamo shranjene v telefonu. V mojem primeru izberem bluetooth modul z oznako HC-06. Vzpostavljanje povezave traja približno 3 sekunde. Če je povezava pravilno vzpostavljena, nas aplikacija samodejno pošlje v glavni meni, poleg bluetooth tipke pa se prikaže zelen napis.



Slika 17: Vzpostavljanje povezave z bluetooth modulom

4.8.5 Vklp in izklop luči

V glavnem meniju pod tipko bluetooth najprej zagledamo napis luči. Pod njim se nahajajo navedeni prostori, v katerih imamo luči, povezane z Arduino mikrokontrolerjem. V mojem primeru lahko upravljamo luči v kuhinji, dnevni sobi, spalnici, kopalnici in zunanje luči. Število prostorov je neomejeno in jih lahko po želji dodamo v programiranju App Inventorja.

Luči vklopimo in izklopimo s pritiskom tipk za vklop in izklop, kateri se nahajata na desni strani navedenega prostora.



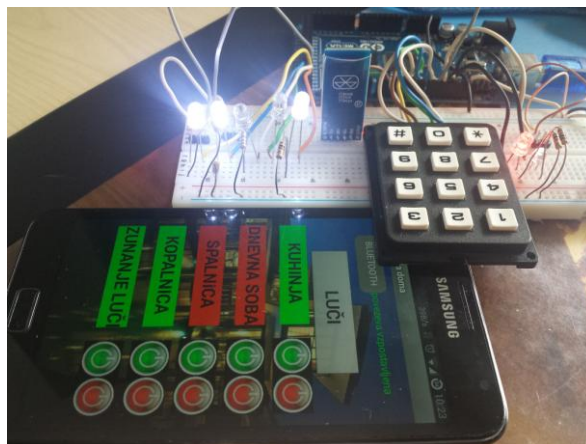
Slika 19: Tipki za vklop in izklop



Slika 18: Meni za luči

Napisi prostorov, v katerih vklopimo in izklopimo luči, so obrobljeni z rdečo barvo, v kolikor so luči v tistem prostoru izklopljene. Kadar luč v posameznem prostoru vklopimo, se le ta

obrobi z zeleno barvo. Tako lahko nadziramo, ali je luč vklopljena ali ne, ne glede na to, v katerem prostoru smo.



Slika 20: Primer vklopa treh od petih luči

4.8.6 Krmiljenje servomotorjev

4.8.6.1 Rolete

Pod navedenim zadnjim prostorom, v glavnem meniju, v katerem lahko vklopimo in izklopimo luči, se nahaja napis rolete. Pod njim se enko kot pri lučeh nahajajo navedeni prostori, v katerih imamo nameščene servomotorje za spust in dvig rolet, povezane z Arduino mikrokontrolerjem. Sam sem v aplikacijo podal le 2 prostora, spalnico in kopalnico, zaradi lažje izdelave makete.

Posamezno roletu dvignemo s pritiskom na tipko osvetlitev, ki se nahaja na desni strani navedenega prostora, v tem primeru se napis prostora obarva belo. Kadar pa želimo roletu spustiti, pa pritisnemo tipko zatemnitev in napis prostora, katerega smo zatemnili se obarva črno.



Slika 21: Meni za rolete

4.8.7 Garažna vrata

Popolnoma enako, kot deluje krmiljenje rolet, velja tudi za garažna vrata. V glavnem meniju pod zadnjim navedenim prostorom za dvig oz. spust rolet, se nahaja napis garažna vrata.

Le ta odpremo s pritiskom na tipko odpri, zapremo pa jih s pritiskom na tipko zapri. Napis garažna vrata se ob pritisku tipke odpri obarva belo, ob pritisku tipke zapri pa črno.



Slika 22: Meni za garažna vrata

5 Sklep

Z uspešno zaključeno inovacijsko nalogo sem zelo zadovoljen, predvsem zaradi tega, ker je za mene do sedaj predstavljala največji in najtežji izziv. Veliko ur mi je vzelo predvsem vizualno programiranje aplikacije, saj sem želel aplikacijo narediti čim bolj uporabniku prijazno. Aplikacijo mi je uspelo narediti tako enostavno, da mislim, da bi se jo lahko naučil upravljati prav vsak, kljub temu, da nebi imel znanja iz področja tehnologije. Programiranje v programskem okolju Arduino mi ni predstavljala večjih težav, saj zanj obstaja veliko že narejenih programov, le prirediti sem jih moral svojim potrebam.

6 Viri

<http://www.instructables.com/id/How-control-arduino-board-using-an-android-phone-a/> (januar, 2014)

<http://www.zmaga.com/content.php?id=4394> (januar, 2014)

<http://www.arduino.cc/> (januar, 2014)

<http://tehnika.telekom.si/aplikacije/google-app-inventor-lego-kocke-za-mobilne-telefone-android> (januar, 2014)