Model pametne kuće

Pletikos, David

Undergraduate thesis / Završni rad

2017

Degree Grantor / Ustanova koja je dodijelila akademski / stručni stupanj: **Polytechnic Pula - College of Applied Sciences / Politehnika Pula - Visoka tehničko-poslovna škola s pravom javnosti**

Permanent link / Trajna poveznica: https://urn.nsk.hr/urn:nbn:hr:212:574395

Rights / Prava: In copyright / Zaštićeno autorskim pravom.

Download date / Datum preuzimanja: **2024-04-18**

Image not found or type unknown

Repository / Repozitorij:

Digital repository of Istrian University of applied sciences

zir.nsk.hr

Image not found or type unknown

Politehnika Pula
Visoka tehničko-poslovna škola

PULA

PREDDIPLOMSKI STRUČNI STUDIJ POLITEHNIKE

DAVID PLETIKOS

MODEL PAMETNE KUĆE

ZAVRŠNI RAD

PULA, 2017

Politehnika Pula
Visoka tehničko-poslovna škola
PULA

PREDDIPLOMSKI STRUČNI STUDIJ POLITEHNIKE

MODEL PAMETNE KUĆE

ZAVRŠNI RAD

Kolegij: Elektrotehnika

Nositelj: mr.sc Radovan Jokić

Student: David Pletikos

JMBAG: 0233006371

PULA, Rujan 2017.

ZAHVALA

Zahvaljujem se svome mentoru prof.mr.sc Radovanu Jokiću koji je svojim znanstvenim i stručnim savjetima pomogao u izradi ovog završnog rada.

Također, zahvaljujem se svim svojim prijateljima i prijateljicama, koji su uvijek bili uz mene i bez kojih cijeli ovaj tijek mog studiranja ne bi prošao tako lako i zabavno.

Konačno, zahvaljujem se svojoj obitelji na potpori i razumijevanju u periodu izrade završnog rada i općenito na potpori tijekom studiranja.

IZJAVA

Kojom izjavljujem da sam završni rad s naslovom MODEL PAMETNE KUĆE izradio samostalno pod voditeljstvom prof.mr.sc, Radovana Jokića, koristeći se literaturom i znanjem stečenim tijekom studiranja na Politehnici Pula - Visoko tehničko-poslovnoj školi. Spoznaje, stavove, zaključke, teorije i zakonitosti koje sam izravno ili neizravno koristio u ovome završnom radu, navedeni su na standardni način uporabom fusnota i bibliografskih jedinica. Rad je pisan u duhu hrvatskog jezika.

_____

Student: David Pletikos

SAŽETAK

U ovom će se radu govoriti o pametnoj kući s tehničkog aspekta, odnosno o upravljanju sustavom unutar objekta. Definirat će se mogućnosti koje nudi integrirani sustav automatizacije u kućanstvu, navesti će se protokoli koji se koriste. Prikazat će se primjena samog integriranog sustava automatizacije. Nadalje, u samu svrhu prikazivanja funkcioniranja sustava automatizacije kućanstva, odnosno pametne kuće, izrađen je mali funkcionalni model pametne kuće i opisat će se način funkcioniranja samog modela, te navesti kako je moguće upravljati modelom.

SUMMARY

This work talks about the smart house from the tehnical perspective, system regulation inside the object. The possibilities which the integrated system of automation offers in households will be defined and the protocols which are used will be brought up. The application of the integrated automation circuit will be shown. Further in, for the purpose of showing the system of household automation, or to say the smart house, a small functional model of a smart house will be made and the way it works will be described, and the way of regulating the model will be shown.

SADRŽAJ

SAŽETAK 1

SUMMARY 1

1. UVOD 4

1.1. Cilj rada 4

1.2. Metode rada 4

1.3. Struktura rada 4

2. PAMETNA KUĆA (SMART HOUSE) 5

2.1. Što je to pametna kuća? 6

2.2. Sustavi upravljanja i nadzora 8

2.3. Komunikacijski protokoli u automatizaciji kućanstva 11

2.3.1. X – 10 12

2.3.2. LonWorks 14

2.3.3. CEBus 16

2.3.4. Konnex 17

2.4. Primjena pametne kuće 18

2.5. Ušteda energije pametne kuće 20

3. MODEL PAMETNE KUĆE 21

3.1. Potrebne komponente 22

3.1.1. Arduino UNO mikrokontroler 22

3.1.2. Bluetooth HC 06 modul 24

3.1.3. Android uređaj 26

3.1.4. Ultrazvučni modul 27

3.1.5. PIR senzor 29

3.1.6. Servo motor 32

3.1.7. Senzor temperature DHT 11 34

3.1.8. Ostalo 36
3.2. Konstrukcija i izrada modela pametne kuće 40
3.3. Izrada programa za upravljanje Arduino uređajem 52
3.4. Izrada aplikacije za upravljanje pomoću Android uređaja 59
3.4.1. Izrada dizajna aplikacije 61
3.4.2. Izrada logičkog dijela aplikacije 67
3.5. Opis rada modela pametne kuće 74
3.5.1. Automatski način rada 74
3.5.2. Ručni način rada 75
3.6. Financijsko izvješće izrade modela pametne kuće 76
4. ZAKLJUČAK 78
LITERATURA 79
POPIS SLIKA 82
POPIS TABLICA 84
POPIS PRILOGA 85

1. UVOD

Pametna kuća je inteligentna građevina pod kojom se podrazumijeva integracija raznih uređaja, najčešće električnih uređaja za automatsko upravljanje nad kućom. Pametne kuće su budućnost građevinarskog zanata zbog brzog i opširnog razvoja novih tehnologija, koji će omogućiti potpunu automatizaciju upravljanja samom kućom i tako doprinijeti zajednici ljudi potpuno zadovoljstvo u domaćinstvu.

1.1. Cilj rada

Cilj rada je na cjelovit način prikazati, što su to pametne kuće, kako se koriste i kolika je zastupljenost u građevinarstvu i stanogradnji. Rad sustava pametne kuće zorno je prikazan na konkretnom modelu izvedbom pomoću Arduino mikrokontrolera.

1.2. Metode rada

Metode koje su korištene u radu su sljedeće:

Kod definiranja cilja, svrhe, te pri opisu izvođenja svakog od postupka koristili smo opisnu metodu. Kod izrade tablica koristili smo grafičku metodu, te kod izračunavanja financijske analize matematička metoda.

1.3. Struktura rada

Rad se sastoji od četiri glavna poglavlja.

U uvodnom dijelu su sažeto navedeni ciljevi rada, te korištene metode.

Drugo poglavlje daje definiciju i opis pametne kuće, te prednosti i uštede koje se postižu u odnosu na klasičnu kuću. Također su pobrojani komunikacijski protokoli potrebni za komunikaciju među sustavima pametne kuće.

Treće poglavlje detaljno opisuje izrađeni model pametne kuće. Korak po korak opisana je izrada modela, obuhvaćeni sustavi, specifikacije uporabljenih elemenata, te opis funkcioniranja. Na koncu je naveden i troškovnik nabavljenih komponenata.

Zadnje poglavlje je Zaključak. Nakon zaključka navedena je literatura korištena u svrhu izrade rada, prikazan je popis slika i tablica, te na samom kraju su priložena dva priloga u popisu priloga.

2. PAMETNA KUĆA (SMART HOUSE)

Pojam pametne kuće (ili pametne zgrade) nastao je sedamdesetih godina prošlog stoljeća. Tada se pametnom kućom smatrala ona koja je bila sagrađena imajući na umu energetsku učinkovitost. Pojam pametne kuće od tada se mijenjao prateći razvoj tehnologije na tom području. Tako je tijekom osamdesetih godina pametna kuća bila ona u kojoj se s osobnog računala moglo upravljati raznim sustavima. Današnje poimanje pametne kuće predstavlja nadogradnju definicija iz prošlosti dodatkom podsustava za gospodarenje energijom i utjecaj na okoliš te primjenom raznih senzora i sustava za automatizaciju.[1] Slika 1. nam predstavlja sam koncept pametne kuće.

Slika 1. – Koncept pametne kuće

Izvor: http://www.hyperion.rs/images/fancybox/SmartHome.jpg (29.08.2017.)

[1]Laboratorij za sustave i signale: Sigurnost pametnih kuće, Fakultet elektrotehnike i računarstva, Sveučilište u Zagrebu, Travanj 2012.

2.1. Što je to pametna kuća?

Pametna kuća se može opisati kao jedan sustav kod kojega se podrazumijeva integracija raznih električnih uređaja u domaćinstvu u jedinstveni sustav kojim se upravlja na različite načine. Ovisno o životnom stilu, navikama ili potrebama korisnika sustavom pametne kuće se može upravljati na više načina.[2]

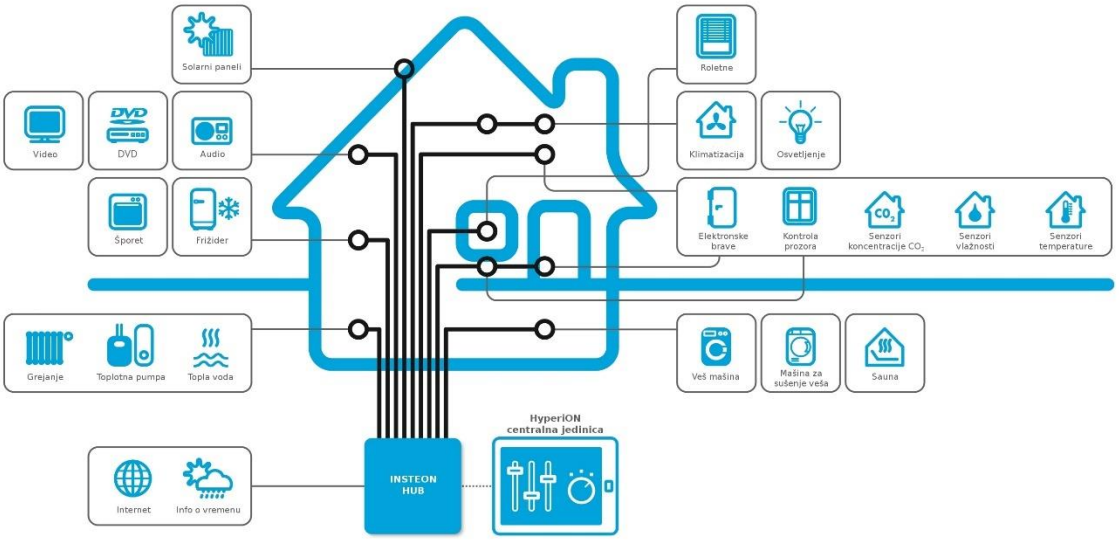
Pametna kuća je objekt nad kojom imamo apsolutnu mogućnost upravljanja, podešavanja, prilagođavanja i kontrole prema specifičnim željama i potrebama korisnika. Sustav pametne kuće sagledava kuću na isti način kao i kod klasičnih električnih instalacija, ali pomoću sustava pametne kuće se može vrlo jednostavno upravljati baš svime, npr. grijanjem, klimatizacijom, multimedijom, zastorima, itd.. Instalirani sustav pametne kuće omogućava jednostavno upravljanje svim elementima sustava, koristeći iste sklopke kao i kod obične električne instalacije. Osim sklopkama raspoređenim po kući sustavom se može upravljati pomoću: LCD ekrana osjetljivih na dodir, računala, mobilnim telefonima, IR daljinskim upravljačima.[3] Kao što je vidljivo na slici 2. postoji više mogućnosti upravljanja pametnom kućom.

Slika 2. – Upravljačka jedinica pametne kuće

Izvor: http://pimami.hr/wp-content/uploads/2014/05/image7_hclite_eng.png (29.08.2017.)

[2] https://bs.wikipedia.org/wiki/Sistem_ku%C4%87ne_automatike (16.08.2017.)
[3] https://www.veleri.hr/?q=node/1133 (31.08.2017.)

Sustavi pametne kuće (slika 3.) integriraju sustav grijanja, hlađenja, protuprovalni i protupožarni sustav, kućnu rasvjetu, audio/video sustav u jedinstvenu cjelinu koja omogućuje optimalno korištenje energije i komforno upravljanje kućom. Cijeli sustav je temeljen na centralnom procesorskom sustavu koji kontrolira i upravlja svim ostalim kućnim sustavima. Centralni procesor dobiva informacije iz vanjskog svijeta putem različitih senzora temperature (prostorije, vanjski prostor), svjetlosti, mehaničkih senzora otvorenosti/zatvorenosti vrata i prozora, senzora vlage (kupaonica), senzora detekcije pokreta, vatre, dime i dr. Dobivene informacije se obrađuju u procesoru i na temelju definiranih pravila, procesor upravlja ostalim uređajima, svjetlom, klima uređajima, sustavom za grijanje, automatskim sustavima prozora, vrata, ventilima za vodu, plin, jednom riječju svim što postoji u kući. Svi uređaji su međusobno povezani u centralnu mrežu kako bi se mogli nadzirati i upravljati s različitih pozicija. Mreža može biti izvedena pomoću žičanih kabela, bežično ili korištenjem postojeće strujne mreže. Pravila procesiranja iako unaprijed definirana, mogu se programirati prema zahtjevima korisnika, uz istovremenu fleksibilnost i jednostavnost upravljanja sustavom.[4]



Slika 3. – Integrirani sustav pametne kuće

Izvor: http://www.hyperion.rs/images/fancybox/HyperiON_Concept.jpg (29.08.2017.)

[4] http://www.eko.zagreb.hr/default.aspx?id=109 (16.08.2017.)

2.2. Sustavi upravljanja i nadzora

- UPRAVLJANJE RASVJETNIM TIJELIMA

Pametna kuću omogućuje upravljanje rasvjetnim tijelima pomoću upravljača ili kontrolne ploče. Umjesto klasičnih prekidača koji se koriste za uključivanje pojedinih izvora svijetla, upravljanje rasvjetom unutar pametne kuće omogućuje definiranje svjetlosne scene za različite prigode, također je moguće podesiti različite intenzitete svijetla u bilo kojoj prostoriji, te kombiniranje rasvjete s više izvora. Rasvjetom se upravlja s jednog mjesta, bilo to daljinskim upravljačem ili nekim od zidnih prekidača. Pametna kuća nudi mogućnost gašenja sve rasvjete jednim jedinim prekidačem.[5]

- UPRAVLJANJE KLIMATIZACIJOM

Integrirano grijanje i klimatizacija u sustav pametne kuće daje mogućnost kontrole temperature po prostorijama, te napredne funkcije programiranja pomoću upotrebe nekih od senzora kao što su : senzor temperature, senzor osvjetljenja, senzor pokreta, itd.[6]

Mogućnost koju nam nudi upravljanje klimatizacijom i grijanjem je ta da se upravlja grijanjem automatski i optimalno ovisno o namjeni prostorije ili potrebi korisnika. Pametna kuća nam radi uštede ne dopušta paljenje klimatizacije i grijanja ako su prozori otvoreni, te sami ventilacijski sustav može reagirati na prisutnost ljudi u prostoriji.[7]

Sustavi pametne kuće povećat će učinkovitost i ekonomičnost grijanja i hlađenja prostorija, te smanjiti potrošnju električne energije. Postojeći sustav centralnog grijanja ili sustav klimatizacije može se integrirati pod nadzor pametne kuće. Vanjski temperaturni senzor omogućit će optimalno grijanje i hlađenje unutarnjeg prostora u skladu s godišnjim dobom i vanjskom temperaturom čime se izbjegavaju temperaturna nadvišena i smanjuje potrošnja.[8]

[5] http://www.eko.zagreb.hr/default.aspx?id=109 (16.08.2017.)
[6] http://www.pametna-kuca.ba/index.php/pametna-kuca/mogucnosti-sistema (16.08.2017.)
[7] http://www.knx.org/hr/knx/stambene-zgrade/ (16.08.2017.)
[8] http://www.eko.zagreb.hr/default.aspx?id=109 (16.08.2017.)

- SIGURNOST KUĆE

Protuprovalna zaštita uz standardne alarmne omogućava i napredne funkcije upravljanja kućnom automatikom, kao što je mogućnost zoniranja prostora pa u slučaju detekcije obavještava korisnika pomoću telefonske dojave ili zvučnog signala alarmom. Sustav protuprovalne zaštite nudi korisniku simulaciju prisustva dok je korisnik van kuće u namjeri protuprovalne prevencije. Upravljanje protuprovalnom zaštitom moguće je preko telefonskog upravljanja uređajima preko centralne jedinice, pomoću mini daljinskog upravljača, kontrola putem Android/iOS aplikacija, SMS dojava stanja, itd.[9]

Pametna kuća ima mogućnost da imitira stil života korisnika. Ako korisnik nije kod kuće, sustav može imitirati korisnikove zahtjeve za zaštitu. Npr. uključivati i isključivati vanjsku i unutarnju rasvjetu, podizati i spuštati rolete, paliti i gasiti multimediju, što je dovoljno da odbije mogući napad provalnika. U slučaju provalne pametna kuća može iznenaditi provalnika s nekim od mogućnosti kao što su: uključivanje alarma, uključivanje cjelokupne rasvjete , podizanje roleta, te dojava o provali samom korisniku. U slučaju nepogoda kao što su poplava ili požar pametna kuća isključuje glavni ventil u slučaju poplave, te glavno napajanje strujom čime će spriječiti moguć nastanak štete. Sličan scenarij se može dogoditi i kod požara.[10]

- UPRAVLJANJE MULTIMEDIJOM I KUĆANSKIM APARATIMA

Kod upravljanja multimedijom i kućanskim aparatima najčešće se u automatizaciju integriraju sustavi ozvučenja i video sustavi, a od kućanskih uređaja bojler, toster, mašina za pranje suđa, rolete i zastori s elektromotorom, garažna vrata, audio i video uređaji, caffe aparat.[11]

Najčešće za komfor u pametnoj kući je integracija audio/video sustava koji korisnikove želje i potrebe za opuštanjem čini još ugodnijima. Odabirom filma ili glazbe koju korisnik želi slušati ili gledati, pametna kuća prilagođava ugođaj i atmosferu u prostoriji. A to je najčešće prigušivanje svijetla, spuštanje platna i projektora, spuštanje zastora. Postoji i mogućnost u tim situacijama da prema želji korisnika sustav onemogući zvonjavu na vratima ili isključi telefone zbog uznemiravanja.[12]

[9] http://www.pametna-kuca.ba/index.php/pametna-kuca/mogucnosti-sistema (16.08.2017.)
[10] http://www.eko.zagreb.hr/default.aspx?id=109 (16.08.2017.)
[11] http://www.pametna-kuca.ba/index.php/pametna-kuca/mogucnosti-sistema (16.08.2017.)
[12] http://www.eko.zagreb.hr/default.aspx?id=109 (16.08.2017.)

- POMOĆ INVALIDIMA

Ljudima s problemima kretanja, pametna kuća omogućuje bolju autonomiju i olakšava kontrolu kućnih sustava, rasvjete, grijanja, otvaranja ulaznih vrata, kontrole ulaza u zgradu i dr. Prema željama i potrebama korisnika, može se projektirati sustav koji će omogućiti maksimalnu slobodu i samostalnost u korisnikovu domu..[13]

- MOGUĆNOST IP VIDEO NADZORA

IP video nadzor prostorija i područja od interesa putem računala lokalno i preko interneta. Pametne kuće pomoću IP video nadzora imaju mogućnost cjelokupnog nadzora kamerama u što ulaze mogućnosti: pomicanje, zumiranje, sigurnosni pristup, senzor pokreta, vidljivost u malim uvjetima svijetla, audio, više rezolucijski način, e-mail notifikacije.

- MOGUĆNOST PROGRAMIRANJA

Mogućnost neograničenog programiranja i kreiranja događaja (primjeri događaja: gledanje filma, party, romantična večera, pokreni alarmni način rada, odlazak od kuće-simulacija prisustva, dobro jutro, laku noć, štedi energiju...). Osim ovoga, moguće je jednostavno i brzo aktivirati po želji definirati programski scenarij kojim se stambeni prostor prilagođava trenutnoj aktivnosti, raspoloženju ili potrebi svojih stanara. Primjeri scenarija mogu biti: gledanje kućnog kina, posjeta prijatelja, odlazak na posao, ručak, godišnji odmor itd.[14]

Programiranjem pametne kuće može se maksimalno prilagoditi svoju okolinu svojem životu: Uključivanje intenzivnog grijanja ili hlađenja kuće neposredno prije ustajanja. Tijekom dana korisnik ima mogućnost dok nije u kući da podesi optimalno grijanje ili hlađenje ovisno o potrebi. Moguće je SMS porukom aktivirati klimatizaciju na željenu temperaturu. Mogućnost definiranja moda rada grijanja ili hlađenja tijekom dana i noći te tijekom tjedna i vikendom, podešavanje temperature za svaku prostoriju posebno.

[13] http://www.eko.zagreb.hr/default.aspx?id=109 (16.08.2017.)
[14] http://www.pametna-kuca.ba/index.php/pametna-kuca/mogucnosti-sistema (16.08.2017.)

2.3. Komunikacijski protokoli u automatizaciji kućanstva

Kao i u ljudskoj komunikaciji, različiti mrežni i računalni protokoli moraju međusobno funkcionirati kako bi mrežna komunikacija bila uspješna. Stoga protokol možemo definirati kao skup pravila i formata koji određuju komunikacijske odnose između dva uređaja radi točne razmjene poruka, u ovom slučaju signala.[15]

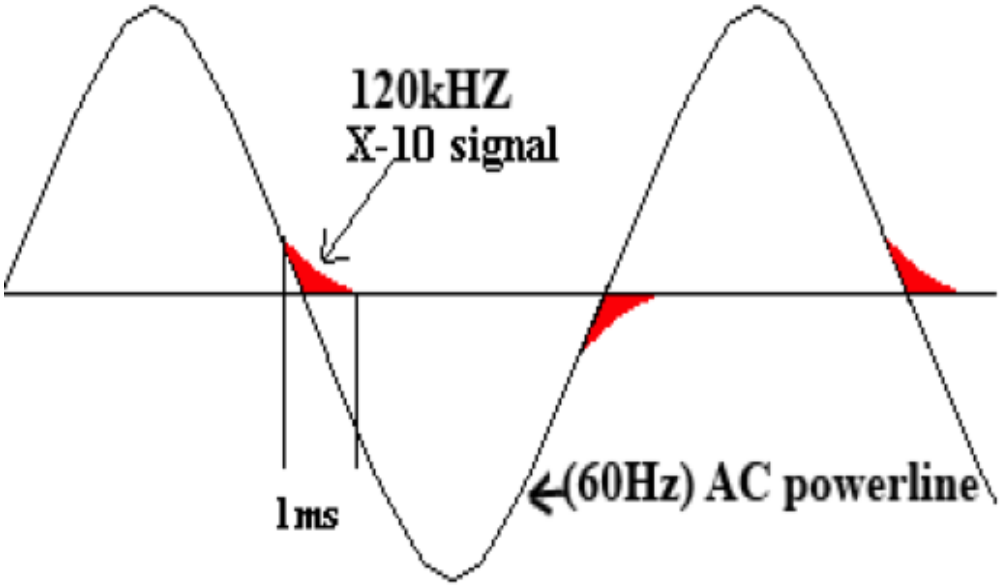
Kod automatizacije za pametnu kuću najčešće se koriste otvoreni sustavi (engl. open source systems) sa poznatom strukturom i komunikacijskim protokolom. Oni omogućuju laku zamjenljivost uređaja bez degradacije sustava. U povijesti kućne automatizacije kristalizirala su se dva glavna standarda: europski EIB(engl. European Installation Bus) i američki LonWorks. Devedesetih godina 20. stoljeća, uz EIB, u Europi se još koriste European Home Systems i BatiBUSstandardi. Oni se zajedno sa standardom EIB stapaju u jedinstveni europski standard Konnex KNX. Jedan od najstarijih protokola u kućnoj automatizaciji koji primarno koristi elektroenergetsku mrežu kao medij jest X10 standard. Potrebno je naglasiti da oni protokoli koji su strukturno sabirnički mogu koristiti kao medij i gradsku mrežu i radio medij. No, ipak ćemo podrazumijevati radio vezu načinom spajanja bežičnih standarda. Većina standarda u kućnoj i zgradarskoj automatizaciji se bazira na ISO/OSI modelu sa integriranim korisničko usmjerenim slojevima tj. na ISO/OSI modelu sa 5 formalnih slojeva.[16]

[15] Matas, K. : Mrežni protokoli i norme, Računalne mreže, Materijali sa predavanja
[16] Matković, P. :Norme i protokoli za automatizaciju kućanstva, Završni rad, Lipanj 2013., Zagreb

2.3.1. X – 10

X- 10 protokol je izumljen 1975. godine od strane tvrtke Pico Electronics u Škotskoj. Ime X-10 proizlazi iz činjenice da je ovaj protokol za kućnu automatizaciju bio deseti projekt tvrtke Pico Electronics. 1978. godine počinju se pojavljivati prve X-10 komponente u trgovinama Radio Shack. On je bio vrlo veliki trgovac sa elektroničkom opremom u Sjedinjenim Američkim Državama koje su pomogle da tehnologija X-10 bude poznata. X-10 protokol rađen je za BSR (British Sound Reproduction) s ciljem omogućavanja bez žične metode daljinskog upravljanja njihovih komponenti.[17]

Princip rada je jednostavan, dakle imamo odašiljače i prijemnike. Naredba odnosno informacija koju odašiljemo, frekvencije 120kHz, sastoji se od osam bitova (jedan prolazak sinusoide kroz nulu predstavlja jedan bit) i ukomponirana je u val nosilac, sinus 50 Hz gradske mreže što je prikazano na slici 4. Iz ovoga slijedi da možemo adresirati maksimalno 256 uređaja koji mogu raditi neovisno jedan o drugom, a i više njih u slučaju da nekima pridijelimo iste adrese ako nam odgovara da se neke radnje izvode paralelno. [18]

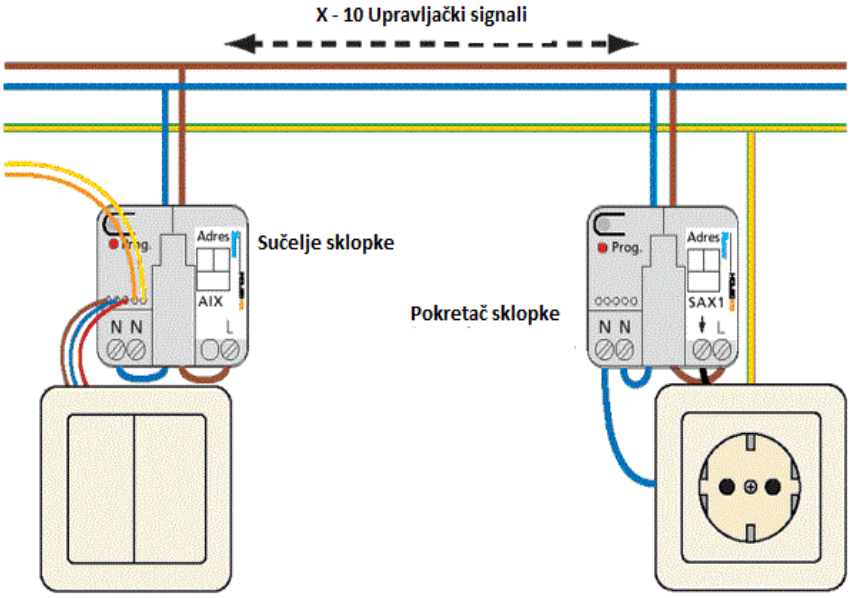


Slika 4. – Signal koji šalje X-10 protokol

Izvor: http://www.ee.ic.ac.uk/leon.zhang11/yr2proj/img/sinewave.png (29.08.2017.)

[17] http://home.planet.nl/~lhendrix/x10_history.htm (29.08.2017.)
[18] Jakovčić, A.; Marić, M. :Pametna kuća, Seminarski rad, Svibanj 2007., Zagreb

Prednosti ovog protokola su jeftina implementacija zbog korištenja postojećih instalacija, jednostavnost same instalacije te tridesetogodišnja dominacija na tržištu. Problem predstavljaju kućanski aparati sa elektromotorom koji unose smetnju (šum) u napon gradske mreže pa na prijemniku može doći do pogrešnih detekcija. Drugi problem su uređaji poput računala koji rade sa posebno ugođenim naponima napajanja podešenim pred filtrima koji imaju tendenciju iz filtrirati upravljački X-10 signal. Ovi problemi se rješavaju raznim filtrima za šum. Problemi vezani uz fazne pomake rješavaju se pomoću couplera. Slika 5. prikazuje kako se povezuje X-10 protokolom.[19]



Slika 5. – Povezivanje sa X-10 protokolom

Izvor: https://www.bwired.nl/images/how/schema_Xanura.gif (29.08.2017.)

[19] Jakovčić, A.; Marić, M. :Pametna kuća, Seminarski rad, Svibanj 2007., Zagreb

2.3.2. LonWorks

LonWorks je platforma za umrežavanje stvorena posebno za to kako bi se zadovoljile potrebe aplikacija za upravljanje. Stvorio ju je Echelon Corporation za umrežavanje uređaja preko medija kao što su upletene parice, električni vodovi, optički kablovi. Koristi se za automatizaciju raznih funkcija u zgradama kao što su rasvjeta i HVAC.[20]

Jedna LonWorksov-a platforma sadrži:

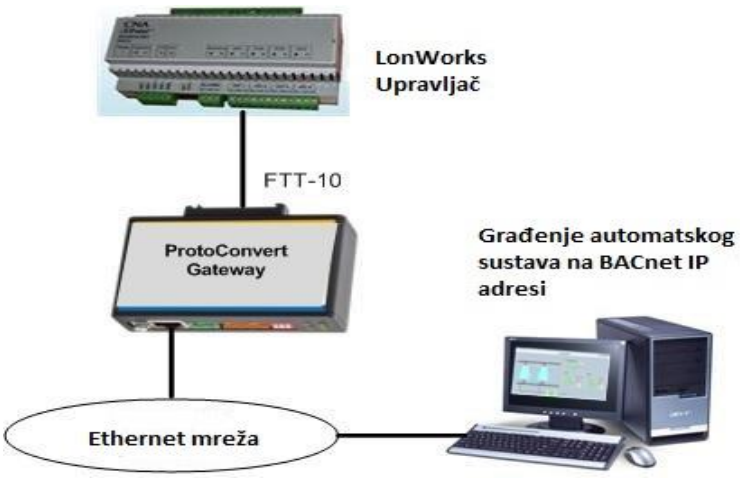
1.Neuron mikroprocesor

2.LONtalk komunikacijski protokol

3.LONworks Network Services (LNS)

Komunikacija između uređaja povezanih LonWorks mrežom je moguća preko medija koji može biti upredena parica, gradska mreža, RF1, IR2, koaksijalni kabel ili optička vlakna. Osnova LonWorks platforme je neuron mikroprocesor kojim upravlja program. Arhitektura protokola zasniva se na 'peer to peer' komunikaciji između uređaja što znači da dva uređaja komuniciraju između sebe direktno i sa jednakim ovlastima. Za razliku od komunikacije preko servera koji u slučaju prenatrpanosti postaje usko grlo sustava ovdje do sličnog problema ne može doći. Primjer serverom upravljane mreže i p2p mreže. Slika 6. prikazuje kako se LonWorks model spaja pomoću automatiziranog sustava na računalu preko Internet mreže na LonWorks – ov upravljač za upravljanje pametnom kućom.[21]

[20] https://en.wikipedia.org/wiki/LonWorks (29.08.2017.)
[21] Jakovčić, A.; Marić, M. :Pametna kuća, Seminarski rad, Svibanj 2007., Zagreb



Slika 6. – LonWorks model spajanja

Izvor:

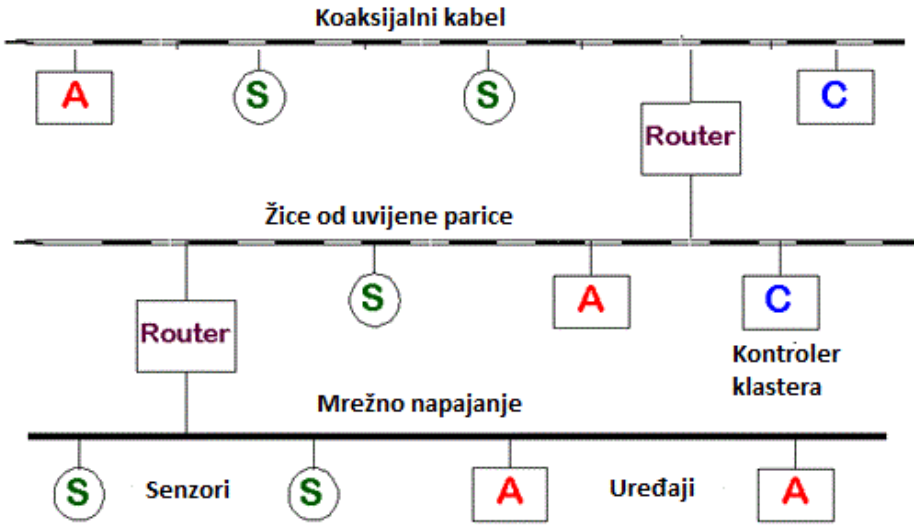
http://www.protoconvert.com/portals/0/DiagramsForIndividualPage/Lonworks%20to%20BACnet%20IP%20Scenario%201.jpg (29.08.2017.)

Brzina komunikacija između pojedinih uređaja varira ovisno o mediju i dostiže 1.25 Mbps na uparenim paricama. LonWorks sustavi mogu koristiti različite kanale na istim ili različitim tipovima medija. Kanali su međusobno povezani pomoću LonWorks usmjerivača. Neuron čipovi hardverski implementiraju slojeve. (engl. Open Systems InterConnection Basic Reference Model) referentnog modela, što olakšava razvoj novih aplikacija za upravljačke mreže koje koriste LONWorks tehnologiju. Programska implementacija protokola LONTalk je sadržana u ROM-u (engl. Read Only Memory) svakog neuron mikroprocesora, a parametri specifični za karakteristične aplikacije pohranjuju se u RAM(engl. Random Access Memory) rezerviran upravo za te parametre. LONWorks Network Services (LNS) je skup alata i programa za instalaciju, održavanje, nadgledanje i upravljanje Inter operabilnim LONWorks upravljačkim mrežama. Takvi alati omogućuju udaljeni nadzor i parametriranje upravljačke mreže, kao i konfiguraciju uređaja spojenih u mrežu, spuštanje aplikacijskih programa i parametara na uređaje spojene u mrežu, prijavu grešaka i sl. LNS također omogućuje integraciju sustava upravljanja sa ostalim informacijskim sustavima, pa danas LONWorks arhitektura podržava klijente većine operacijskih sustava. Domena je skup čvorova jednog ili više kanala. Komunikacija čvorova moguća je unutar jedne domene. Domenu je moguće podijeliti na 127 pod mreža od po 256 čvorova s time da jedan čvor može biti u više pod mreža samo ako je u svakoj definiran. Jednostavnim računom slijedi da je moguće povezati 33485 uređaja.[22]

[22] Jakovčić, A.; Marić, M. :Pametna kuća, Seminarski rad, Svibanj 2007., Zagreb

2.3.3. CEBus

CEBus (engl. Consumer Electronics Bus) je skup električnih standarda i komunikacijskih protokola za elektroničke uređaje za prijenos naredbi i podataka. Pogodan je za uređaje u kućanstvima i uredima, a mogu biti korisni i za korisničko sučelje i industrijske aplikacije. Stvoren je u suradnji članova društva EIA (engl. Electronic Industries Alliance) kao protokol s više mogućnosti i boljim karakteristikama od tada jedinog postojećeg X-10 protokola. Komunikacija između uređaja je ostvarena na p2p modelu. Mediji korišteni pri prijenosu informacija su gradska mreža, upredena parica, koaksijalni kablovi, IR, RF, optička vlakna i audio/video sabirnice što nam prikazuje slika 7.[23]



Slika 7. – Primjer povezivanja sa CEBus protokolom

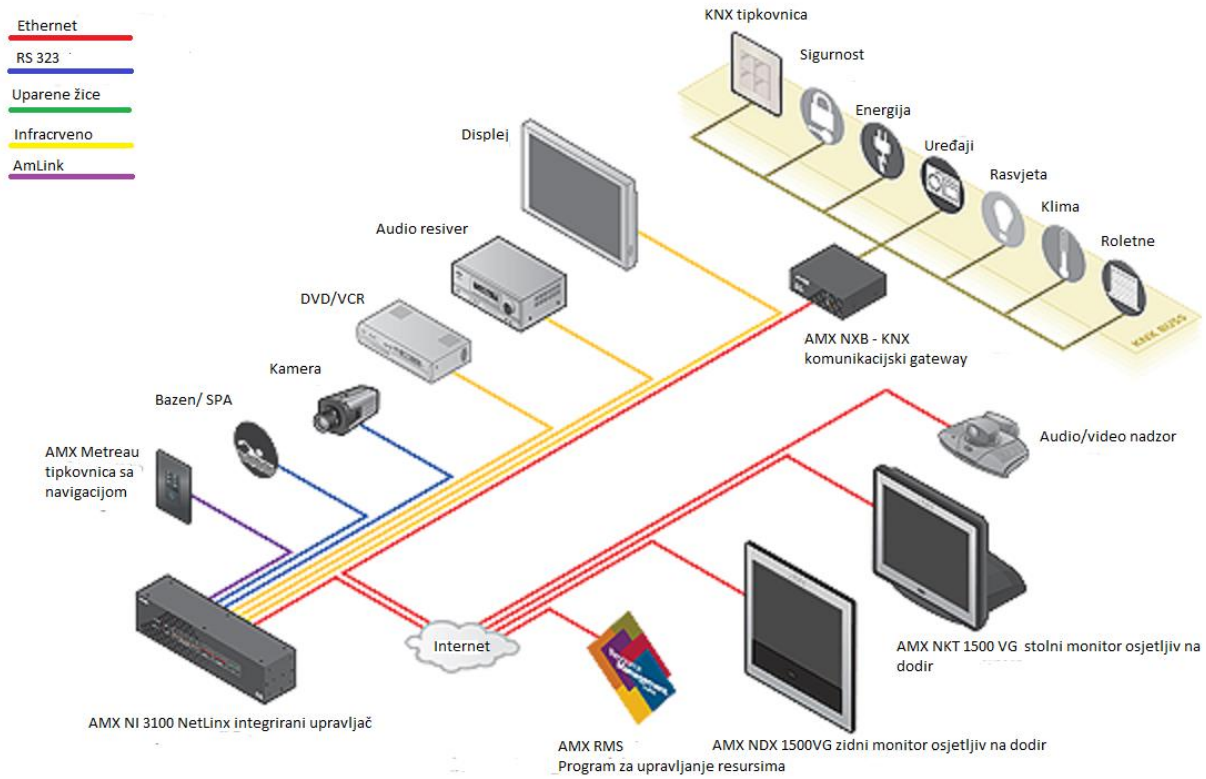
Izvor: http://www.hometoys.com/htinews/aug97/articles/kwacks//img02.gif (29.08.2017.)

Najčešće korišten medij je gradska mreža. Prijenos bursta počinje na 100kHz i kroz 100ms linearno raste do 400kHz. Zbog stalne promjene frekvencije nosioca informacije mala je vjerojatnost da se ona iz filtrira nekim od filtara ugrađenim u uređaje. Brzina prijenosa podatka je oko 10000bps. Nedostaci CEBus protokola su relativno malen broj kompatibilnih proizvoda dostupnih na tržištu i njihova visoka cijena.[24]

[23] https://en.wikipedia.org/wiki/CEBus (29.08.2017.)
[24] Jakovčić, A.; Marić, M. :Pametna kuća, Seminarski rad, Svibanj 2007., Zagreb

2.3.4. Konnex

Konnex ili skraćeno KNX je standardizirani protokol za upravljanje zgradom ili u upotrebi za kućnu automatizaciju. Njegova najveća prednost je neovisnost od bilo koje posebne hardverske platforme što znači da se može izvesti i na 8- bitnom mikrokontroleru. Tehnologija se prvenstveno fokusira na sigurnost i uštedu energije. Zato nadzor svih različitih komponenti u KNX sustavu igra važnu ulogu. Ako se klima uređajem, grijanjem i osvjetljenjem upravlja jednim sustavom, nastaju višestruke mogućnosti rezanja troškova. KNX se temelji na tri tehnologije: EIB (European Installation Bus), EHS (European Home System Protocol) i BatiBus. Stoga je izgrađen na dobro utvrđenom temelju. Štoviše, to omogućuje razne načine komunikacije uključujući upletene parice, radio frekvenciju, infracrvenu mrežu, mrežu za napajanje i IP/Ethernet (slika 8.). U KNX sustavu automatizacije nema potrebe za centraliziranim kontrolnim centrima, budući da svi uređaji komuniciraju istim jezikom i preko sabirnice.[25]



Slika 8. – Povezivanje pametne kuće sa Konnex protokolom

Izvor: http://trade.amx.com/imgs/diagrams/NXB-KNX.Diagram.png (29.08.2017.)

[25] http://buildyoursmarthome.co/home-automation/protocols/knx/ (29.08.2017.)

Svi uređaji u KNX instalaciji su povezani dvostrukom sabirnicom (najčešći oblik instalacije), čime se omogućuje razmjena podataka. Funkcija pojedinačnih sabirničkih uređaja određena je projektiranjem, koje se može mijenjati i prilagoditi u bilo kojem trenutku. Jedan KNX sustav se sastoji od senzora koji prikupljaju informacije i šalju ih preko sabirnica. Aktuatori su oni koji primaju informacije sa senzora te primljene podatke pretvaraju u akcije, odnosno uključuju rolete, osvjetljenje ili grijanje i hlađenje.[26]

2.4. Primjena pametne kuće

Sustavi pametne kuće svojom udobnošću, povećanjem sigurnosti i ekonomičnosti podosta privlače svoje korisnike. Uređaji koji se koriste se dijele u više podsustava koji rade ili samostalno ili zajedno sačinjavaju sustav pametne kuće. Ovi podsustavi su najčešće oni za upravljanje: rasvjetom, klimatizacijom i grijanjem, kućanskim aparatima, tendama i zavjesama, roletama, garažnim i ulaznim vratima, sigurnosnim sustavima (protupožarni, protuprovalni i sustavi protiv poplava), te i sa multimedijom. Kod ugradnje sustava pametne kuće najviše se koristi protokol X-10 jer je najjednostavniji za primjenu i implementaciju. Standardna oprema sustava pametne kuće je najčešće sačinjena od LCD monitora osjetljivog na dodir, mikro modula, senzora, tipkala i daljinskog upravljača. Centralna konzola za upravljanje pametnom kućom je LCD monitor osjetljiv na dodir. LCD monitor osjetljiv na dodir ima u sebi ugrađen modem kojim se omogućuje komuniciranje sa telefonskom centralom koja pruža mogućnost upravljanja preko mobilnog telefona. Centralna konzola sadrži i primopredajnik koji prima radio frekventne signale sa daljinskog upravljača te ih pretvara u X-10 upravljačke signale. Pomoću te centralne konzole je spojena sva kućna automatika i sigurnosni sustav. Ako se ne upravlja pomoću mobilnog uređaja ili daljinskog upravljača, zahtjeve možemo davati preko monitora osjetljivog na dodir naredbama koje sami definiramo. Centralne konzole su napravljene tako da u sebi imaju programiran software koji korisniku omogućuje jednostavno programiranje naredbi, programiranje scenarija, postavljanje ikona za određene naredbe. Na samu centralnu konzolu spajaju se i sigurnosne kamere, koje se postavljaju na korisnikov zahtjev u određene prostorije, te moguće je spojiti 30 bežičnih i nekoliko žičnih senzora. Senzori se koriste zbog umanjenja mogućnosti od nastanka poplava, požara i provala. Za

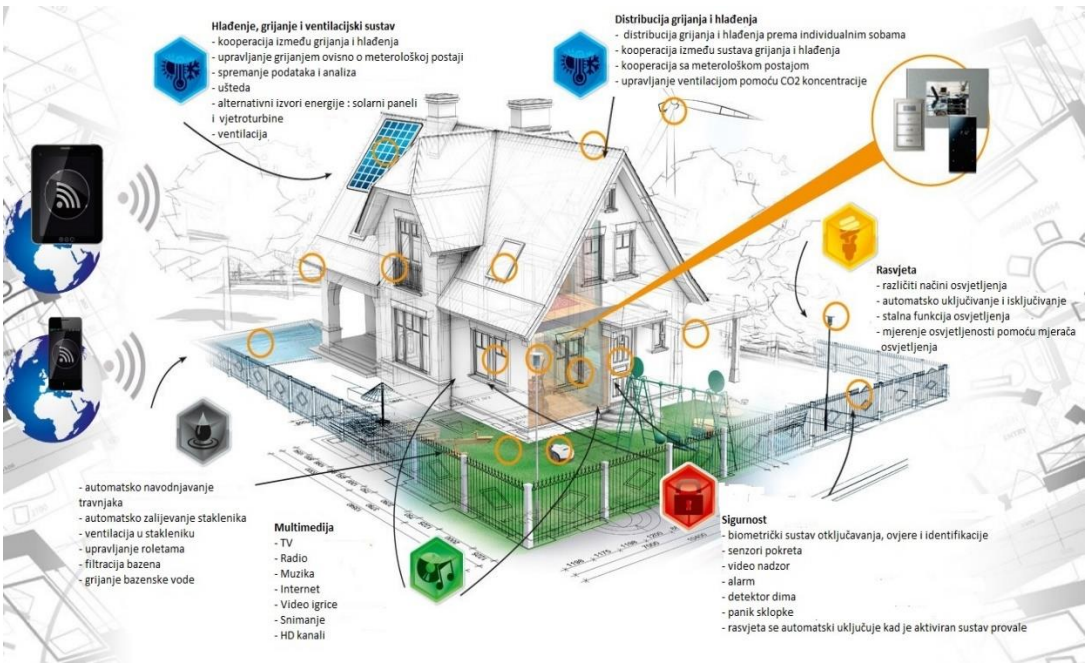
[26] https://en.wikipedia.org/wiki/KNX_(standard) (29.08.2017.)

paljenje i gašenje svijetla pri ulasku ili izlasku iz prostorije koriste se senzori za detekciju pokreta. Jedan od najvažnijih uređaja za upravljanje je sam daljinski upravljač koji omogućava udobnost i praktičnost sustava pametnih kuća, zbog toga što se s njima može upravljati iz bilo kojeg dijela kuće, pomoću daljinskog upravljača se upravlja čitavim sustavom ili samo nekim njegovim komponentama. U strujne utičnice se ugrađuju mikro moduli koji su bitni zbog spajanja na gradsku mrežu jer definiraju adresu uređaja. Oni se koriste i kod standardnih prekidača za rasvjetu koje nije potrebno mijenjati zbog ugradnje mikro modula. Ukoliko se želi upravljati roletama, prozorima i zavjesama potrebno je u karniše i na prozore ugraditi elektromotore i module kojima se može upravljati dizanje i spuštanje roleta i zavjesa, otvaranje prozora itd. Ti moduli su povezani sa centralnom konzolom, tako da taj podsustav možemo uključiti u cjelokupnu automatizaciju. Treba spomenuti i filtre koji dolaze kao standardna oprema ovisno o protokolu koji se koristi u svrhu automatizacije. Najčešće, za X-10 protokol, u standardnom paketu opreme dolaze tri filtra koji zauzimaju šest mjesta osigurača u razvodnoj kutiji. Što se tiče sigurnosnih sustava, u mnogo kuća su takvi sustavi već ugrađeni, tako da se kod pametnih kuća uz malo promjena na postojećem sustavu implementira noviji sustav sigurnosti pametnih kuća. Što se tiče ugrađivanja sustava pametne kuće u već sagrađene domove, postavlja se pitanje dali se isplati ugrađivati takav sustav, jer potencijalne korisnike ne privlači mijenjanje infrastrukture svog objekta. Kao odgovor na pitanje isplativosti ugrađivanja pametne kuće u već sagrađen objekt je potvrdan, samo što su onda troškovi nešto veći. Objekti koji su još u fazi projektiranja mogu optimalnije iskoristiti sustav. Sustavi koji se u Hrvatskoj primjenjuju koriste postojeću mrežu od 220 V, uz minimalna prilagođavanja i korekcije. Prilagođavanja se najviše odnose na ekrane osjetljive na dodir koji zahtijevaju posebno ožičenje. Svi senzori i daljinski upravljači najčešće koriste napajanje od AA baterija od 12 V. Već duže vrijeme se u svijetu proizvode uređaji koji podržavaju implementaciju određenih protokola, pa tako i cijelog sustava pametne kuće u njihov rad, pa nije potrebno mijenjanje kućanskih aparata kad se korisnik odluči ugraditi sustav pametne kuće. Oprema i protokoli koje možemo pronaći na tržištu najčešće su standardizirani po svjetskim i europskim normama (ISO,IEC,EN…), što osigurava njihovu pouzdanosti i sigurnost. Najmanju primamljivost za ugrađivanje sustava pametnih kuća je cijena koja se kreće ovisno o željama i potrebama korisnika. Stoga za neki minimum opreme, tj. za osnovnu opremu moramo izdvojiti 1000 eura, pa sve do 10000 eura. A neka srednja cijena se postavlja na otprilike 2500 eura.[27]

[27] Jakovčić, A.; Marić, M. :Pametna kuća, Seminarski rad, Svibanj 2007., Zagreb

2.5. Ušteda energije pametne kuće

Kontroliranje opterećenja i nadzor sustava pametne kuće je važna značajka zbog uštede energija sustava pametne kuće. Kako bi spriječio rizik od nestanka struje radi preopterećenosti sustav pametne kuće automatski invertira kada su zadane vrijednosti prekoračene. On na taj način, olakšava korisniku da od neugodnosti vraća glavni prekidač u kutiji s osiguračima do kojega je teško pristupiti. Sustav se zbog uštede energije, programira tako da ako dođe do preopterećenja, automatski isključuje zadana trošila koja su određena kao manje bitna. Integracijom klime i upravljanja rasvjetom u sustavu pametne kuće dodatno se poboljšava ušteda energije. Na primjer: moguće je postaviti sobnu temperaturu na ekonomični režim rada kad nitko nije kod kuće, ili da se sva svijetla automatski ugase kada se napusti objekt. Prednosti samog sustava pametne kuće se uočavaju kroz veliku uštedu energije, povećanjem ugodnosti života, slobodnog vremena i smanjenjem briga oko svakidašnjih zadataka. Npr. zalijevanje cvijeća i vrtova, paljenja vanjske rasvjete, zatvaranja prozora i roleta prije spavanja.[28] Slika 9. prikazuje cjelokupni sustav pametne kuće i njezinih pojedinačnih djelova kojima je moguće upravljati.



Slika 9. – Pametna kuća

Izvor: http://plane-tehnika.hr/wp-content/uploads/2015/07/pametne-kuce.jpg (29.08.2017.)

[28]http://pimami.hr/vimarautomatika/?gclid=Cj0KEQjwwqXMBRCD1afDldbp5qEBEiQAP4BDaJipAxrcGP1m2nwZVpvh1wzDQCzzcwgIxTVedtNpvkgaAsh-8P8HAQ (29.08.2017.)

3. MODEL PAMETNE KUĆE

Model pametne kuće sastoji se od dva glavna dijela, prvi dio je sami konstrukcijski dio, sačinjen od drvene mase, koji poprima izgled i oblik kuće. Drvena masa od koje je napravljen konstrukcijski dio, naziva se šperploča, debljine 4 mm. Nakon obrade drvene mase rezanjem, lijepljenjem se postigao izgled kuće. Drugi dio koji sačinjava ovaj model pametne kuće je onaj upravljački dio. U upravljačkom dijelu za upravljanje ostalim komponentama koristi se Arduino UNO mikrokontroler, baziran na ATmega 238P mikro čipu. Komponente kojima se upravlja pomoću ovog mikrokontrolera su: senzor pokreta, ultrazvučni senzor, temperaturni senzor, servo motor, Bluetooth modul, led diode, zujalica i sklopka. Mogućnosti koje će koristiti ovaj sustav upravljanja modelom pametne kuće su sljedeće:

1. SUSTAV UPRAVLJANJA RASVJETOM

   Za sustav upravljanja rasvjetom koristiti će se jedan senzor pokreta koji ovisno o svojem stanju postavlja led diodu koja predstavlja rasvjetno tijelo na ovom modelu pametne kuće. U ovom upravljačkom sustavu rasvjetom se može i upravljati pomoću android uređaja u ručnom režimu rada pomoću Bluetooth modula i Android uređaja.

2. SUSTAV UPRAVLJANJA KLIMATIZACIJOM

   U sustavu upravljanja klimatizacijom koristi se temperaturni senzor koji ovisno o izmjerenoj temperaturi postavlja dvije led diode, od kojih jedna predstavlja klima uređaj za hlađenje, a druga grijač prostorije. I u ovom slučaju moguće je upravljati ručno pomoću Android uređaja.

3. SUSTAV UPRAVLJANJA GARAŽNIM VRATIMA

   Sustav upravljanja garažnim vratima koristi senzor pokreta koji ovisno o detekciji prolaska daje signal Arduino uređaju koji daje signal servo motoru koji otvara i zatvara garažna vrata. Naravno moguće je i ručno upravljanje servo motorom pomoću Android uređaja.

4. SUSTAV UPRAVLJANJA ALARMNIM SUSTAVOM

   Sustav upravljanja alarmnim sustavom koristi ultrazvučni modul koji mjeri udaljenost osobe u ovom slučaju bilo kakva detekcija unutar namještenih postavka. Aktivacijom pomoću ultrazvučnog senzora Arduino daje signal koji uključuje zujalicu koja predstavlja zvučne signale i uključuje rasvjetu koja vizualno prikazuje detekciju neželjenog pristupa.

Bluetooth modul nam služi za ručno upravljanje sustavom modela pametne kuće, preko njega se spajamo sa Android uređajem na Arduino kojemu šaljemo komande koje želimo da naš sustav izvrši, kao što su u ovom slučaju: paljenje i gašenje rasvjete, paljenje i gašenje klimatizacije, otvaranje i zatvaranje garažnih vrata, te na posljetku i aktiviranje i deaktiviranje alarmnog sustava . Sklopka koja je navedena u korištenje u sustavu modela pametne kuće služi nam za odabir opcije dali želimo upravljati sustavom ručno preko Android uređaja ili automatski gdje su svi senzori aktivni.

3.1. Potrebne komponente

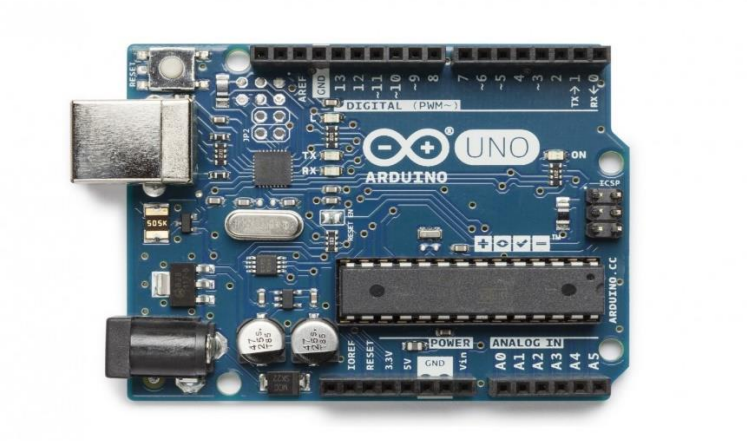
3.1.1. Arduino UNO mikrokontroler

Arduino je ime za otvorenu računarsku i softversku platformu koja omogućava dizajnerima i konstruktorima stvaranje uređaja i naprava koje omogućava spajanje računala s fizičkim svijetom tj. stvaranje interneta stvari. Arduino je stvorila talijanska tvrtka SmartProjects 2005. rabeći 8-bitne mikrokontroler Amtel AVR, da bi stvorili jednostavnu malu i jeftinu platformu s kojom bi mogli lakše povezivati računala s fizičkim svijetom. Dizajneri su izabrali ime Arduino po imenu kafića u kojem su se dizajneri sastajali kada su stvarali projekt.[29]

Arduino Uno kojeg nam prikazuje slika 10. je mikrokontrolerska pločica temeljena na ATmega328P mikro čipu. Ima 14 digitalnih ulaznih / izlaznih priključaka (od kojih se 6 može koristiti za pulsno - širinsku modulaciju), 6 analognih ulaza, 16 MHz kvarcnog kristala, USB priključak, utičnicu za napajanje, ICSP priključak i gumb za resetiranje. Sadrži sve što je potrebno za podršku mikrokontrolera, jednostavno se može spojiti na računalo pomoću USB kabela ili s adapterom ili baterijom AC-DC za početak rada. Arduino UNO je jako poželjan

[29] https://hr.wikipedia.org/wiki/Arduino (29.08.2017)

mikrokontroler zbog toga što i ako nešto pođe po zlu, može se zamijeniti mikro čip, koji je mozak ove mikrokontrolerske pločice.[30]



Slika 10. – Arduino UNO mikrokontroler

Izvor:

https://storecdn.arduino.cc/uni/catalog/product/cache/1/image/1800x/ea1ef423b933d797cfca49bc5855eef6/A/0/A000066_front_2.jpg (29.08.2017.)

Tablica 1. – Karakteristike Arduino UNO uređaja

Naziv karakteristike	Karakteristika
Mikrokontroler	ATmega 328
Radni napon	5 V
Ulazni napon (preporučeno)	7 – 12 V
Ulazni napon (limit)	6 – 20 V
Digitalni pinovi	14 (6 njih sa PWM)
Analogni pinovi	6
Struja prema pinovima	40 mA DC
Struja prema 3,3 V pinu	50 mA
Flash memorija	32 kB
SRAM	2 kB
EEPROM	1 kB
Takt	16 MHz

Izvor: http://arduino.bloger.index.hr/default.aspx (13.09.2017.)

[30] https://store.arduino.cc/arduino-uno-rev3 (29.08.2017.)

3.1.2. Bluetooth HC 06 modul

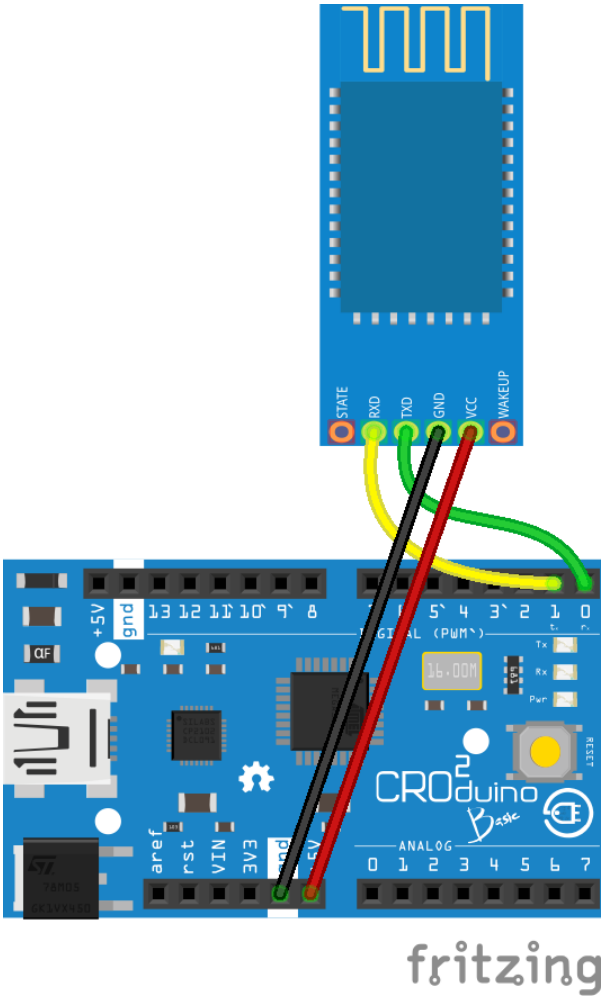
Bluetooth je način bežičnog prijenosa informacija, osmišljen od strane Ericssona 1994. kao alternativa za RS-232 žični prijenos podataka. Za prijenos podataka koriste se radio valovi frekvencije od 2.402GHz do 2.480GHz. Prednost Bluetootha, ispred drugih bežičnih načina komunikacije, je vrlo jednostavno uparivanje. Kako bi izbjegao međusobnu interakciju s drugim uređajima koji koriste iste frekvencije, Bluetooth šalje vrlo slabi signal oko 1mW. Za usporedbu, signal jačih mobilnih telefona je 3W. To mu ograničava domet na 10m, što se naravno može povećati i do 100m. Uređaji koji se povezuju ne moraju biti fizički vidljivi jedan drugom. Signal prolazi kroz zidove i druge objekte, tako da se više uređaja može povezati u različitim prostorijama. Kada već spominjemo više uređaja, Bluetooth može biti povezan na 8 uređaja istovremeno u prostoru od 10m bez da smetaju jedan drugom. Kako bi to omogućio Bluetooth uređaj mijenja frekvenciju, u gore spomenutom rasponu, 1,600 puta svake sekunde.[31]

Slika 11. - Bluetooth HC 06 modul

Izvor: http://www.instructables.com/id/Add-bluetooth-to-your-Arduino-project-ArduinoHC-06/ (29.08.2017.)

[31] https://e-radionica.com/hr/blog/2016/04/06/bluetooth/ (29.08.2017.)

Bluetooth HC 06 modul prikazan na slici 11. je modul koji se koristi za spajanje Arduino uređaja sa svakim uređajem koji podržava Bluetooth. HC 06 modul ima četiri pina za spajanje na Arduino uređaj a spajanje je prikazano na slici 12, a nazivi pinova su: VCC - napajanje samog modula koje se spaja na Arduino uređaj ovisno o naponu koji može biti od 3-6 VDC, GND – uzemljenje modula koje se spaja na uzemljenje Arduino uređaja, RXD – serijski ulaz modula, koji se spaja na TX priključak Arduino uređaja, TDX – serijski izlaz modula, spaja se na RX priključak Arduino uređaja.[32]



Slika 12. – Spajanje Bluetooth modula sa Arduino uređajem

Izvor: https://e-radionica.com/hr/blog/2016/04/06/bluetooth/ (29.08.2017.)

[32] https://e-radionica.com/hr/blog/2016/04/06/bluetooth/ (29.08.2017.)

3.1.3. Android uređaj

Google Android je prvi otvoreni operacijski sustav za mobilne uređaje (mobilni telefoni, tableti, netbook računala, Google TV) pokrenut od strane Google Inc. i vođen od strane Open Handset Alliance - grupe koja danas broji preko 80 tehnoloških kompanija između kojih se nalaze T-Mobile, HTC, Intel, Motorola, Qualcomm, i drugi, čiji je cilj ubrzati inovacije na području mobilnih operacijskih sustava, a samim time ponuditi krajnjim kupcima bogatije, jeftinije i bolje iskustvo uporabe. Android je modularan i prilagodljiv pa tako postoje slučajevi njegovog prenošenja (portanja) na razne uređaje kao što su čitači elektronskih knjiga, mobilni telefoni, prijenosnici, te multimedijski playeri.[33] Slika 13. prikazuje najobičniji Android uređaj kojim se u današnje vrijeme svi susrećemo.

Slika 13. - Android uređaj

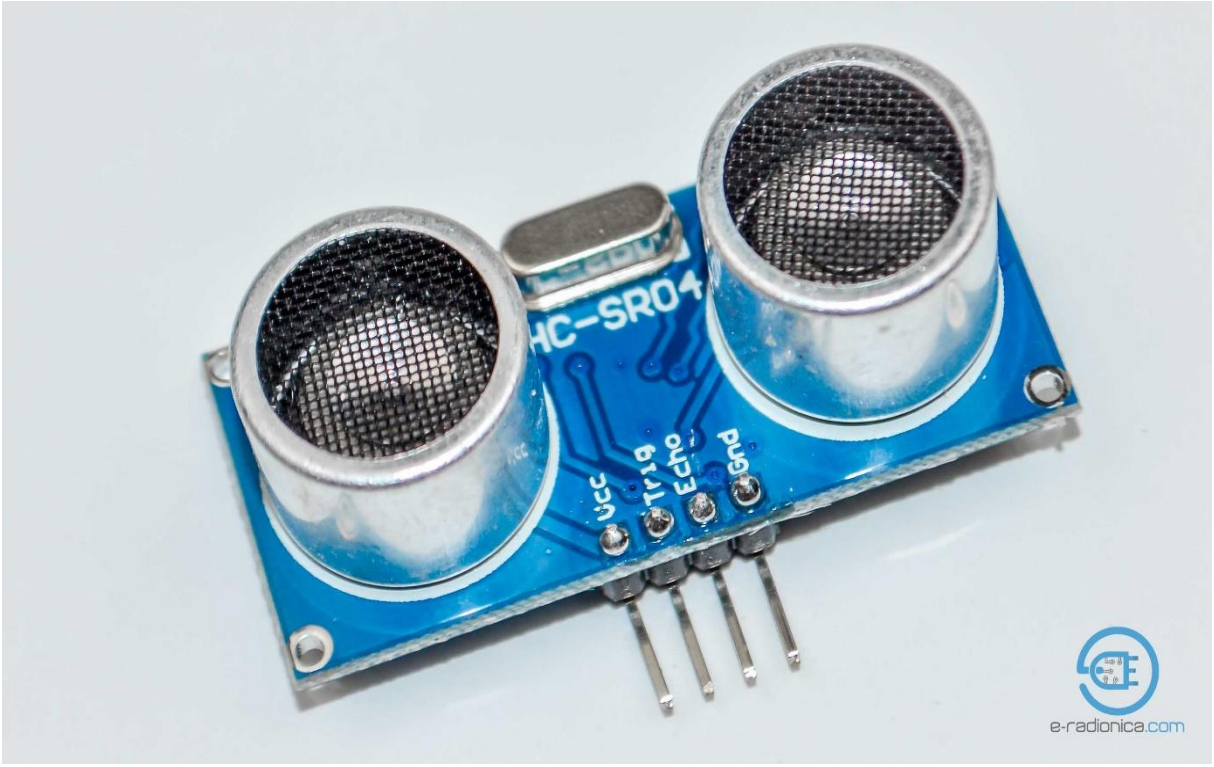
Izvor:

https://hr.wikipedia.org/wiki/Android_(operacijski_sustav)#/media/File:Nexus_5_L.png (29.08.2017.)

[33] https://hr.wikipedia.org/wiki/Android_(operacijski_sustav) (29.08.2017.)

3.1.4. Ultrazvučni modul

Ultrazvučni modul prikazan na slici 14. je modul koji radi na principu slanja ultrazvučnih valova kako bi pomoću njih odredio udaljenost od nekog predmeta ili osobe. Dva osnovna dijela modula na kojima se temelji princip rada su trig (okidač) i echo (refleksija). Mikrokontrolerom šaljemo 5V na trig pin modula u trajanju, minimalno, 10 mikro sekundi. Na taj način aktiviramo ultrazvučni transduktor koji odašilje 8 impulsa od 40 kHz i čeka njihovu refleksiju. Kada senzor registrira reflektirani impuls šalje podatke natrag mikrokontroleru preko echo pina. Navedeni podaci su zapravo vrijeme trajanja reflektiranog pulsa, od 150 mikrosekundi do 25 milisekundi. Ako "jeka" traje duže od 35 milisekundi, senzor registrira da je predmet izvan dosega.[34]



Slika 14. – Ultrazvučni modul

Izvor: http://e-radionica.com/wp/wp-content/uploads/2015/03/DSC_8467.jpg (29.08.2017.)

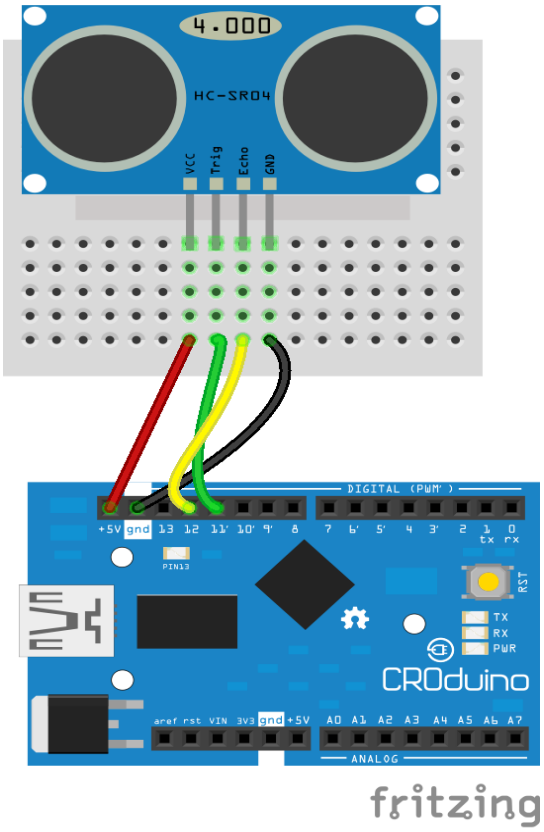
[34] https://e-radionica.com/hr/blog/2015/08/19/kkm-ultrazvucni-modul-hc-sr04/ (29.08.2017.)

Tablica 2. – Karakteristike ultrazvučnog senzora HC-SR04

Naziv karakteristike	Karakteristika
Domet	2 – 200 cm
Preciznost	3 mm
Efektivni kut mjerenja	15 °
Napon	5 V
Maksimalna mirna struja	2 mA
Radna struja	15 mA
Ultrazvučna frekvencija	40 kHz
Dimenzije	45 x 20 x 15 mm

Izvor: https://cdn.sparkfun.com/datasheets/Sensors/Proximity/HCSR04.pdf (15.09.2017.)

Slika 15. nam prikazuje kako povezati ultrazvučni modul sa Arduino uređajem.

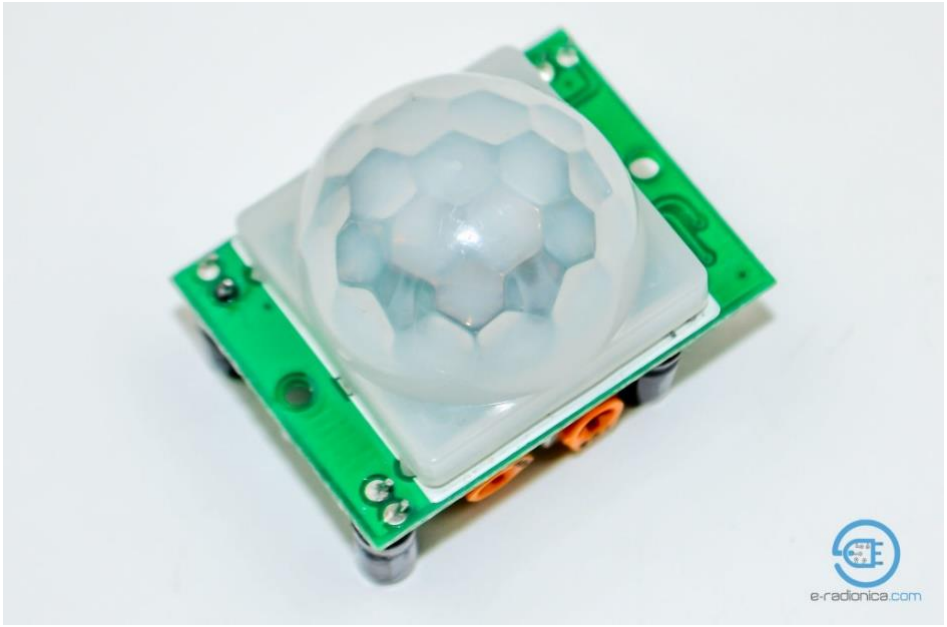


Slika 15. – Povezivanje Ultrazvučnog modula sa Arduinom

Izvor: http://e-radionica.com/wp/wp-content/uploads/2015/03/Untitled-Sketch_bb.png (29.08.2017.)

3.1.5. PIR senzor[35]

PIR senzor, Passive Infrared, Pyroelectric ili IR motion senzor, je prikazan na slici 16. te on omogućava detektiranje pokretan unutar dosega senzora. Osnovni dio je piro električni senzor koji detektira količinu infracrvene radijacije. Svako biće, predmeti zrače s određenim malim stupnjem radijacije, a što je nešto toplije zrači jače. Stoga se ovaj senzor, uglavnom, koristi za detekciju pokreta živih bića. Imajte na umu da Vam ovaj senzor ne može reći na kojoj udaljenosti je nešto detektirano.



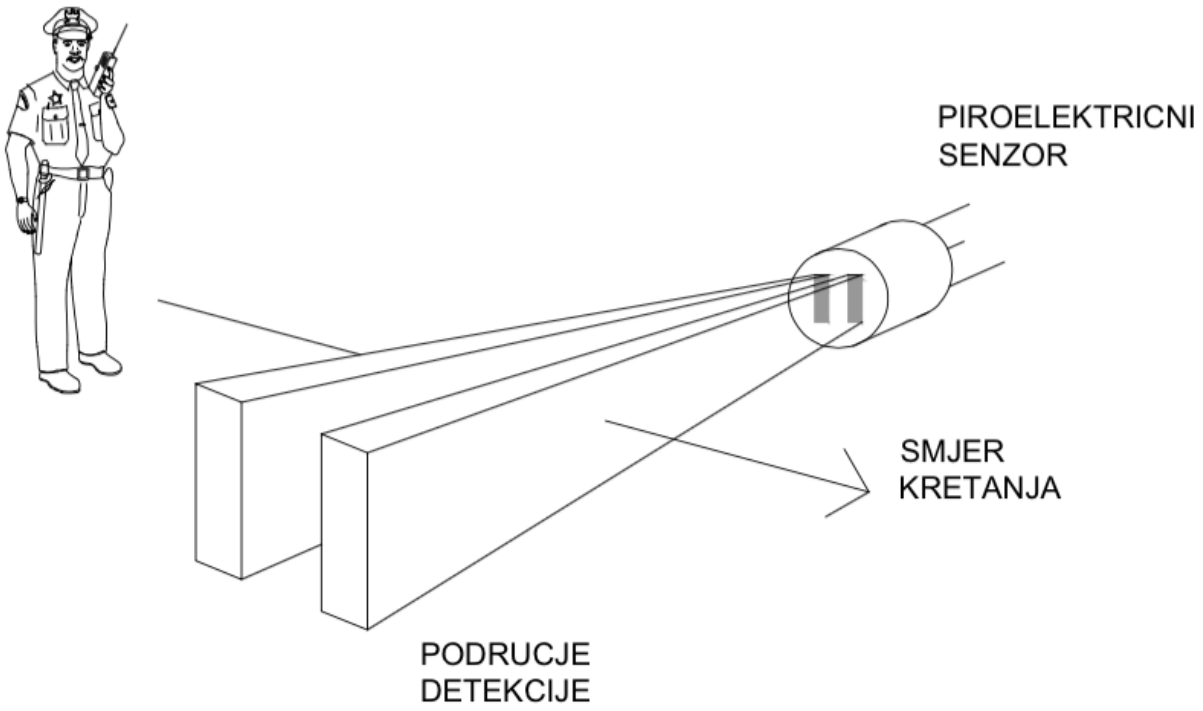
Slika 16.– PIR senzor

Izvor: http://e-radionica.com/wp/wp-content/uploads/2015/05/DSC_8529-1024x678.jpg (29.08.2017.)

Kao što je već spomenuto osnovni dio je piro električni senzor koji na sebi ima dva proreza, polovice. Ako jedan prorez očitava više radijacije od drugog senzor će se automatski prebaciti iz HIGH u LOW, ili obratno. Primjerice, senzor se nalazi na zatvorenom i očitava radijacije zidova i drugih predmeta u svom dosegu. Kada prolazi toplo tijelo, poput čovjeka ili životinje, prvo zahvaća jednu polovicu senzora koja očitava promjenu radijacije u prostoru što je vidljivo na slici 17. Obratno se događa kada toplo tijelo napušta prostor u dosegu senzora. Upravo su te

[35] https://e-radionica.com/hr/blog/2015/08/19/kkm-hc-sr501/ (29.08.2017.)

promjene ono što senzor registrira. Drugi važan dio HC-SR501 senzora je leća. Ona može varirati s obzirom na zakrivljenost, širinu, materijal, uzorak i sl., pa se upravo većina prave čarolije ovog senzora događa s optikom. Sve navedeno uvelike mijenja konačne karakteristike senzora, što otvara prostor vrlo detaljnoj prilagodbi.



Slika 17. – Princip rada PIR senzora

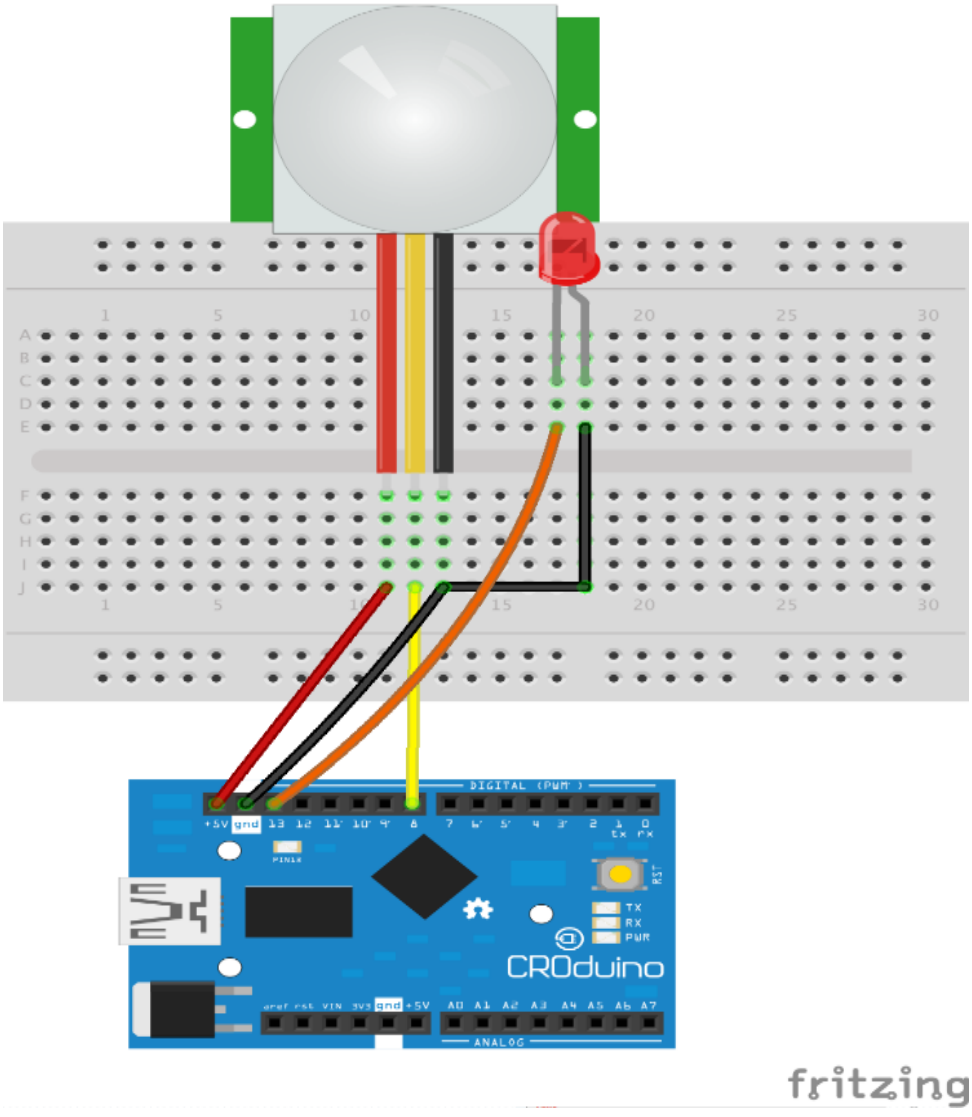
Izvor: http://e-radionica.com/wp/wp-content/uploads/2015/05/Screen-Shot-2015-05-18-at-16.24.48.png (29.08.2017.)

Tablica 3. – Karakteristike PIR senzora HC-SR501

Naziv karakteristike	Karakteristika
Napon	DC 4,5 – 20 V
Struja	50 µA
Kut	< 120
Udaljenost	3 m – max. 7 m
Promjer leće	23 mm
Dimenzije	32 x 24 mm

Izvor: https://cdn-learn.adafruit.com/downloads/pdf/pir-passive-infrared-proximity-motion-sensor.pdf (15.09.2017.)

Prije spajanja pinova modula na Arduino značajna su dva potenciometra koja se nalaze na modulu. Pomoću njih podešavamo osjetljivost (na pločici označeno s "Sx") i vrijeme delay-a ("Tx") na sljedeći način: okretanjem u smjeru kazaljke na satu veća osjetljivost i duži delay, smjer obratno od kazaljke na satu niža osjetljivost i kraći delay. To znači da se modul može koristiti i bez kontrolera. Ovaj modul daje HIGH u stanju mirovanja te LOW kada detektira pokret. Slika 18. nam prikazuje kako spojiti PIR senzor na Arduino uređaj.



Slika 18. – Spajanje PIR senzora sa Arduino uređaj

Izvor: http://e-radionica.com/wp/wp-content/uploads/2015/05/Progetto-PIR-con-accensione-LED_bb.png (29.08.2017.)

3.1.6. Servo motor[36]

Servo motor je vrlo mali, precizni i povoljni motorčić kojemu možemo mu odrediti tj. zadati kut, brzinu okretanja i akceleraciju. Servo motori sastoje se od DC motora, potenciometra i kontrolne pločice. Kako se motor okreće, otpor na potenciometru se mijenja. Na taj način kontrolna pločica može precizno regulirati okret, brzinu, akceleraciju i smjer okreta. Slika 19. predstavlja servo motor.

Slika 19. - Servo motor

Izvor: https://e-radionica.com/wp/hrvatski/wp-content/uploads/sites/3/2016/06/dsc_4840-1024x678.jpg (29.08.2017.)

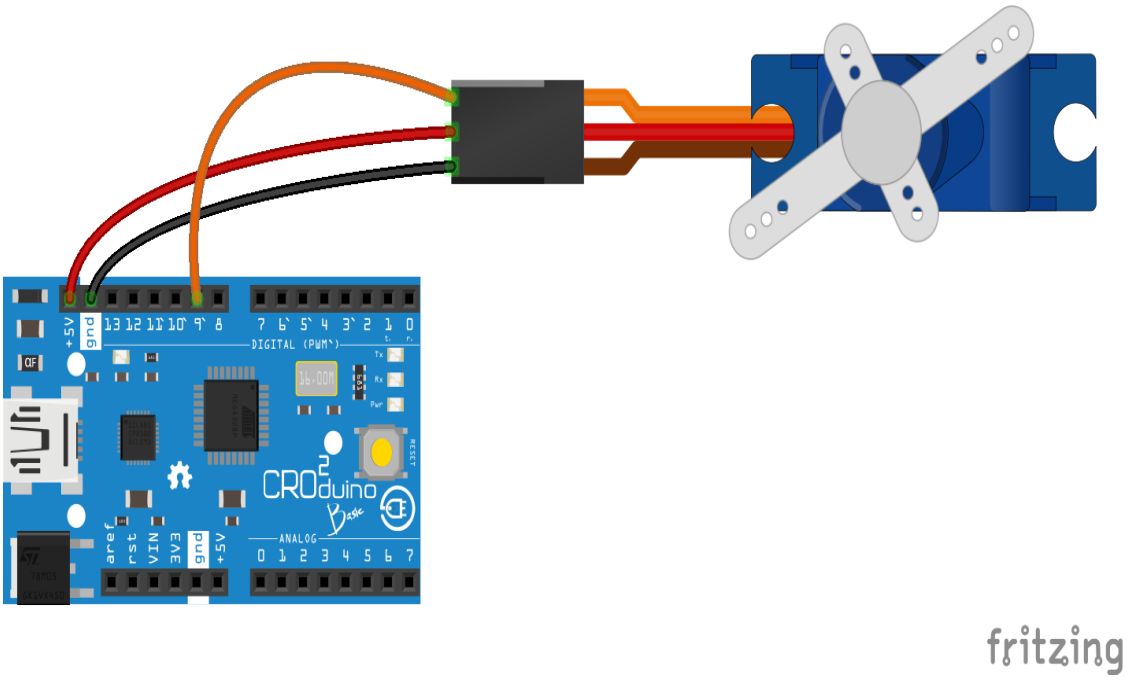
Tablica 4. – Karakteristike Servo motora SG90

Naziv karakteristike	Karakteristika
Težina	9 g
Dimenzije	23x 12 x 31 mm
Okretni moment	1,8 kg/cm
Brzina okretaja	0,1 s / 60 º
Radni napon	4,8 V
Temperatura	0 ºC – 55 º C

Izvor: http://www.micropik.com/PDF/SG90Servo.pdf (15.09.2017.)

[36]https://e-radionica.com/hr/blog/2016/06/23/kkm-servo-motor-tower-pro-sg90/

Servo motore kontroliramo slanjem PWM signala preko kontrolne žice. Obavezno se koriste neke od PWM pinova s Aroduina (3, 5, 6, 9, 10 ili 11). Ovisno o duljini pulsa rotor se okreće na željenu poziciju. Servo motor ima mogućnost okreta 90 stupnjeva u oba smjera ukupno 180 stupnjeva. Kada servo motor dobije naredbu za pokret okretati će se do zadane pozicije i zadržati ju. Ako vanjska sila djeluje na motor dok drži poziciju, servo motor će se odupirati pokušavajući zadržati istu. Maksimalna sila koju servo može podnijeti je zapravo moment iz karakteristika. Slika 20. Predstavlja kako se spaja servo motor sa Arduino uređajem.

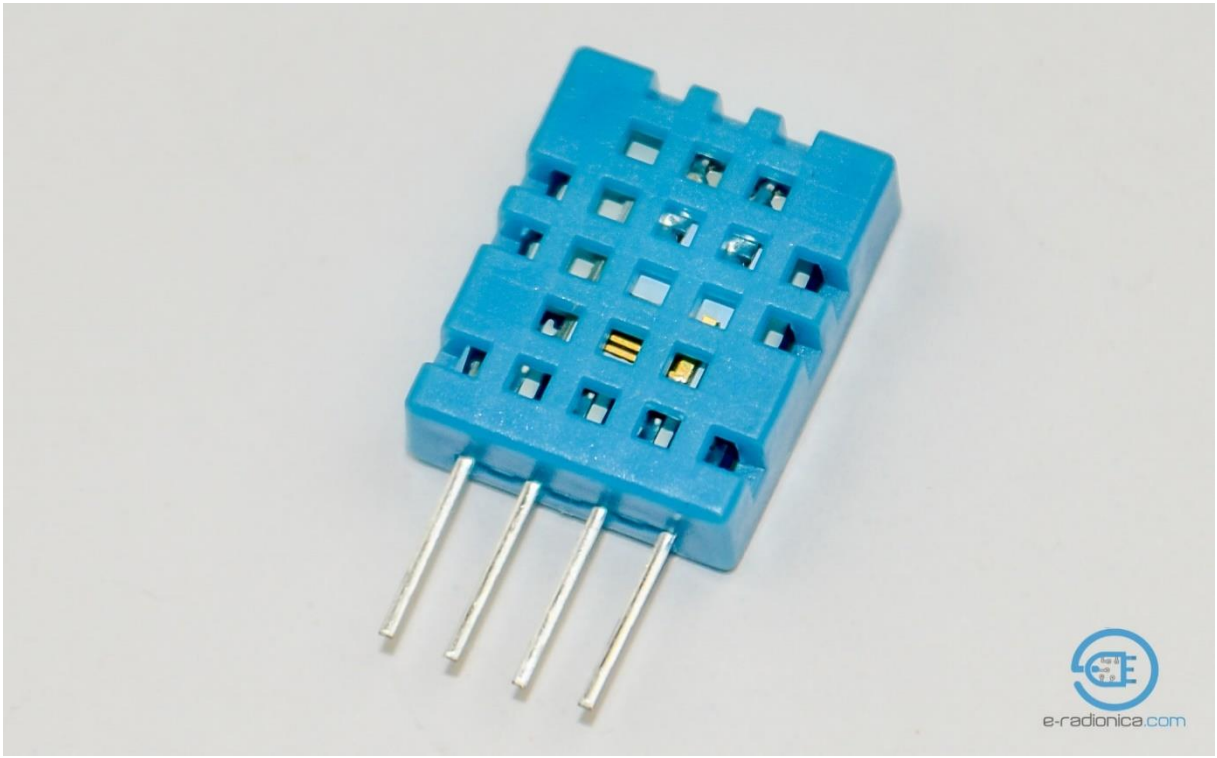


Slika 20. – Spajanje servo motora sa Arduino uređajem

Izvor: https://e-radionica.com/wp/hrvatski/wp-content/uploads/sites/3/2016/06/Untitled-Sketch_bb.png (29.08.2017.)

3.1.7. Senzor temperature DHT 11[37]

DHT 11 je senzor koji mjeri temperaturu i vlažnost zraka, a prikazan je na slici 21.. Unutar samog modula nalaze se, dvije vrste senzora: senzor temperature: mali termistor(vrsta senzora temperature) zalemljen na pločici i senzor vlage: mala tiskana pločica je nalemljena na osnovnu. Ima otvorene(neprekrivene) bakrene vodove koji su međusobno vrlo blizu. Što je više vlage u zraku, više vlage dolazi i na same vodove te je otpor među njima manji(ili tek počinje postojati kada se pojavljuje voda — to je razlog zašto senzor očitava vrijednosti tek od 20%; tek onda može očitati nekakav otpor). Osim navedene dvije komponente, tu je još jedan integrirani krug koji analizira ulaze od ova dva senzora te komunicira s Arduinom.



Slika 21. – DHT 11 temperaturni senzor

Izvor: http://e-radionica.com/wp/wp-content/uploads/2015/03/dht11.jpg (29.08.2017.)

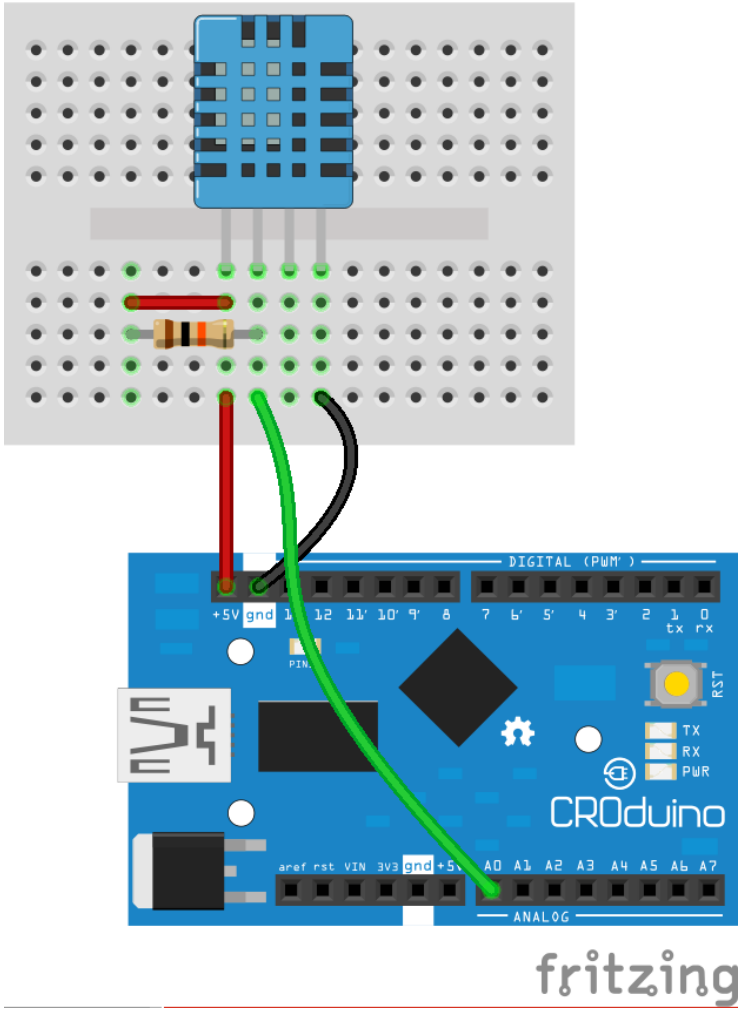
[37] https://e-radionica.com/hr/blog/2015/08/19/kkm-dht11/ (29.08.2017.)

Tablica 5. – Karakteristike DHT 11 temperaturnog senzora

Naziv karakteristike	Karakteristika
Raspon mjerenja temperature	0 ° - 50 ° C
Odstupanje pri mjerenju temperature	+/- 2 ° C
Napon	3 V – 5 V
Struja u mirovanju	100 µA
Radna struja	2,5 mA

Izvor: http://www.micropik.com/PDF/dht11.pdf (15.09.2017.)

Slika 22. prikazuje spajanje DHT 11 temperaturnog senzor sa Arduino uređajem.



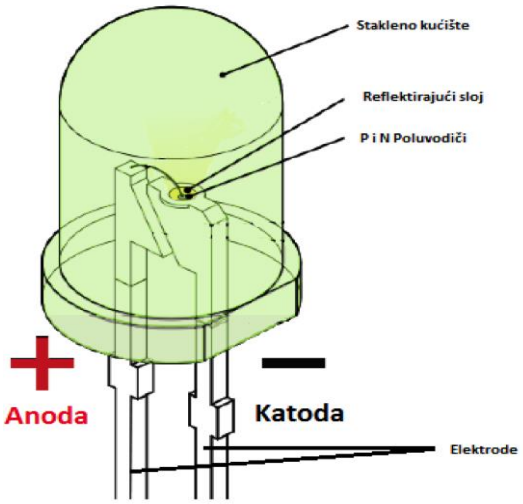
Slika 22. – Povezivanje DHT 11 senzora sa Arduinom

Izvor: http://e-radionica.com/wp/wp-content/uploads/2015/03/Untitled-Sketch_bb2.png (29.08.2017.)

3.1.8. Ostalo

1. Svjetleća dioda

Svjetleća dioda prikazana na slici 23. (LED, skraćeno. od engl. Light Emitting Diode) je poluvodički elektronički element koji pretvara električni signal u optički. Propusno polarizirana svjetleća dioda emitira elektromagnetsko zračenje na principu spontane emisije uzrokovane rekombinacijom nosilaca električnoga naboja. Elektroni prelazeći iz vodljivog u valentni pojas, oslobađaju energiju, koja se dijelom očituje kao toplina, a dijelom kao zračenje. Izrađuju se od galija, arsena i fosfora (GaAsP), gdje se omjerom arsena i fosfora određuje širina zabranjenoga pojasa i time posredno frekvencija zračenja, ili pak, za heterostrukturne izvedbe pogodne za optičke komunikacije, od istih elemenata uz dodatak Indija. Primjenjuje se najčešće kao indikator, npr. na komandnim i signalnim pločama uređaja i strojeva ili kao alfanumerički pokazivač na zaslonima džepnih kalkulatora, zatim za ukrasno osvjetljenje te u industriji zabave, za signalnu rasvjetu za bicikle, automobilska svjetla, kao elementi za daljinski prijenos signala u upravljačkim krugovima, npr. kod televizorskih daljinskih upravljača. Veliko područje primjene imaju i u optičkim komunikacijama, gdje služe za prijenos informacija na kraće udaljenosti multi modnim optičkim vlaknom.[38]



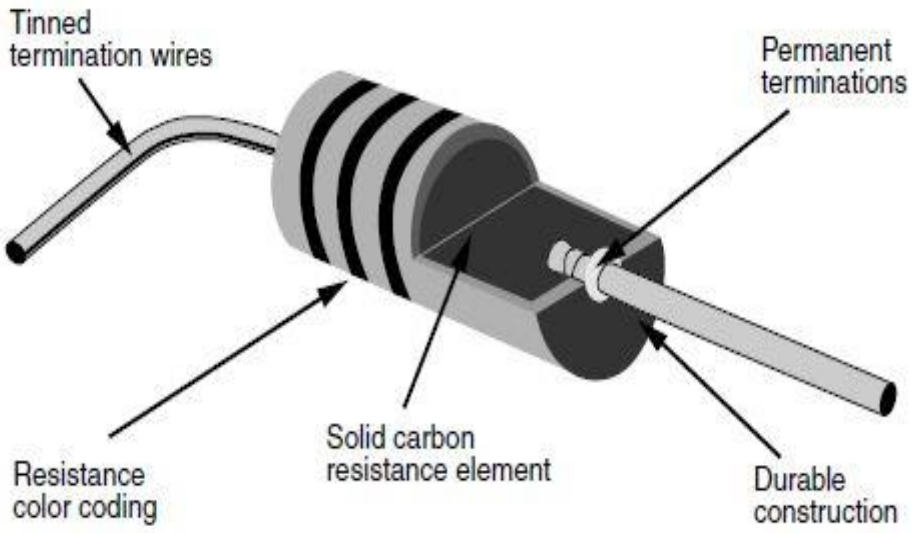
Slika 23. -Svjetleća dioda

Izvor: http://e-radionica.com/wp/wp-content/uploads/2015/05/x.png (31.08.2017.)

[38] http://www.enciklopedija.hr/natuknica.aspx?ID=59111 (31.08.2017.)

2. Otpornik

Otpornik je element strujnog kruga koji pruža otpor prolasku struje pri čemu je odnos između jakosti struje i napona između priključaka u skladu s Ohmovim zakonom. Karakteristična veličina otpornika je električni otpor koji je jednak naponu na otporniku podijeljenom sa strujom koja protječe kroz otpornik. U otporniku se električna energija pretvara u toplinu (općenito u neki drugi oblik). Otpornik ponekada nazivamo trošilo. Kod idealnog otpornika otpor ostaje konstantan bez obzira na dovedeni napon ili protjecanje struje kroz element. Stvarni otpornici osim otpornosti mogu imati primjese induktivnosti ili kapacitivnosti. Osim toga stvarni otpornici mogu mijenjati svoj otpor ovisno o naponu, temperaturi, tlaku i sl. U početku smatramo da su otpornici idealni.[39] Slika 24. prikazuje od čega se sastoji jedan klasični žični otpornik.



Slika 24.– Otpornik

Izvor: http://3.bp.blogspot.com/-h4OVYkj6lZ4/Uelgt9ye01I/AAAAAAAAAU0/bzU-wyBM38Y/s1600/konstrukcija-otpornika.jpg (31.08.2017.)

[39] http://osnove.tel.fer.hr/VJEZBEOE/DC_2.htm (31.08.2017.)

3. Zujalica

Zujalica ili beeper je zvučno-signalizacijski uređaj prikazan na slici 25.. Može biti mehanički, elektromehanički ili piezoelektrični. Primjer mehaničkog je "joy buzzer" koji se koristi na starim budilicama, elektromehanički je onaj povezan na zvono vaših ulaznih vrata. U ovom slučaju koristi se piezoelektrični tip, koji koristi piezo efekt. Piezo efekt je sposobnost materijala da generira električni naboj na dano mehaničko opterećenje. Ime potječe od grčke riječi piezo što znači pritisnuti. Kada se piezoelektrični materijal izloži mehaničkom opterećenju (pritisak) događa se premještanje pozitivnog i negativnog naboja, što rezultira vanjskim električnim poljem. Odličan piezoelektrični materijal je quartz, koji pokreće quartz satove. Što se tiče zujalice, u njoj se nalazi mala zavojnica i magnet. Kada struja poteče kroz zavojnicu, ona se magnetizira i povlači prema magnetu. Iz prethodno objašnjenog piezo efekta, jasno nam je da će to izazvati mali "klik". Kako se to ponavlja nekoliko tisuća puta u sekunda, "klik" postaje zvuk.[40]

Slika 25. – Zujalica
Izvor: http://e-radionica.com/wp/wp-content/uploads/2015/05/DSC_8597-1024x678.jpg (31.08.2017.)

[40] https://e-radionica.com/hr/blog/2015/08/19/buzzer/ (31.08.2017.)

4. Foto otpornik

Foto otpornik (engl.. photoresistor ili Light dependent resistor - LDR) je otpornik, čiji se električni otpor smanjuje s povećanjem intenziteta ulazne svjetlosti. Foto otpornik se izrađuje od poluvodiča sa velikim električnim otporom. Ako svjetlo padne na foto otpornik, sa dovoljno velikom frekvencijom (granična frekvencija), poluvodič će upiti fotone svjetlosti i izbaciti elektrone, koji stvaraju električnu struju, u zatvorenom strujnom krugu. Ima puno vrsta foto otpornika. Jeftine kadmij sulfid ćelije se mogu naći u puno različitih uređaja, kao što su kamere, ulične svjetiljke, radio satovi, alarmi i vanjski satovi. Olovni sulfid (PbS) i Indiji antimon (InSb) foto diode se koriste za srednje infracrveno područje, Germaniji – bakar (Ge:Cu) foto diode su najbolje za daleko infracrveno područje i koriste se u infracrvenoj astronomiji i infracrvenoj spektroskopiji.[41] Slika 26. prikazuje fotootpornik.

Slika 26. – Fotootpornik

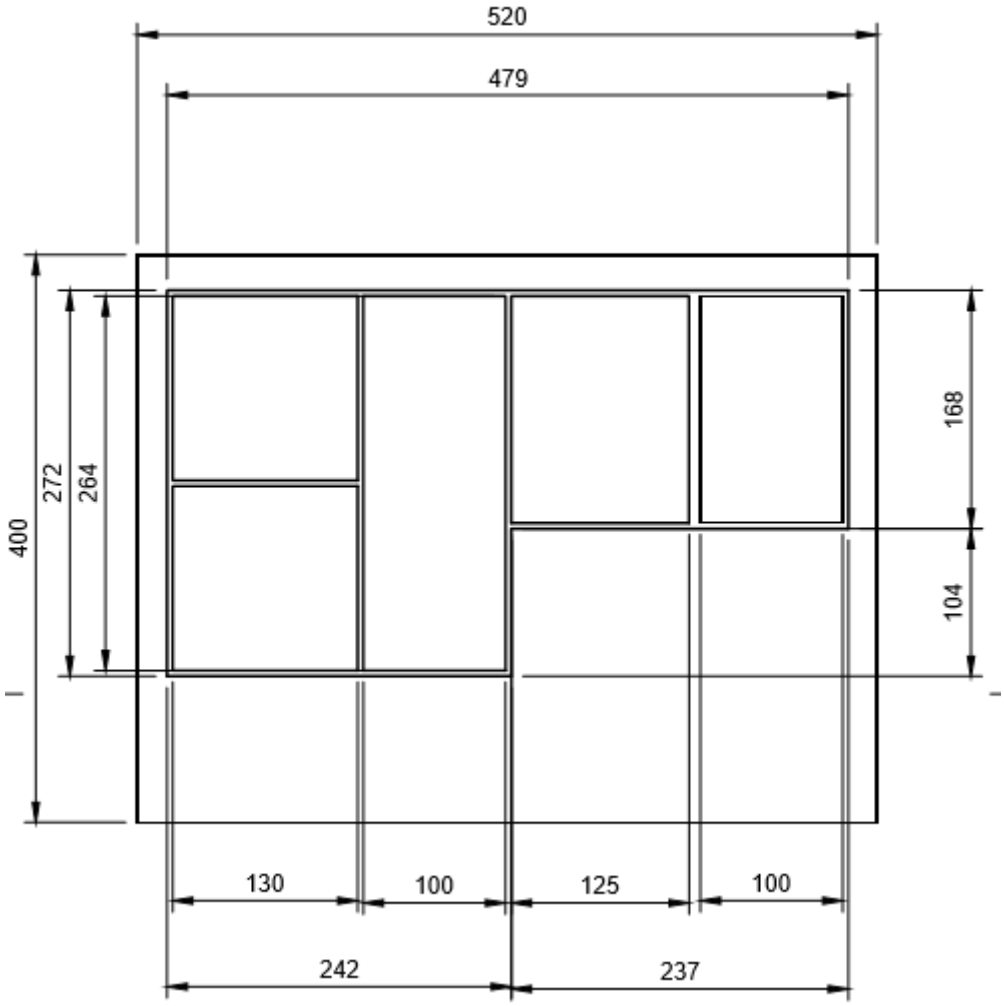
Izvor:

http://www.saperel.com/image/cache/data/Proizvodi/Senzori/Foto%20otpornik%20mali/09088-02-L-500x500.jpg (31.08.2017.)

[41] https://hr.wikipedia.org/wiki/Fotootpornik (31.08.2017.)

3.2. Konstrukcija i izrada modela pametne kuće

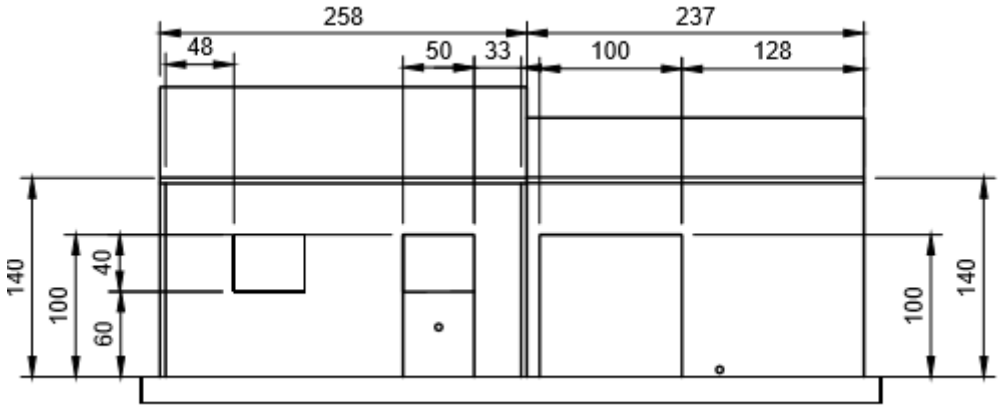
Za konstruiranje modela pametne kuće koristio se program Autodesk Fusion 360. Model pametne kuće je konstruiran u 3D obliku, te potom su iz 3D oblika izvučeni nacrti. Model pametne kuće je dizajniran tako što se sastoji od dva sloja, jedan je unutrašnji sloj modela, a drugi je vanjski sloj. Slika 27. nam prikazuje unutarnji sloj modela pametne kuće.



Slika 27. – Nacrt unutrašnjeg sloja modela pametne kuće

Izvor: Autor

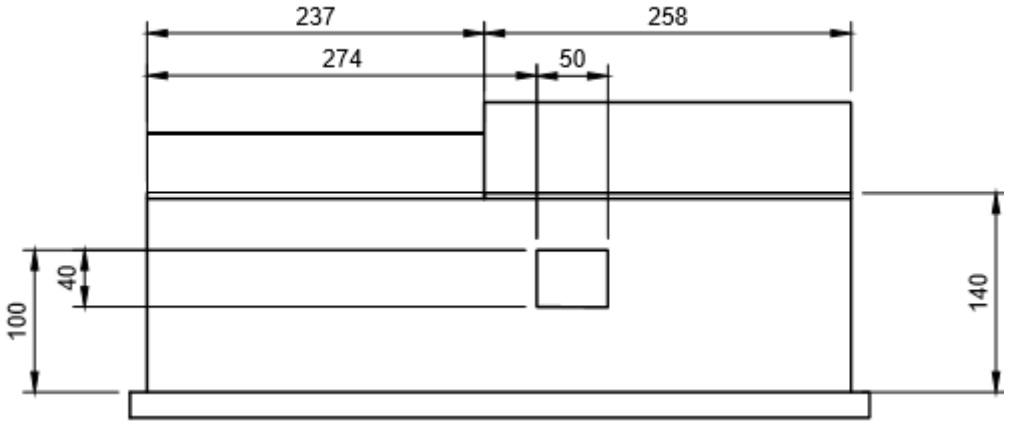
Kao što vidimo glavna podloga modela pametne kuće je dimenzija (sve dimenzije su u milimetrima) 400 x 520. Unutarnji sloj modela je u obliku slova „L“, te sve pripadajuće dužine stranica su prikazane na slici .



Slika 28. – Prednji pogled na konstrukciju modela pametne kuće

Izvor: Autor

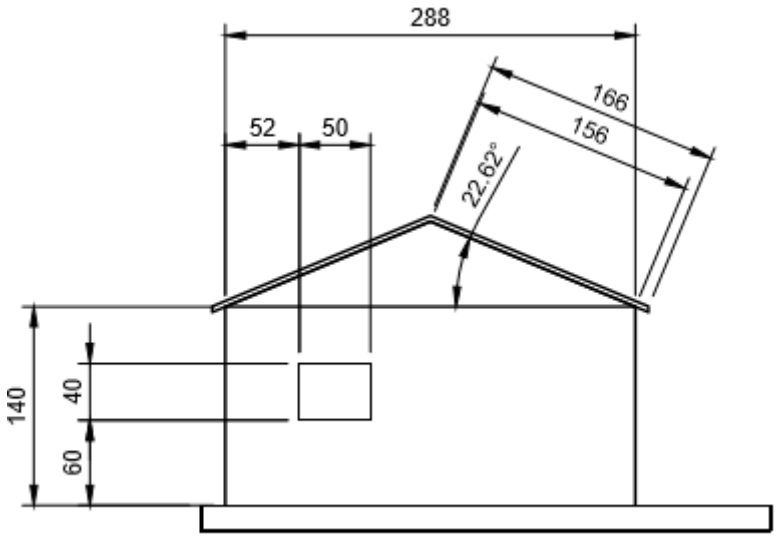
Prednja strana modela pametne kuće prikazana je na slici 28. Vidimo da nam je visina modela pametne kuće 140 mm, a širina modela 495 mm, na ovoj slici još i vidimo otvor za garažna vrata koja su dimenzije 100x100, te otvor za prozor dimenzija 40x50 i otvor za ulazna vrata dimenzija 50x100. Na ovoj slici je prikazano i krovište kuće koje nije uračunato u visinu modela kuće.



Slika 29. – Stražnji pogled na konstrukciju modela pametne kuće

Izvor: Autor

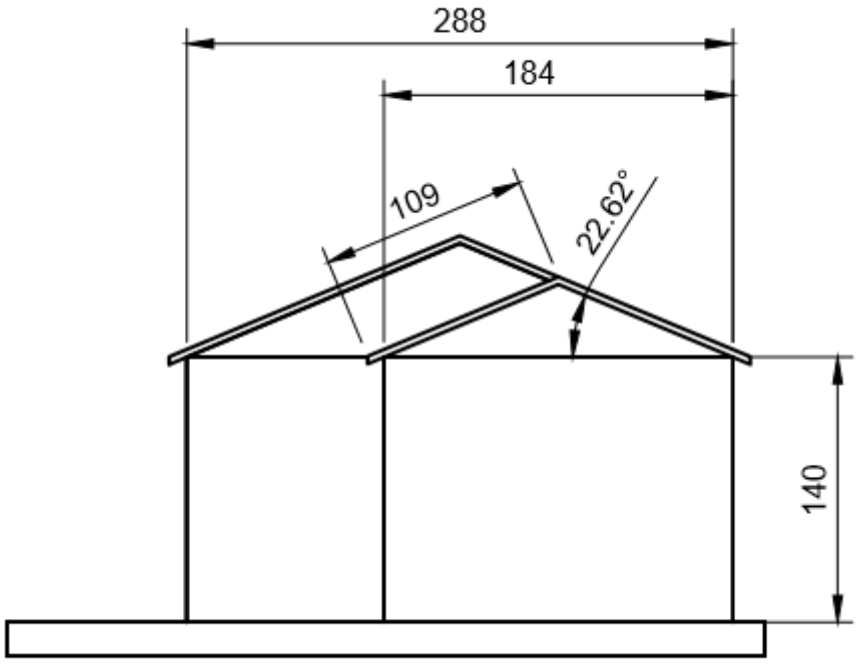
Na slici 29. je prikazan stražnji pogled na model pametne kuće i vidimo da su dimenzije širine i visine iste kao i na prednjem pogledu, razlika je ta jedino što na ovom pogledu imamo jedan prozor koji je dimenzija 50x40.



Slika 30. – Bočni pogled sa lijeve strane na model pametne kuće

Izvor: Autor

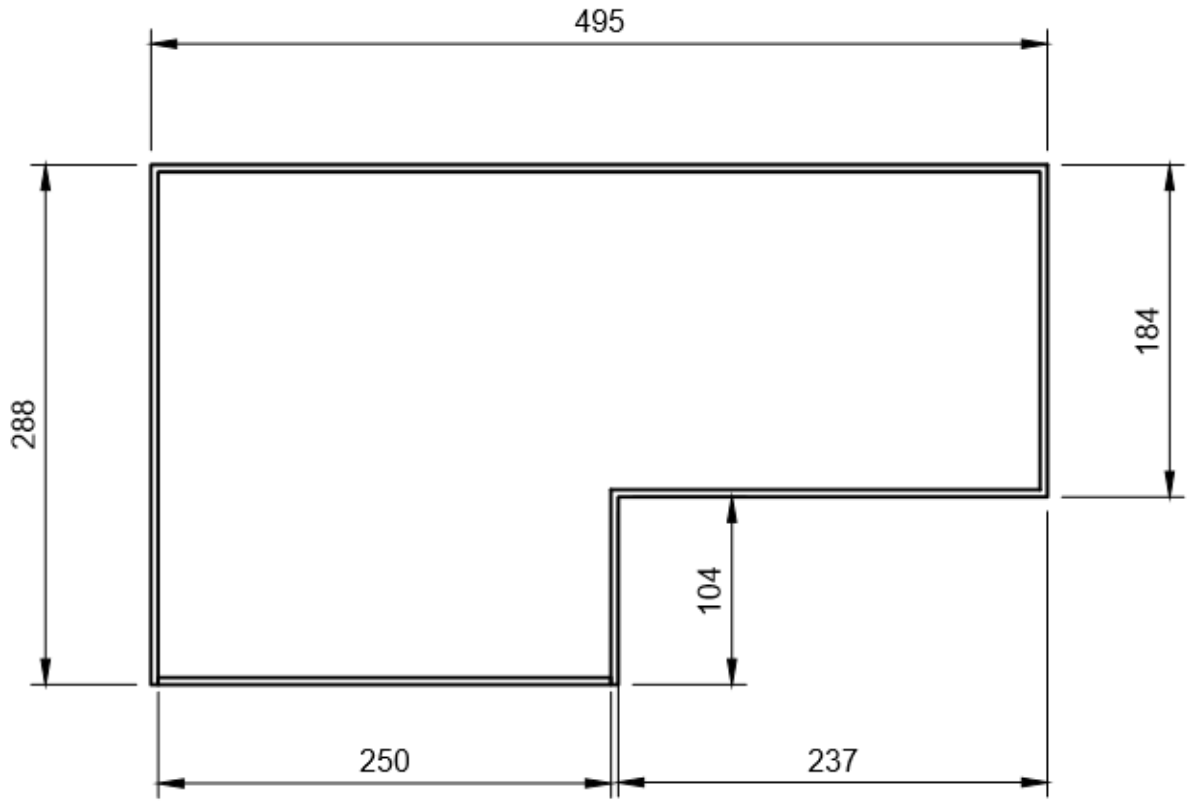
Na slici 30. vidimo bočni pogled sa lijeve strane modela pametne kuće na kojoj su prikazane dimenzije krovišta i kut pod kojim stoji krovište, te širina koja iznosi 288 mm. Prikazan je i jedan otvor za prozor dimenzija 50x40.



Slika 31. – Bočni pogled sa desne strane na model pametne kuće

Izvor: Autor

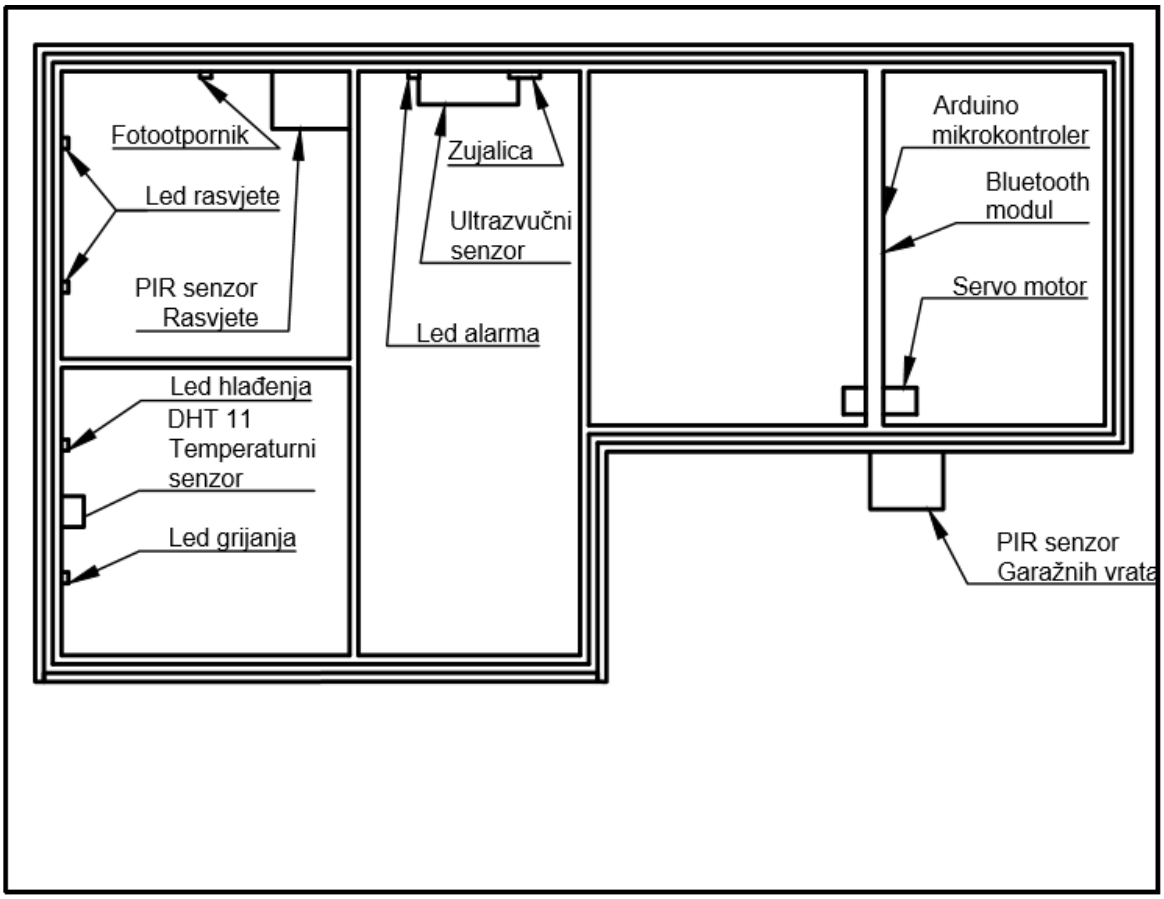
Na slici 31. je prikazan bočni pogled sa desne strane na model pametne kuće i vidimo da imamo suženje modela po širini. Suženje koje iznosi 184 mm, nam na modelu pametne kuće daje oblikovanje u oblik slova „L“. Te prikazano je i krovište suženog dijela modela koje je pod istim kutom kao i ono prethodno samo stranice krovišta su manjih dimenzija koje iznose 109 mm.



Slika 32. – Nacrt vanjskog sloja modela pametne kuće

Izvor: Autor

Slika 32. nam prikazuje vanjski sloj modela pametne kuće, tj. obrub koji su završni gabariti. Ovaj obrub je isto oblika slova „L“ sa dimenzijama uvećanim za 8 mm, kako bi dobili međuprostor gdje će se smjestiti žice za spajanje električnih komponenata za ovaj model pametne kuće.



Slika 33. – Nacrt smještaja elektroničkih komponenata na modelu pametne kuće

Izvor: Autor

Slika 33. nam prikazuje gdje su smještene elektroničke komponente koje su potrebne za funkcionalnost modela pametne kuće. Vidimo da u jednoj prostoriji imamo PIR senzor za rasvjetu te dvije led diode koje predstavljaju rasvjetno tijelo. U drugoj prostoriji imamo temperaturni senzor DHT 11, te dvije led diode od kojih jedna predstavlja led diodu za hlađenje, a druga led dioda predstavlja led diodu za grijanje. U trećoj prostoriji smješten je ultrazvučni senzor sa led diodom koja predstavlja led obavijest za alarmni sustav, te zujalica. U četvrtoj prostoriji postavljen je servo motor koji služi za otvaranje garažnih vrata. Peta prostorija nam predstavlja mozak modela pametne kuće, tj. prostor u kojem je smješten Arduino mikrokontroler i Bluetooth modul za upravljanje. Te jedan senzor koji je smješten sa vanjske strane modela pametne kuće, a to je PIR senzor koji je zadužen za otvaranje i zatvaranje garažnih vrata.

Izrada započinje ocrtavanjem linija na šperploču, pomoću kojih ćemo dobiti dijelove potrebne za sklapanje modela pametne kuće. Jedan primjer ocrtavanja možemo vidjeti na slici 34.

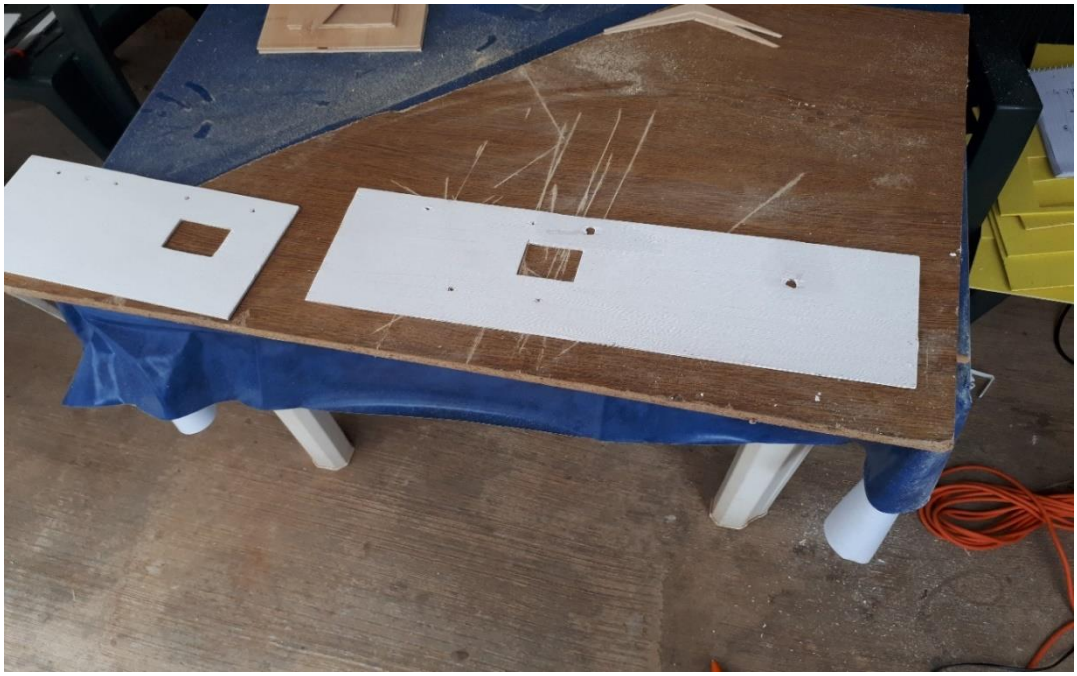
Slika 34. – Ocrtavanje linija za izrezivanje na šperploči
Izvor: Autor

Nakon ocrtavanja linija na šperploči kreće se sa piljenjem istih. Piljenje šperploče izvršava se pomoću multi funkcijskog alata kojim uz pomoć rezne oštrice odvajamo drvenu masu što je prikazano na slici 35.. Nakon obrade piljenja šperploče za finalnu obradu koristili smo brusni papir pomoću kojega smo izravnali rubove ispiljene šperploče. Za potrebe postavljanja elektroničkih komponenata bušene su rupe od 5 mm za postavljanje led dioda i provlačenje žica.

Slika 35. – Piljenje šperploče sa multifunkcijskim alatom

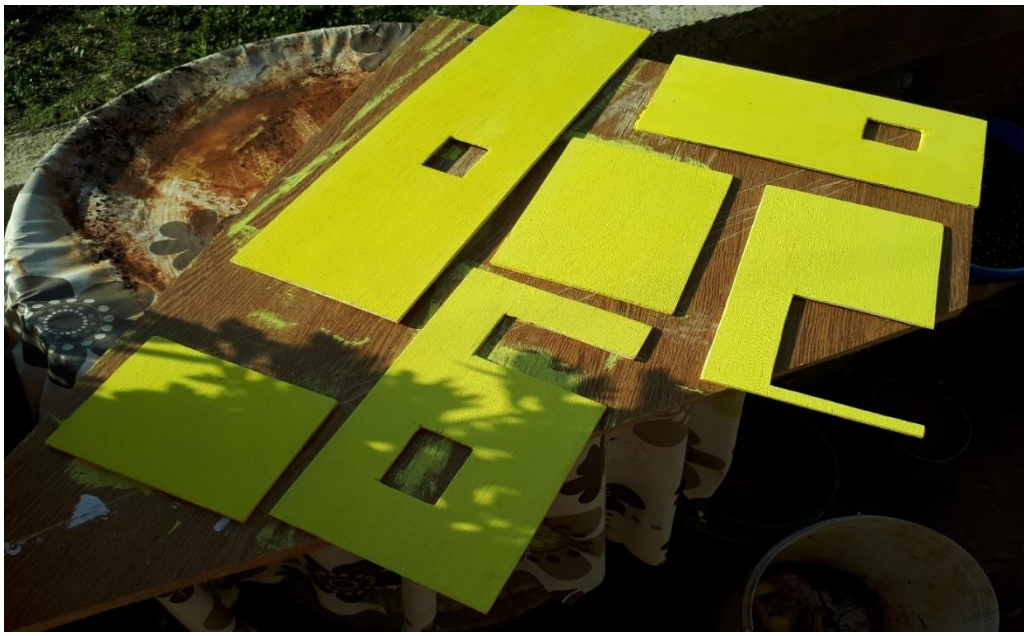
Izvor: Autor

Nakon obrade piljenjem odvija se bojanje ispiljenih šperploča, bojanje će se vršiti tako da će se s unutarnje strane modela pametne kuće koristiti bijela boja, a s vanjske strane svijetlo žuta boja kao što vidimo na slikama 36. i 37..



Slika 36. – Bojanje unutarnje strane modela pametne kuće

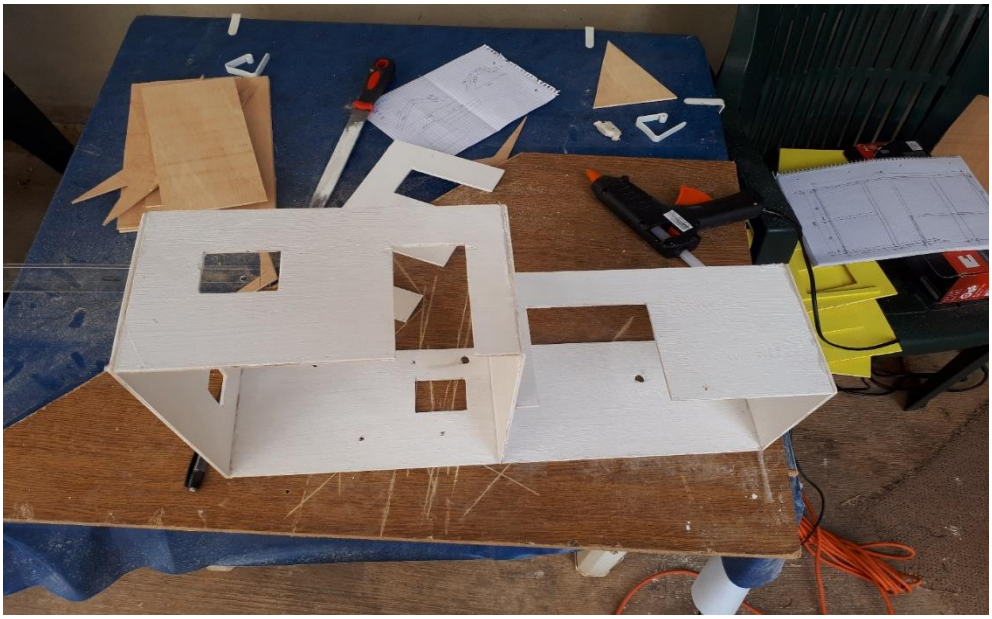
Izvor: Autor



Slika 37. – Bojanje vanjske strane modela pametne kuće

Izvor: Autor

Ploču koja nam predstavlja podlogu modela pametne kuće je u prirodnoj boji drva od kojeg je izrađena. Poslije bojanja slijedi lijepljenje šperploča sa pištoljem na vruću plastiku što nam prikazuje slika 38.. Lijepit će se odgovarajući izrezani oblici tako da tvore model pametne kuće prema konstrukcijskom nacrtu. Na mjesta gdje su ostavljeni otvori za prozore nalijepiti će se takozvano staklo pleksiglas.



Slika 38. – Lijepljenje šperploče

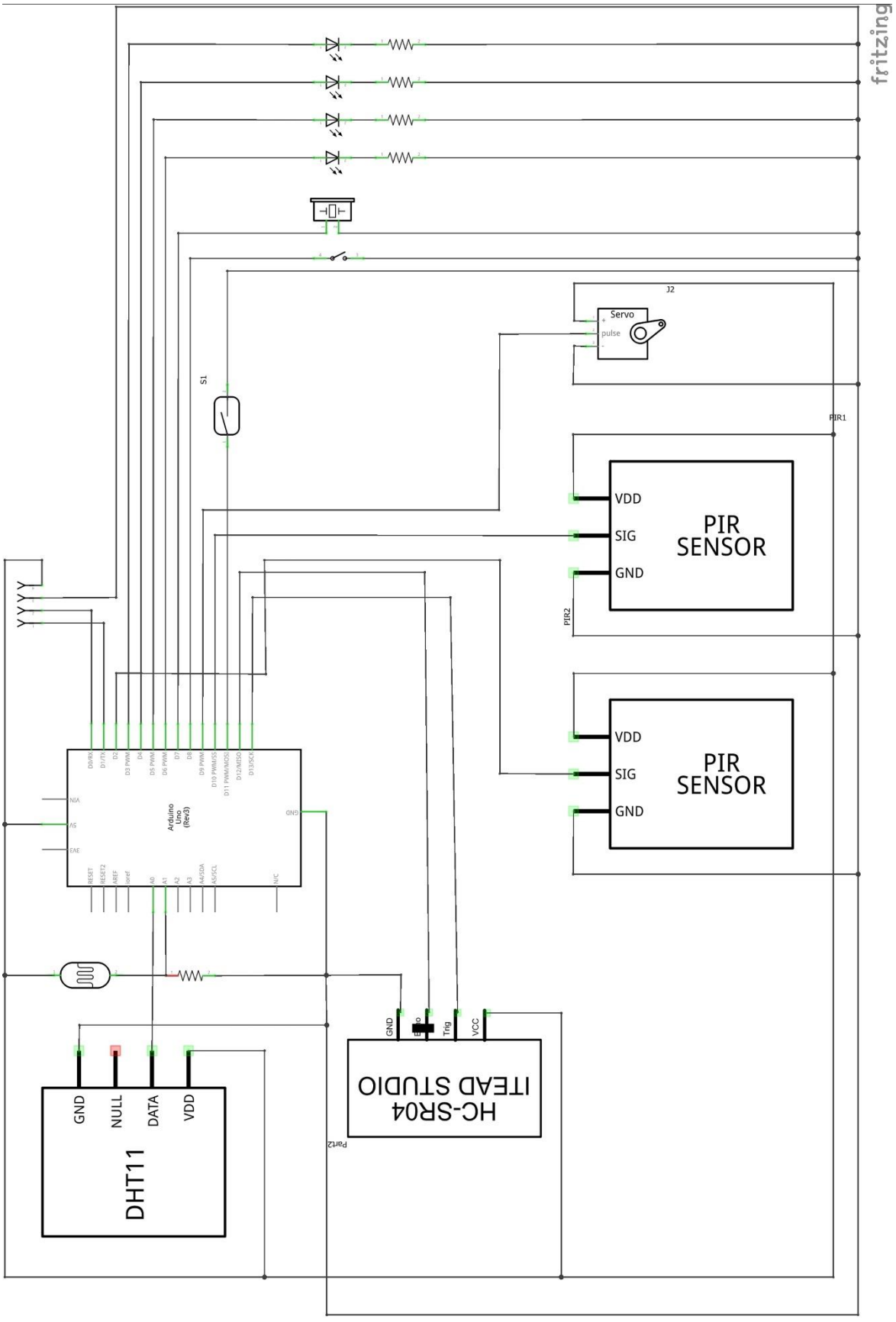
Izvor: Autor

Uspješnim lijepljenjem šperploča nastaje skoro gotov oblik modela pametne kuće koji možemo vidjeti na slici 39. Da bi dobili pomični krov koriste se dva šarnira koja pričvršćujemo sa jedne strane na oblik modela pametne kuće a drugi na krovište.

Slika 39. – Gotov unutarnji oblik modela pametne kuće

Izvor: Autor

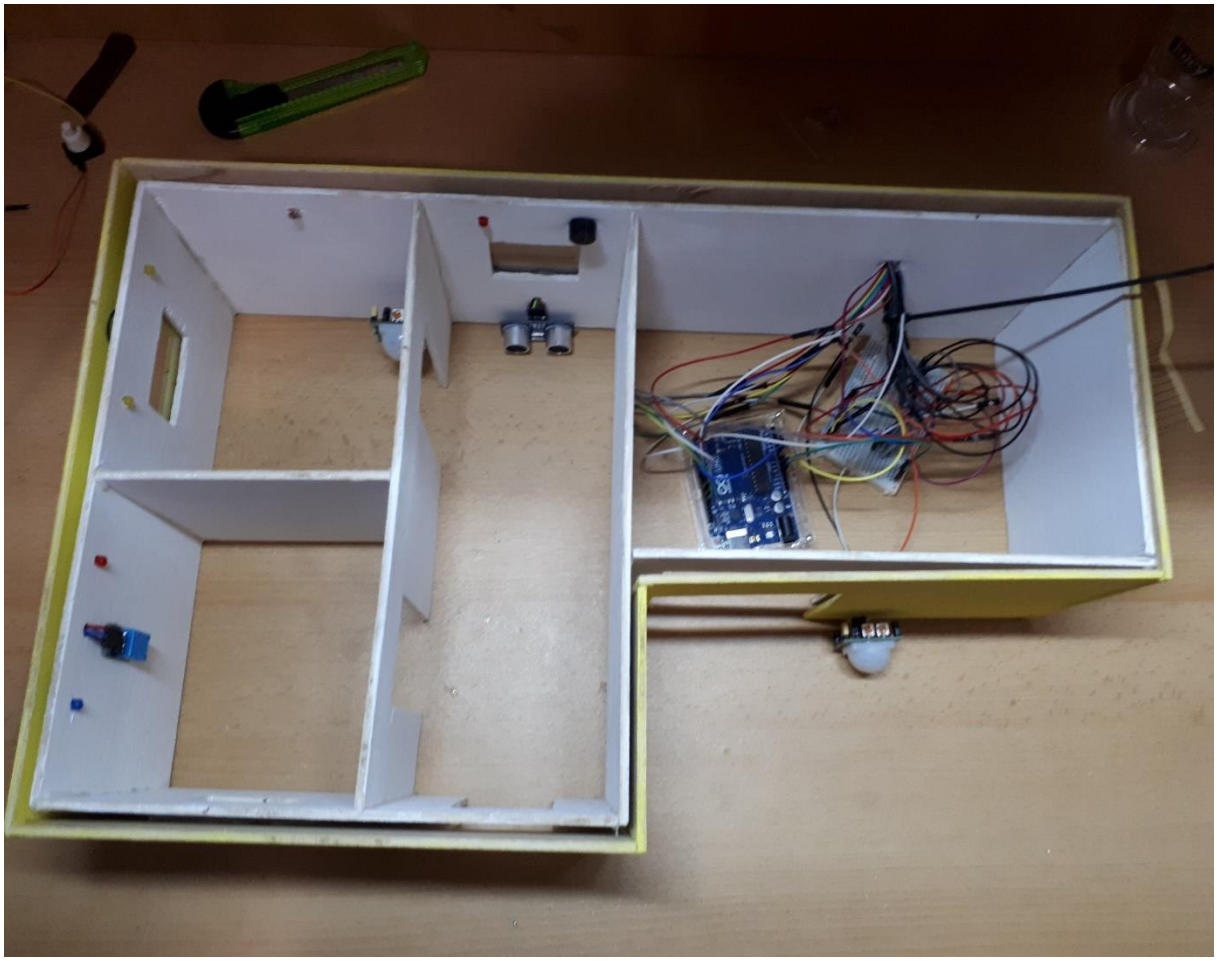
Završetkom izrade konstrukcijskog dijela unutarnjeg oblika modela makete pametne kuće kreće se sa spajanjem elektroničkog dijela prema shemi prikazanoj na slici 40.



Slika 40. – Elektronički dio modela pametne kuće

Izvor: Autor

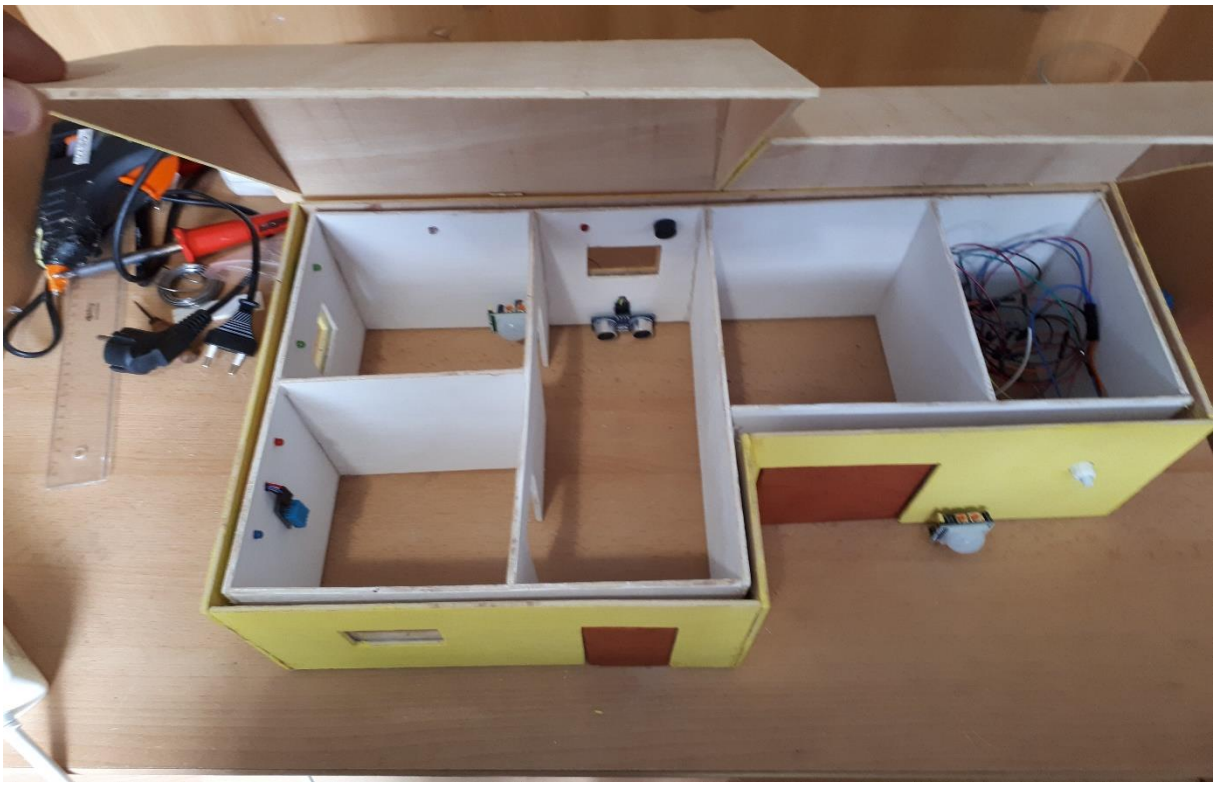
Za spajanje elektroničkih komponenata na model pametne kuće koristila se lemilica i lem kojim su se spojne žice spajale sa komponentama. Spojne žice su spajane od komponenata do eksperimentalne pločice koja je smještena unutar modela pametne kuće. Nadalje se sa spojnim žicama spajalo sa eksperimentalne pločice na sam Arduino uređaj što možemo vidjeti na slici 41.



Slika 41. – Spajanje elektroničkih komponenata na eksperimentalnu pločicu

Izvor: Autor

Kao što se vidi na slici 42. model pametne kuće izgleda kao na navedenoj slici. Sastoji se od oblika slova „L“. Ima 5 prostorija u kojima su smještene komponente potrebne za funkcionalnost, te samo krovište modela pametne kuće koje je pomično radi boljeg uvida u smještaj komponenata i samog pogleda na mogućnosti koje obavlja.



Slika 42. – Otvoren gotov model pametne kuće

Izvor: Autor

Slika 43. – Zatvoren gotov model pametne kuće

Izvor: Autor

3.3. Izrada programa za upravljanje Arduino uređajem

Za potpuno funkcioniranje modela pametne kuće potrebno je imati adekvatni program kojim se upravlja Arduino uređajem. Za izradu tog programa moguće je koristiti open source software koji se naziva Arduino IDE. Cijeli programski kod se nalazi u popisu priloga pod nazivom Prilog 1. – Program za upravljanje Arduino uređajem.

Arduino IDE[42] je open source program koji nam omogućava pisanje programskog djela potrebnog za upravljanje Arduino uređajem. To je program koji ima svoja tri glavna dijela u koja upisujemo željeni program koji će Arduino izvršavati. Prvi dio programa je onaj u kojega pišemo koje komponente su nam spojene na koji priključak na Arduino uređaju, i dodjeljujemo varijable koje su nam potrebne za neka očitanja. Drugi dio, koji se naziva „Void setup“ je dio u kojem se dodjeljuju stanja priključaka (ovisno o potrebi postavljaju se stanja u logičke „0“ ili „1“), te u tom dijelu se prijašnje dodijeljenim komponentama postavlja uloga koju će izvršavati (ovisno o komponenti postavljaju se u ulazne ili izlazne). Treći i najvažniji dio se naziva glavna petlja (vodi loop). Unutar glavne petlje se izvršava željeni kod. Kod se izvršava beskonačan broj puta, dokle god to netko ne ograniči.

```
#include <Servo.h>                    // uključi library za servo motror
#include <dht.h>                      // uključi library za temperaturni senzor
#define dht_dpin A0                   // spoji temperaturni senzor na analogni ulaz A0
dht DHT;                              // potrebno je defiinirat zbog librarya
Servo servo;                          // potrebno je defiinirat zbog librarya
int sklopka = 8;                      // sklopka je spojena na priključak 8
int poz = 0;                          // varijabla za for petlju za upravljanje servo motorom
int foto = A1;                        // foto otpornik spojen na priključak A1
int ocitanjefo;                       // varijabla za spremanje očitanja na fotootporniku
int jacinaled;                        // varijabla za pretvorbu očitanja na fotootporniku u jačinu led diode
int led_rasvjeta =3;                  // led_rasvjeta je spojena na priključak 3
int led_grijanje =4;                  // led_grijanje je spojena na priključak 4
int led_hladenje = 5;                 // led_hlađenje je spojena na priključak 5
int led_alarm = 6;                    // led_alarm je spojena na priključak 6
int zuj = 7;                          // zujalica je spojena na priključak 7
int echo = 12;                        // echo priključak sa ultrazvučnog modula je spojen na priključak 12
int trig = 13;                        // trig priključak sa ultrazvučnog modula je spojen na priključak 13
int ocitanjeUZV = 0;                  // varijabla za spremanje očitanja na ultrazvučnom modulu
int udaljenost;                       // varijabla za pretvorbu ocitanja na ultrazvučnom modulu u udaljenost od objekta
const int inRasvjeta = 10;            // PIR senzor rasvjete spojen na priključak 10
const int PIRgaraza = 2;              // PIR senzor garaznihvrata spojen na priključak 2
int ocitanjeIN;                       // varijabla za spremanje očitanja na PIR senzoru rasvjete
int ocitanjeG;                        // varijabla za spremanje očitanja na PIR senzoru garaznih vrata
char data = 0;                        // varijabla data za spremanje bluetooth podataka
int rasvjeta = 0;                     // varijabla za spremanje stanja rasvjete
int garaza = 0;                       // varijabla za spremanje stanja garaznih vrata
int alarm;                            // varijabla za spremanje stanja alarma
int temperatura;                      // varijabla za spremanje ocitanja temperature
```

Slika 44. – Početni dio programa

Izvor: Autor

[42] https://www.arduino.cc/en/Main/Software (31.08.2107.)

Kao što vidimo na slici 44. na početku programa za upravljanje modelom pametne kuće potrebno je uključiti biblioteke (library) za upravljanje servo motorom i DHT 11 temperaturnim senzorom. Biblioteke se koriste stoga što je unutar njih već predefinirano što će senzor izvršavati pa to nije potrebno pisati u samom kodu programa. Nakon što je uključen library za Servo motor i DHT 11 temperaturni senzor, potrebno je definirati na koji priključak će se spojiti gore naveden DHT 11 temperaturni senzor što se izvršava sa naredbom *#define dht_dpin A0,* što znači da smo spojili temperaturni senzor na analogni priključak broj 0. Za Servo motor definira se varijabla servo. Nakon toga potrebno je definirati na koje priključke će biti spojene led diode, sklopka, PIR senzori, ultrazvučni senzor i zujalica. Za definiranje spajanja ovih komponenti koristi se naredba *int naziv_komponente = broj priključka;*. Za korištenje serijske komunikacije definirala se varijabla *data* koja je tipa *char*. Za spremanje očitanja sa senzora definirale su se i ostale varijable funkcijom *int naziv_očitanja;*. Te time završava prvi dio ovog programa.

```
void setup() {
  servo.attach(9);                          // spajanje servo motora na priključak 9
  pinMode(led_rasvjeta,OUTPUT);             // definiranje led rasvjete kao izlazni signal
  pinMode(led_grijanje,OUTPUT);             // definiranje led grijanja kao izlazni signal
  pinMode(led_hladenje,OUTPUT);             // definiranje led hlađenja kao izlazni signal
  pinMode(led_alarm,OUTPUT);                // definiranje led alarma kao izlazni signal
  pinMode(inRasvjeta,INPUT);                // definiranje PIR senzora rasvjete kao ulazni signal
  pinMode(PIRgaraza,INPUT);                 // definiranje PIR senzora gražnih vrata kao ulazni signal
  pinMode(sklopka,INPUT_PULLUP);            // definiranje sklopke kao ulazni signal
  pinMode(zuj,OUTPUT);                      // definiranje zujalice kao izlazni signal
  pinMode(trig,OUTPUT);                     // definiranje trig kao izlazni signal
  pinMode(echo,INPUT);                      // definiranje echo kao ulazni signal
  digitalWrite(led_alarm,LOW);              // postavljanje led alarm u nisko stanje
  digitalWrite(led_rasvjeta,LOW);           // postavljanje led rasvjeta u nisko stanje
  digitalWrite(led_grijanje,LOW);           // postavljanje led grijanje u nisko stanje
  digitalWrite(led_hladenje,LOW);           // postavljanje led hlađenje u nisko stanje
  servo.write(0);                           // servo motor se postavlja u 0
  Serial.begin(9600);                       // započni serijsku komunikaciju na frekvenciji od 9600
}
```

Slika 45. – Drugi dio programa (Void setup)

Izvor: Autor

U drugom dijelu programa koji se naziva „Void setup“, koji se prikazan na slici 45.. Potrebno je definirati na koji priključak je spojen Servo motor, a to se izvršava naredbom *servo.attach(broj priključka);*. Nakon toga potrebno je konfigurirati sve komponente koje su spojene na Arduino uređaj, odnosno postaviti ih kao ulazne ili izlazne signale. Ovu konfiguraciju radimo pomoću naredbe *pinMode(naziv_komponente, INPUT/OUTPUT);*. Nakon konfiguriranih izlaznih / ulaznih signala, za potrebe modela pametne kuće sve led diode se postavljaju u stanje „0“ ili nisko stanje, a izvršava se naredbom *digitalWrite(naziv_komponente, stanje);*, te se servo motor postavlja u početno stanje 0. Na

kraju ovog dijela programa potrebno je započeti serijsku komunikaciju što se čini sa naredbom *Serial.begin(frekvencija);*, ovom naredbom se smatra da priključci Rx (priključak 0) i Tx (priključak 1) služe za komunikaciju i ne mogu se koristiti u druge svrhe.

Treći dio programa se naziva „Void loop“ ili na Hrvatski prevedemo glavna petlja, u njoj se beskonačno puta izvršavaju zadane naredbe unutar nje. Slika 46. Nam prikazuje početak koda u glavnoj petlji.

```
void loop() {
  //ALARMNI SUSTAV
  digitalWrite(trig,HIGH);                    // postavi priključak trig u visoko stanje
  delayMicroseconds(10);                      // čekaj 10 mikro sekundi
  digitalWrite(trig,LOW);                     // postavi priključak trig u nisko stanje
  delayMicroseconds(5);                       // čekaj 5 mikro sekundi
  ocitanjeUZV = pulseIn(echo,HIGH);           // primanje reflektivnog triga i spremanje u varijablu ocitanjeUZV
  udaljenost = (ocitanjeUZV/2)/29.1;          // pretvorba reflektivnog triga u udaljenost u cm
  delay(500);                                 // čekaj 0,5 sekunde za stabilizaciju senzora
  if(data=='8'){                              // ako je varijabla data = 8
    alarm = 1;                                // postavi varijablu alarm u 1
    if((udaljenost<20)&&(udaljenost>0)){      // ako je udaljenost manja od 19 cm i veća od 0
      tone(zuj,1000);                         // postavi zujalicu da pišti
      digitalWrite(led_alarm,HIGH);           // postavi led_alarm u visoko stanje
      delay(500);                             // čekaj 0,5 sekunde
      digitalWrite(led_alarm,LOW);            // postavi led alarm u nisko stanje
      delay(500);                             // čekaj 0,5 sekunde
      digitalWrite(led_alarm,HIGH);           // postavi led alarm u visoko stanje
      }
      }
      else if(data == '9'){                   // inače ako je varijabla data = 9
        alarm = 0;                            // postavi varijablu alarm u 0
        digitalWrite(led_alarm,LOW);          // postavi led alarm u nisko stanje
        noTone(zuj);                          // ugasi pištanje zujalice
        }
```

Slika 46. – Početak glavne petlje

Izvor: Autor

U ovom programu definirano je izvršavanje koda za alarmni sustav pametne kuće, u kojem se koristi ultrazvučni modul. Za definiranje koda ultrazvučnog modula koristi se više naredba. Naredbe koje zadaju izvršavanje očitanja ultrazvučnog modula su te da na početku Arduino šalje visoki nivo napona na *trig* priključak ultrazvučnog modula, čeka 10 mikrosekundi zbog toga što toliko treba signalu sa ultrazvučnog modula da se vrati nazad, te onda ponovno postavlja *trig* priključak u nisko stanje, te čeka još 5 mikrosekundi. Ako postoji neko očitanje na ultrazvučnom senzoru pomoću naredbe *naziv_očitanja = pusleIn(echo, HIGH);* se ono vraća u Arduino uređaj i sprema u neko od predefiniranih očitanja. Ako postoji očitanje na ultrazvučnom senzoru ono se sprema u jednu od varijabli, te nakon toga se ta varijabla pretvara u ovom slučaju poželjnu mjernu jedinicu cm. Nadalje imamo serijsku komunikaciju koja slanjem određene naredbe provjerava ako je zadovoljen uvjet koji nam onda uključuje alarmni sustav. Dalje ako pošaljemo drugu naredbu taj isti alarmni sustav se isključuje.

```
// AUTOMATSKI NAČIN RADA
if(digitalRead(sklopka)==LOW){                  // ako je sklopka u niskom stanju izvršava se automatski režim rada
  //AUTOMATSKO UPRAVLJANJE GARAŽNIM VRATIMA
  ocitanjeG = digitalRead(PIRgaraza);           // očitano stanje na PIR senzoru garažnih vrata postavi u varijablu ocitanj
  if(ocitanjeG == HIGH){                        // ako je očitanje na PIR senzoru garažnih vrata u visokom stanju
    if(garaza == 0){                            // ako je stanje garaza = 0
      garaza = 1;                               // postavi varijablu garaza u 1
      for(poz = 0; poz < 90; poz +=1){          // OTVARA GARAŽNA VRATA
      servo.write(poz);
      delay(15);
    }
    }
    else{                                       // inače
      garaza= 0;                                // garaza je 0
      for(poz = 90; poz >=1; poz -=1){          // ZATVARA GARAŽNA VRATA
      servo.write(poz);
      delay(15);
    }
    }
    }
    // AUTOMATSKO UPRAVLJANJE RASVJETOM
 ocitanjefo = analogRead(foto);                 // očitano stanje na fotootporniku postavi u varijablu ocitanjefo
 jacinaled = map(ocitanjefo, 0,1023, 255,0);    // pretvorba stanja na fotootporniku u jacinu led diode
 ocitanjeIN = digitalRead(inRasvjeta);          // očitano stanje na PIR senzoru rasvjete postavi u varijablu ocitanjeIN
  if(ocitanjeIN == HIGH){                       // ako je varijabla ocitanjeIN u visokom stanju
    if(rasvjeta==0){                            // ako je rasvjeta = 0
      rasvjeta = 1;                             // postavi varijablu rasvjeta u 1
      analogWrite(led_rasvjeta,jacinaled);      // postavi led rasvjeta u stanje ovisno o pretvorenom očitanju sa fotootpornika
      delay(3000);                              // čekaj 3 sekunde radi delaya PIR senzora
    }
    else {                                      // inače
      rasvjeta = 0;                             // rasvjeta = 0
      analogWrite(led_rasvjeta,0);              // postavi led rasvjetu u 0
      delay(3000);                              // čekaj 3 sekunde radi delaya PIR senzora
    }
  }
  // AUTOMATSKO UPRAVLJANJE KLIMATIZACIJOM
 DHT.read11(dht_dpin);                          // očitaj stanje na temperaturnom senzoru
 temperatura = DHT.temperature;
 if(DHT.temperature > 25){                      // ako je očitana temperatura veća od 25 stupnjeva
  digitalWrite(led_hladenje,HIGH);              // postavi led hlađenje u visoko stanje
  digitalWrite(led_grijanje,LOW);               // postavi led grijanje u nisko stanje
    }
 else if(DHT.temperature < 20){                 // inače ako je očitana temperatura manja od 20 stupnjeva
  digitalWrite(led_grijanje,HIGH);              // postavi led grijanje u visoko stanje
  digitalWrite(led_hladenje,LOW);               // postavi led hlađenje u nisko stanje
    }
```

Slika 47. – Programski dio za automatski način rada

Izvor: Autor

Slika 47. prikazuje program za automatski način rada modela pametne kuće. Program je osmišljen tako da ovisno o stanju sklopke (ručno/automatski) izvršava automatski način rada modela pametne kuće. Što je i vidljivo iz naredbe *if(digitalRead(sklopka)==LOW);*. Program nakon što dobije očitanje da je sklopka u stanju „LOW“ izvršava ostatak koda u tom bloku. Program je napisan tako da kad je sklopka u niskom stanju svi senzori očitavaju stanja u okolini. Prvi od senzora je PIR senzor garažnih vrata koji uz naredbu *ocitanjeG=digitalRead(PIRgaraza);* očitava stanje na PIR senzoru zaduženom za garažna vrata i očitanje sprema u varijablu. Nadalje ako je očitanje na PIR senzoru garažnih vrata u visokom stanju, pregledava ako je zadana varijabla u stanju 0, te ako je zadana varijabla u stanju 0, u varijablu garaza se upisuje 1, te se pokreće Servo motor. Nakon pokretanja Servo motora i postavljanja u zadanu poziciju Servo motor miruje. Ako od prije opisana varijabla nije u stanju

0 , u varijablu garaza se upisuje 0, te se Servo motor vraća u početnu poziciju. Nakon upravljanja garažnim vratima dolazi kod za upravljanje rasvjetom i on započinje tako što naredbom *ocitanjefo = digitalRead(foto);*, očitavamo vrijednost na foto otporniku i spremamo je u varijablu *ocitanjefo*. Kada očitamo vrijednost na foto otporniku potrebno ju je skalirati da bi se moglo postavljati led diodu u stanje ovisno o očitanju, a to nam određuje naredba *jacinaled = map (ocitanjefo, 0, 1023, 255, 0);*. Drugi senzor koji izvršava program je PIR senzor zadužen za rasvjetu, te funkcionira sa naredbom *ocitanjeIN = digitalRead(inRasvjeta);*. Program provjerava dali je očitanje na senzoru u visokom stanju, ako je u visokom stanju provjerava dali je varijabla stanje u 0. Ako je varijabla stanje jednaka 0, u varijablu stanje se upisuje 1 i varijabla *led_rasvjeta* se postavlja ovisno o pretvorenom očitanju na foto otporniku. Inače ako varijabla *stanje* je jednaka 0, varijabla *led_rasvjeta* se postavlja u nisko stanje. Treći senzor je DHT11 temperaturni senzor koji uz naredbu *DHT.read11(dht_dpin);* počinje mjeriti temperaturu u svojoj okolini. Program ovisno o zadanoj temperaturi varijablama *led_grijanje* i *led_hladenje* visoko stanje, odnosno nisko stanje.

```
 // SLANJE OČITANJA NA ANDROID UREĐAJ
if(rasvjeta == 1){                          // ako je varijabla stanje u 1
 Serial.print("Uključena rasvjeta");        // serijskom komunikacijom šalji da je rasvjeta upaljena
 Serial.print("|");                         // potrebno radi razdvajanja i pozicioniranja teksta na android uređaju
   }
else if(rasvjeta ==0){                      // inače ako je stanje u 0
 Serial.print("Isključena rasvjeta");       // serijskom komunikacijom šalji da je rasvjeta ugašena
 Serial.print("|");                         // potrebno radi razdvajanja i pozicioniranja teksta na android uređaju
   }
 Serial.print(temperatura);                 // serijskom komunikacijom šalji očitanje temperaturnog otpornika
 Serial.print("°C");                        // serijskom komunikacijom šalji mjernu jedinicu
 Serial.print("|");                         // potrebno radi razdvajanja i pozicioniranja teksta na android uređaju

if(garaza==1){                              // ako je varijabla garaza u 1
 Serial.print("Otvorena garažna vrata");    // serijskom komunikacijom šalji da su garažna vrata otovrena
 Serial.print("|");                         // potrebno radi razdvajanja i pozicioniranja teksta na android uređaju
   }
else if(garaza==0){                         // inače ako je varijabla garaza u 0
 Serial.print("Zatvorena garažna vrata");   // serijskom komunikacijom šalji da su garažna vrata zatvorena
 Serial.print("|");                         // potrebno radi razdvajanja i pozicioniranja teksta na android uređaju
   }
if(alarm==1){                               // ako je varijabla alarm u 1
 Serial.print("Aktiviran alarm");           // serijskom komunikacijom šalji da je alarm upaljen
 Serial.print("|");                         // potrebno radi razdvajanja i pozicioniranja teksta na android uređaju
   }
else if(alarm == 0){                        // inače ako je varijabla alarm u 0
  Serial.print("Deaktiviran alarm");         // serijskom komunikacijom šalji da je alarm ugašen
  Serial.print("|");                         // potrebno radi razdvajanja i pozicioniranja teksta na android uređaju
   }
  delay(3000);                              // čeka se 3 sekunde radi slanja očitanja svakih 3 sekunde
```

Slika 48. – Slanje očitanja senzora serijskom komunikacijom

Izvor: Autor

U prvom dijelu programa dodijeli smo varijable *rasvjeta* za očitanje stanja PIR senzora zaduženog za rasvjetu, varijable *garaza* za očitanje stanja PIR senzora zaduženog za garažna vrata i varijable *alarm* zaduženu za provjeru stanja alarmnog sustava i za očitanje temperature varijablu temperatura. Ovisno o varijabli *rasvjeta* pomoću serijske komunikacije se ispisuje stanje rasvjete, odnosno dali je uključena ili isključena rasvjeta. Pomoću varijable *garaza* ispisuje se pomoću serijske komunikacije stanje garažnih vrata (Otvorena ili Zatvorena). Varijabla *alarm* nam predstavlja stanje alarma koje može biti ili aktivirano ili deaktivirano. Što se tiče temperature naredbom *Serial.print (temperatura);* i *Serial.print (' °C');*, šalje se serijskom komunikacijom. Slika 48. prikazuje programski kod za slanje očitanja na Android uređaj.

```
if(Serial.available()>0){                    // ako postoji serijska komunikacija i veća je od 0
data = Serial.read();                        // u varijablu data spremaj podatke primljene serijskom komunikacijom

// RUČNI REŽIM RADA
if(digitalRead(sklopka)==HIGH){              // ako je sklopka u visokom stanju izvršava se ručni režim rada
 // RUČNO UPRAVLJANJE RASVJETOM
ocitanjefo = analogRead(foto);               // očitano stanje na fotootporniku postavi u varijablu ocitanjefo
jacinaled = map(ocitanjefo, 0,1023, 255,0);  // pretvorba stanja na fotootporniku u jacinu led diode
if(data == '1'){                             // ako je varijabla data = 1
   analogWrite(led_rasvjeta,jacinaled);      // postavi led rasvjeta u stanje ovisno o pretvorenom očitanju sa fotootpornika
 }
 else if (data == '0'){                      // inače ako je varijabla data = 0
   analogWrite(led_rasvjeta,0);              // postavi led rasvjeta u 0
 }
 // RUČNO UPARAVLJANJE KLIMATIZACIJOM
 if(data == '2'){                            // ako je varijabla data = 2
   DHT.read11(dht_dpin);                     // očitaj stanje na temperaturnom senzoru
   digitalWrite(led_hladenje,HIGH);          // postavi led hlađenje u visoko stanje
   digitalWrite(led_grijanje,LOW);           // postavi led hladenje u nisko stanje
 }
 else if (data == '3'){                      // inače ako je varijabla data = 3
   digitalWrite(led_hladenje,LOW);           // postavi led hlađenje u nisko stanje
 }
 if(data == '4'){                            // ako je varijabla data = 4
   DHT.read11(dht_dpin);                     // očitaj stanje na temperaturnom senzoru
   digitalWrite(led_grijanje,HIGH);          // postavi led grijanje u visoko stanje
   digitalWrite(led_hladenje,LOW);           // postavi led hladenje u nisko stanje
 }
 else if (data == '5'){                      // inače ako je varijabla data = 5
   digitalWrite(led_grijanje,LOW);           // postavi led grijanje u nisko stanje
 }
 // RUČNO UPRAVLJANJE GARAŽNIM VRATIMA
 if(data == '6'){                            // ako je varijabla data = 6
   for(poz = 0; poz < 90; poz +=1){          // OTVORI GARAŽNA VRATA
     servo.write(poz);
     delay(15);
   }
 }
 else if (data == '7'){                      // inače ako je varijabla data = 7
   for(poz = 90; poz >=1; poz -=1){          // ZATVORI GARAŽNA VRATA
     servo.write(poz);
     delay(15);
   }
 }
}
```

Slika 49. – Programski dio za ručni način rada

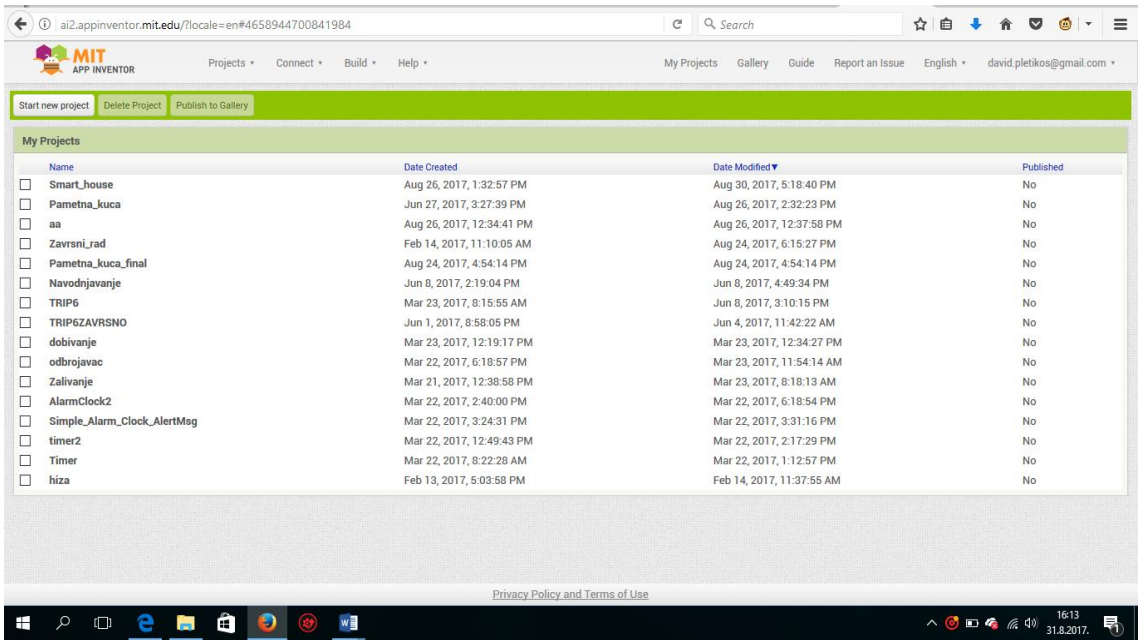
Izvor: Autor

Slika 49. Nam prikazuje program za ručni način rada modela pametne kuće. Program je osmišljen tako da ovisno o stanju sklopke program zna dali treba izvršavati ručni način rada modela pametne kuće. Što je i vidljivo iz naredbe *if(digitalRead(sklopka)==HIGH);*. Program nakon što dobije očitanje da je sklopka u stanju „HIGH“ izvršava ostatak koda u tom bloku. Program je napisan tako da kad je sklopka u viskom svi senzori miruju osim ultrazvučnog senzora. U ovom dijelu program čita dali postoji dolazeća naredba preko serijske komunikacije i sprema podatke u varijablu *data* sljedećom naredbom *data = Serial.read();*. Nakon što spremi pristigle podatke u varijablu *data* program izvršava blokovske naredbe ovisno o podatcima u varijabli. Tako što je zadano za koju varijablu će izvršiti koju naredbu. A te naredbe su uključivanje i isključivanje rasvjete, otvaranje i zatvaranje garažnih vrata, te uključivanje i isključivanje sustava hlađenja i grijanja.

3.4. Izrada aplikacije za upravljanje pomoću Android uređaja

Da bi se upravljalo sustavom modela pametne kuće preko mobilnog uređaja, potrebno je izraditi mobilnu aplikaciju koja će preko *Bluetooth* modula komunicirati s *Arduino* razvojnom pločicom. U ovom radu za izradu aplikacije koristi se razvojno okruženje *App Inventor*. Način izrade mobilne aplikacije potrebne za ovaj rad opisan je u nastavku.

App Inventor je open source web aplikacija koja je razvijena sa strane istraživačkog sveučilišta MIT za multinacionalnu korporaciju Google i nosi naziv MIT App Inventor 2. App Inventor ko programski jezik koristi JAVU. Programiranje je napravljeno jednostavno i olakšano pomoću korištenja blokova koji predstavljaju određeni programski kod. Ovaj alat je namijenjen svima i nije potrebno znanje programskog jezika za njegovo korištenje. Alat se sastoji od dva dijela: dizajnerski i blokovski. U dizajnerskom dijelu se kreira vizualni izgled aplikacije. Komponente na dizajnerskom dijelu mogu biti vidljive ili nevidljive. U blokovskom dijelu stvara se programska logika pomoću blokova. Blokovi određuju kako se komponente trebaju ponašati prilikom neke radnje s njima. Pristup alatu se obavlja prijavom ili registriranjem vlastitog Google korisničkog računa.[43] Slika 50. predstavlja razvojno okruženje open source web aplikacije MIT App Inventor.

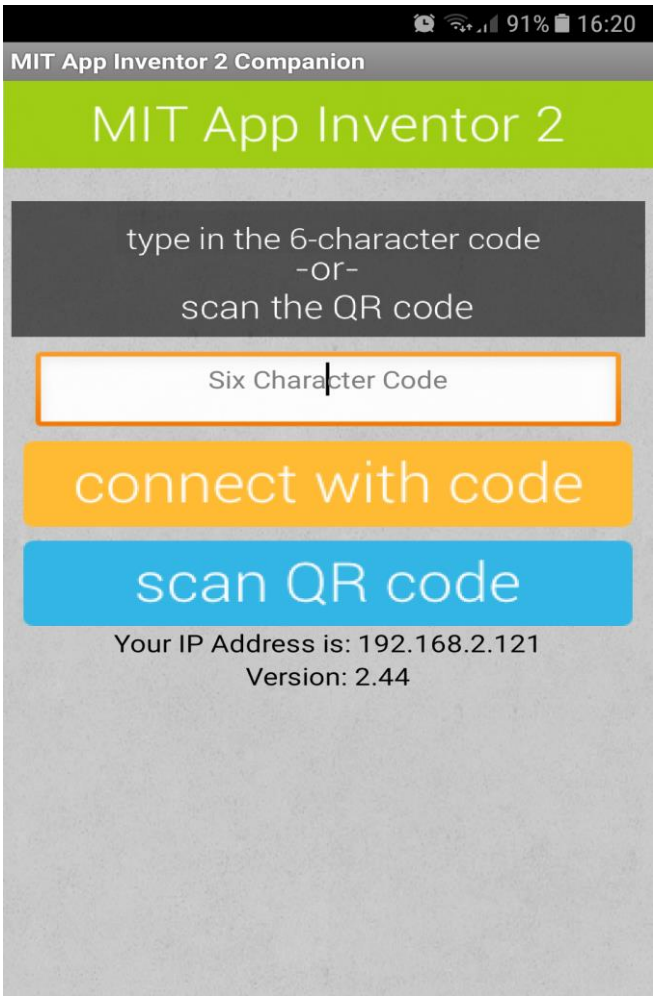


Slika 50. – MIT App Inventor razvojno okruženje

Izvor: http://ai2.appinventor.mit.edu/?locale=en#4658944700841984 (31.08.2017.)

[43] http://appinventor.mit.edu/explore/(31.08.2017.)

App Inventor ima mogućnost testiranja same aplikacije u toku izrade. Testiranje se vrši pritiskom na karticu Connect, nakon čega imamo opciju birati između tri opcije. Najjednostavniji način testiranja je pomoću aplikacije za Android mobilne uređaje koja je dostupna na Google Play trgovini pod nazivom „MIT AI2 Companion“ potpuno besplatno, a početna stranica aplikacije je prikazana na slici 51.. Nakon što se odabere način testiranja pomoću AI Companion aplikacije App Inventor dodjeljuje jedinstveni kod ili QR kod koji se upisuju ili skenira u aplikaciju na mobilnom uređaju. Ostala dva načina testiranja su putem Android emulatora na osobnom računalu, koji se kao software na računalu ponaša isto kao i mobilni uređaj, te opcija testiranja pomoću USB kabela.



Slika 51. – MIT AI2 Companion aplikacija za testiranje

Izvor: Autor

3.4.1. Izrada dizajna aplikacije

Postupak izrade aplikacije za upravljanje modelom pametne kuće počinje sa definiranjem samog izgleda aplikacije. Raspoređivanjem vidljivih i nevidljivih komponenata te definiranje njihovih pozicija na zaslonu Android uređaja uređuje se unutar dizajnerskog dijela MIT App Inventora. Sa lijeve strane imamo izbornik u kojemu se nalaze sve moguće komponente koje možemo koristiti prilikom izrade dizajna aplikacije. Za izradu dizajna aplikacije korištene su opcije iz *User interface* sekcije: gumbi, tekstualne oznake, odabirač liste, prozor za tekst lozinke. Još su se koristile opcije iz *Layout* sekcije za grupiranje i urednije postavljanje komponenti, iz *Sensors* sekcije sat komponentu, iz *Storage* sekcije *TinyDB* za spremanje lozinka, iz *Connectivity* sekcije *BluetoothClient* za spajanje na Bluetooth i *ActivityStarter* za brzo uključivanje Bluetootha.

Slika 52. – Početni zaslon aplikacije

Izvor: Autor

Slika 52. prikazuje početni zaslon aplikacije na kojem se nazale dva odabira prvi je spajanje na Bluetooth a drugi je opcija od spajanja sa Bluetootha. Ispod ta dva odabira nalazi se jedna tekstualna oznaka koja se ovisno o stanju spojenosti na Bluetooth postavlja u njihovo stanje odnosno „Spojen na Bluetooth“ ili „Nema spojenih uređaja“. Ispod toga nalaze se podatci o autoru aplikacije, ustanove za koju je izrađena i u okviru čega je napravljena. Ti podatci se prikazuju samo kada nema spojenih uređaja.

Slika 53. – Odabir načina rada

Izvor: Autor

Nakon uspješnog spajanja sa Bluetooth uređajem, tekst u tekstualnoj oznaci se promijeni u „Spojeno!“, te se pojavi opcija biranja između dva načina rada. Jedna opcija je automatski način rada, a druga je ručni način rada. Te to je prikazano na slici 53..

Slika 54. – Automatski način rad

Izvor: Autor

Nakon odabira automatskog načina rada otvara nam se novi prozor koji nam nudi mogućnosti u automatskom načinu rada modela pametne kuće kojeg vidimo na slici 54.. Vidimo prikazana nam je tekstualna oznaka u kojem načinu rada se nalazimo, te ispod nje se nalaze tekstualni okviri u kojima se prikazuju očitanja izmjerena na senzorima. Ispod tekstualnih okvira nalazi se opcija upravljanja alarmnim sustavom. Mogućnosti koje nam nudi ova opcija su sljedeće: pritiskom na opciju promjeni pojavljuje se još jedan okvir u koji upisujemo staru lozinku i jedan okvir u kojeg upisujemo novu lozinku, te još jednim klikom na gumb promjeni spremamo lozinku. Pritiskom na opciju reset, lozinka se resetira i postavlja u pred definiranu lozinku „admin“. U okvir za upisivanje lozinke upisuje se lozinka kojom možemo pritiskom na jedan od dva gumba (Aktiviraj alarm ili Deaktiviraj alarm) upravljati alarmnim sustavom u ručnom režimu rada.

Slika 55. – Dizajn ručnog režima rada

Izvor: Autor

Odabirom ručnog režima rada pojavljuju nam se četiri sličice koje su definirane tako da pritiskom na jednog od njih, aplikacija otvara novi prozor ovisno o odabiru što je prikazano na slici 55.. Sličice nam prikazuju Upravljanje rasvjetom (gore lijevo) , Upravljanje klimatizacijom (gore desno), Upravljanje garažnim vratima (dole lijevo), Upravljanje alarmnim sustavom (dole desno).

Slika 56. – Dizajnerski dio upravljanja rasvjetom

Izvor: Autor

Odabirom na Upravljanje rasvjetom otvara se novi prozor prikazan na slici 56. koji omogućuje pritiskom na gumb „Uključi rasvjetu“ uključivanje rasvjete na modelu pametne kuće, te gumb „Isključi rasvjetu“ isključivanje rasvjete. Pritiskom na gumb „Natrag na ručni način rada“ vraćamo se na početni zaslon gdje se nalaze opcije biranja između automatskog i ručnog režima rada.

Slika 57. – Dizajnerski dio Upravljanja klimatizacijom

Izvor: Autor

Odabirom na Upravljanje klimatizacijom otvara nam se novi prozor u kojem imamo tri opcije za odabir i tekstualni okvir u kojem se naznačuje da odaberemo jednu od opcija, a to su „Natrag na ručni način rada“, „Upravljanje hlađenjem“ i „Upravljanje grijanjem“. Pritiskom na gumb „Natrag na ručni način rada“ vraćamo se na početni zaslon gdje se nalaze opcije biranja između automatskog i ručnog režima rada. Odabirom na upravljanje hlađenjem otvaraju nam se još dvije opcije i jedan tekstualni okvir koji naznačuje da upravljamo hlađenjem što je prikazano na slici 57 na lijevoj strani. Dvije opcije koje možemo odabrati, jedna je „Upali hlađenje“ i druga „Ugasi hlađenje“. Također i odabirom na „Upravljanje grijanjem“ imamo dvije opcije prikazane na slici 57. na desnoj strani, „Upali grijanje“ i „Ugasi grijanje“.

Slika 58. – Dizajnerski dio za Upravljanje garažnim vratima

Izvor: Autor

Odabirom na „Upravljanje garažnim vratima“ otvara se novi prozor koji možemo vidjeti na slici 58.. U novom prozoru pojavljuju se opcije „Natrag na ručni način rada“, „Promjeni lozinku“, „Resetiraj lozinku“, tekstualni okvir u kojeg moramo upisati lozinku, te gumb kojim potvrđujemo lozinku. Pritiskom na gumb „Natrag na ručni način rada“ vraćamo se na početni zaslon gdje se nalaze opcije biranja između automatskog i ručnog režima rada. Upisivanjem lozinke u tekstualni okvir i pritiskom na gumb „Potvrdi lozinku“ otvaraju se dvije nove mogućnosti jedna je „Otvori garažna vrata“, a druga „Zatvori garažna vrata“. Pritiskom na

„Promjeni lozinku“ prikazuje nam se još jedan tekstualni okvir u kojega upisujemo novu lozinku, a u drugi tekstualni okvir upisujemo staru lozinku. Ponovnim pritiskom na gumb „Promjeni lozinku“ sprema se nova lozinka. Gumbom „Resetiraj lozinku“ aplikacija nam postavlja lozinku u pred definiranu lozinku koja glasi „admin“.

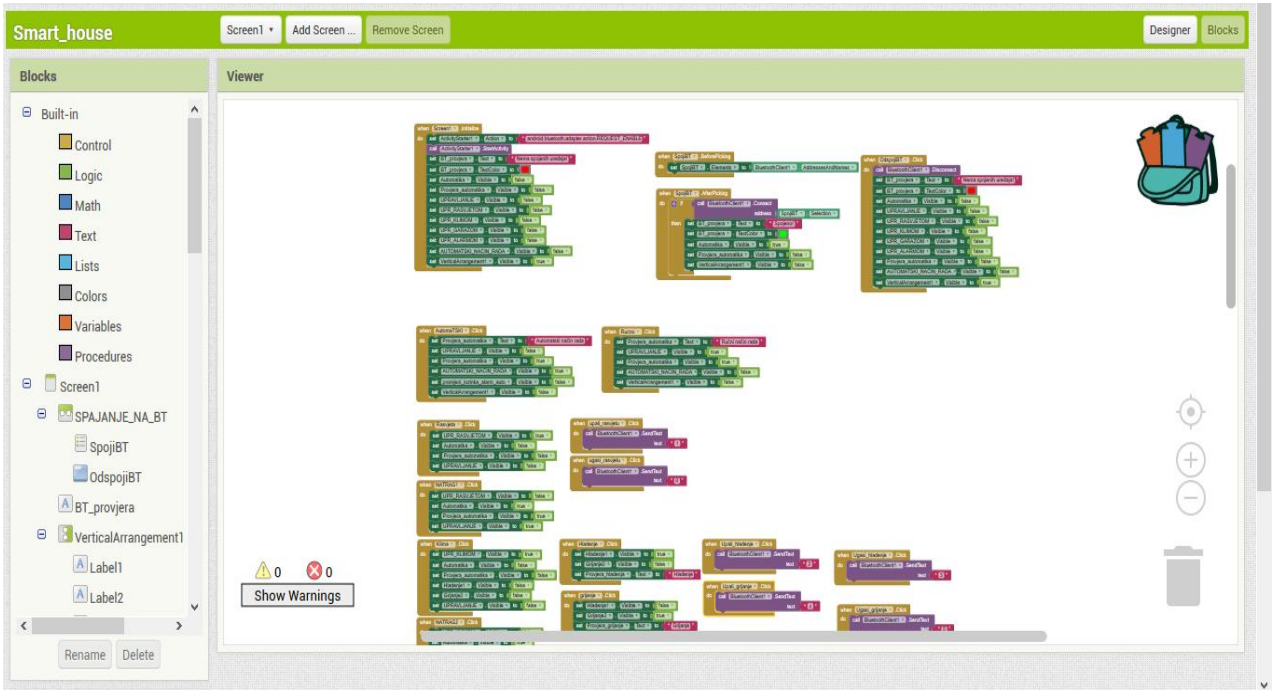
Slika 59. – Dizajnerski dio za Upravljanje alarmnim sustavom

Izvor: Autor

Odabirom „Upravljanja garažnim vratima“ otvara se prozor u kojem se nalazi pet opcija i dva tekstualna okvira od kojih je jedan naznaka u kojem dijelu upravljanja se nalazimo, a drugi za upisivanje lozinke za upravljanje alarmnim sustavom što je prikazano na slici 59.. Pritiskom na gumb „Natrag na ručni način rada“ vraćamo se na početni zaslon gdje se nalaze opcije biranja između automatskog i ručnog režima rada. Upisivanjem lozinke u tekstualni okvir i pritiskom na gumb „Upali alarm“ aktivira se alarmni sustav na modelu pametne kuće, a nakon upisane točne lozinke i pritiskom na gumb „Ugasi alarm“ deaktiviramo alarmni sustav. Pritiskom na „Promjeni lozinku“ prikazuje nam se još jedan tekstualni okvir u kojega upisujemo novu lozinku, a u drugi tekstualni okvir upisujemo staru lozinku. Ponovnim pritiskom na gumb „Promjeni lozinku“ sprema se nova lozinka. Gumbom „Resetiraj lozinku“ aplikacija nam postavlja lozinku u pred definiranu lozinku koja glasi „admin“.

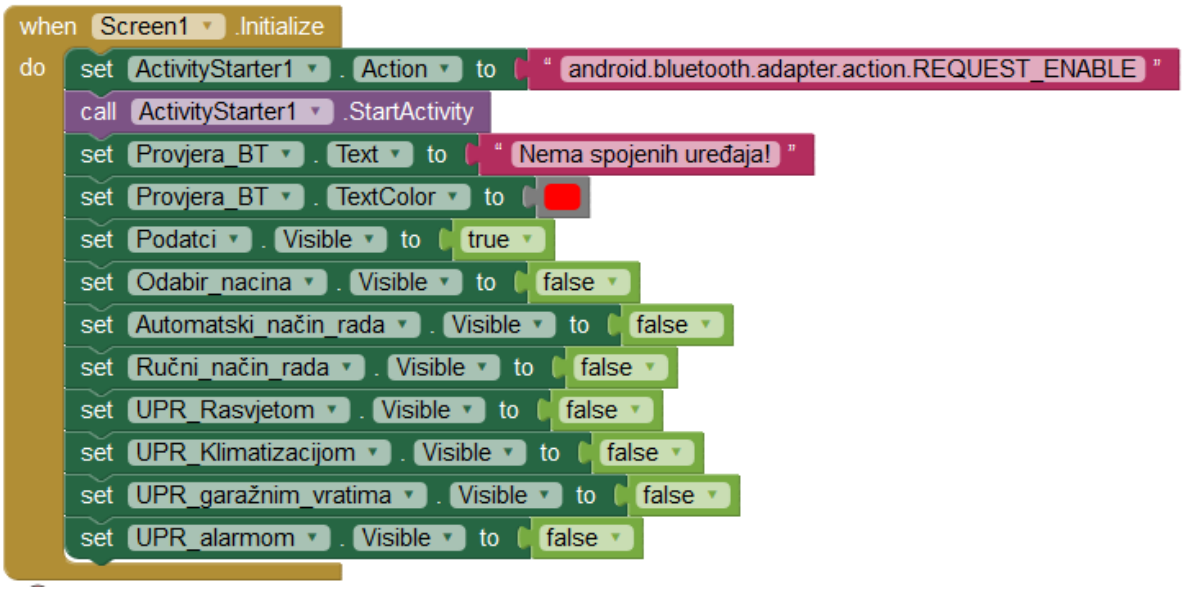
3.4.2. Izrada logičkog dijela aplikacije

Izrada logičkog dijela aplikacije sastoji se u povezivanju raznih blokova koji se spajaju u jednu cjelinu i izvršavaju određeni dio aplikacijskog bloka. Za izradu logičkog dijela aplikacije koristile su se logičke funkcije *when do, if then, set* (za postavljanje tekstualnih okvira), *call* (za pozivanje nevidljivih komponenata, *initialize global* (za definiranje liste), *set to* (za postavljanje očitanja u njihove tekstualne oznake), *split text to* (za razdvajanje očitanja), *select list item list indeks* (za sortiranje očitanja). Slika 60. prikazuje izgled logičkog dijela u MIT App Inventoru. Cjelokupni logički dio izrade aplikacije moguće je pogledati u popisu priloga pod nazivom Prilog 2. – Logički dio aplikacije za upravljanje modelom pametne kuće.



Slika 60. – Logički dio u MIT App Inventoru

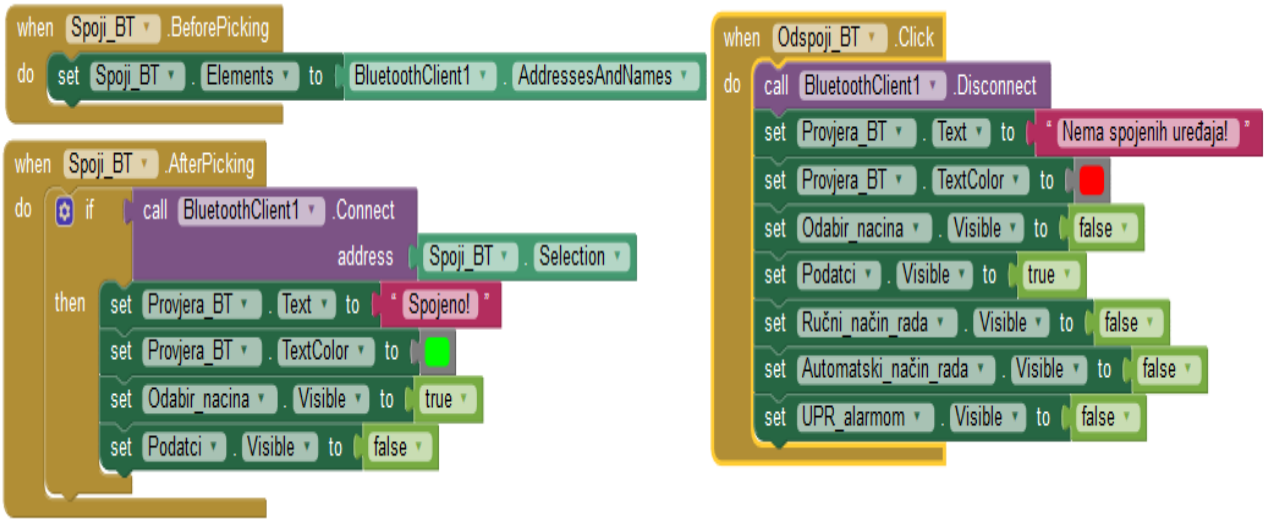
Izvor: Autor



Slika 61. – Logički dio početne stranice

Izvor: Autor

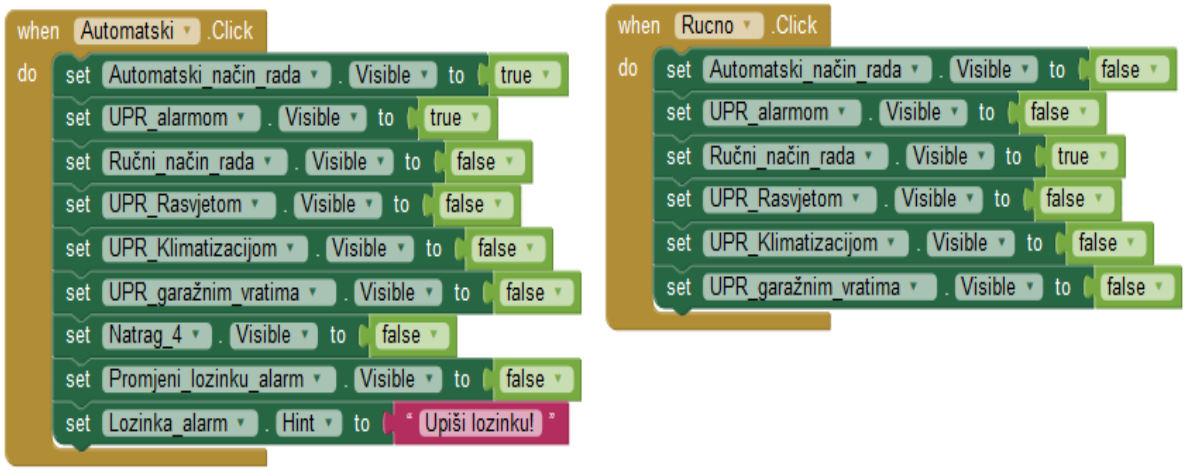
Na samom početku aplikacije kao što prikazuje slika 61., kada se inicijalizira početni zaslon aplikacije, poziva se *ActivityStarter* koji nam služi za brzo spajanje na Bluetooth uređaj. Tekstualni okvir se postavlja u „Nema spojenih uređaja!“, te se tekst unutar njega postavlja u crvenu boju, a svi ostali tabovi i opcije se postavljaju u nevidljivo.



Slika 62. – Logički dio spajanja na Bluetooth uređaj

Izvor: Autor

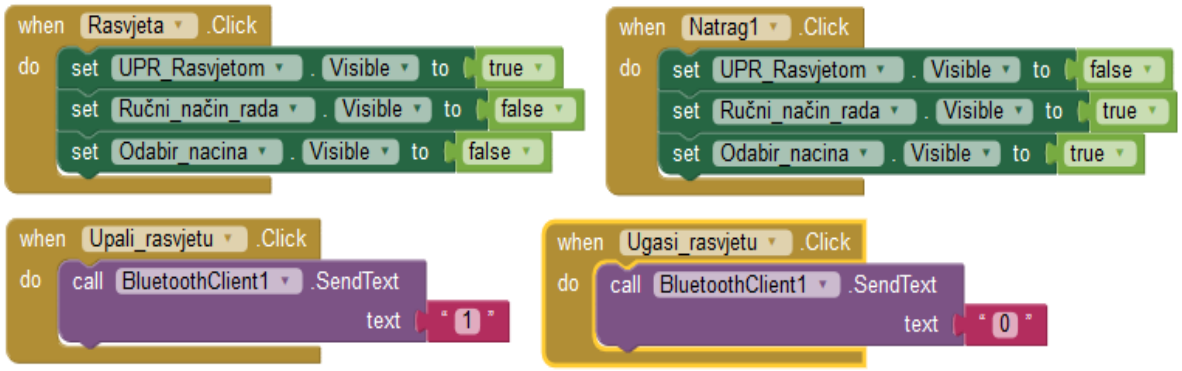
Nakon početnog zaslona na red dolazi logički dio zadužen za spajanje i od spajanje sa Bluetooth uređaja što je i prikazano na slici 62.. Prije nego što odaberemo opciju „Spoji BT“, *Bluetooth client 1* postavlja adrese i imena Bluetooth uređaja u listu „Spoji BT“. Nakon što odaberemo opciju „Spoji BT“ ako je dostupna Bluetooth konekcija, da adrese dostupnih uređaja postavi u „Spoji BT“ selekciju. Nakon spajanja na Bluetooth uređaj, tekstualni okvir se postavlja u „Spojeno!“, te tekst poprima zelenu boju, a ostali tabovi se postavljaju u vidljive. Odabirom opcije „Odspoji BT“ poziva se *Bluetooth client 1* da prekine Bluetooth konekciju i svi tabovi i tekstualni okviri postaju nevidljivi, a u tekstualni okvir se upisuje „Nema spojenih uređaja!“ i tekst poprima crvenu boju.



Slika 63. – Logički dio nakon uspješnog spajanja na Bluetooth uređaj

Izvor: Autor

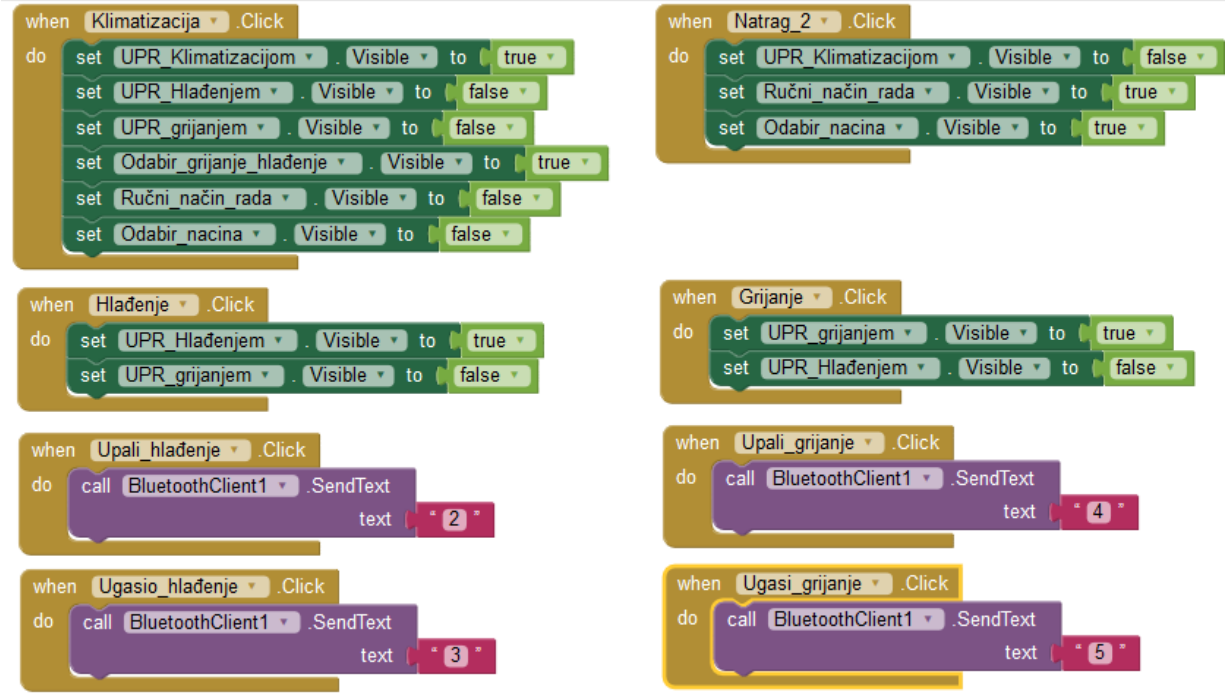
Nakon uspješnog spajanja na Bluetooth uređaj opcija biranja između automatskog i ručnog režima rada nam postaju vidljive, te na slici 63.. Se vidi kako su definirani načini rada. Ako odaberemo automatski režim rada u tekstualni okvir nam se upisuje „Automatski način rada“, te nam se ručni režim rada postavlja u nevidljivo, a automatski režim rada u vidljivo i ostali tabovi koji su zaduženi za automatski režim rada, te naravno i tekstualni okviri u koje se upisuju očitanja očitana na senzorima. Odabirom ručnog režima rada tekstualni okvir se postavlja u „Ručni način rada“, te tab sa ručnim upravljanjem nam postaje vidljiv a tab sa automatskim režimom rada nevidljiv.



Slika 64. – Logički dio za Upravljanje rasvjetom

Izvor: Autor

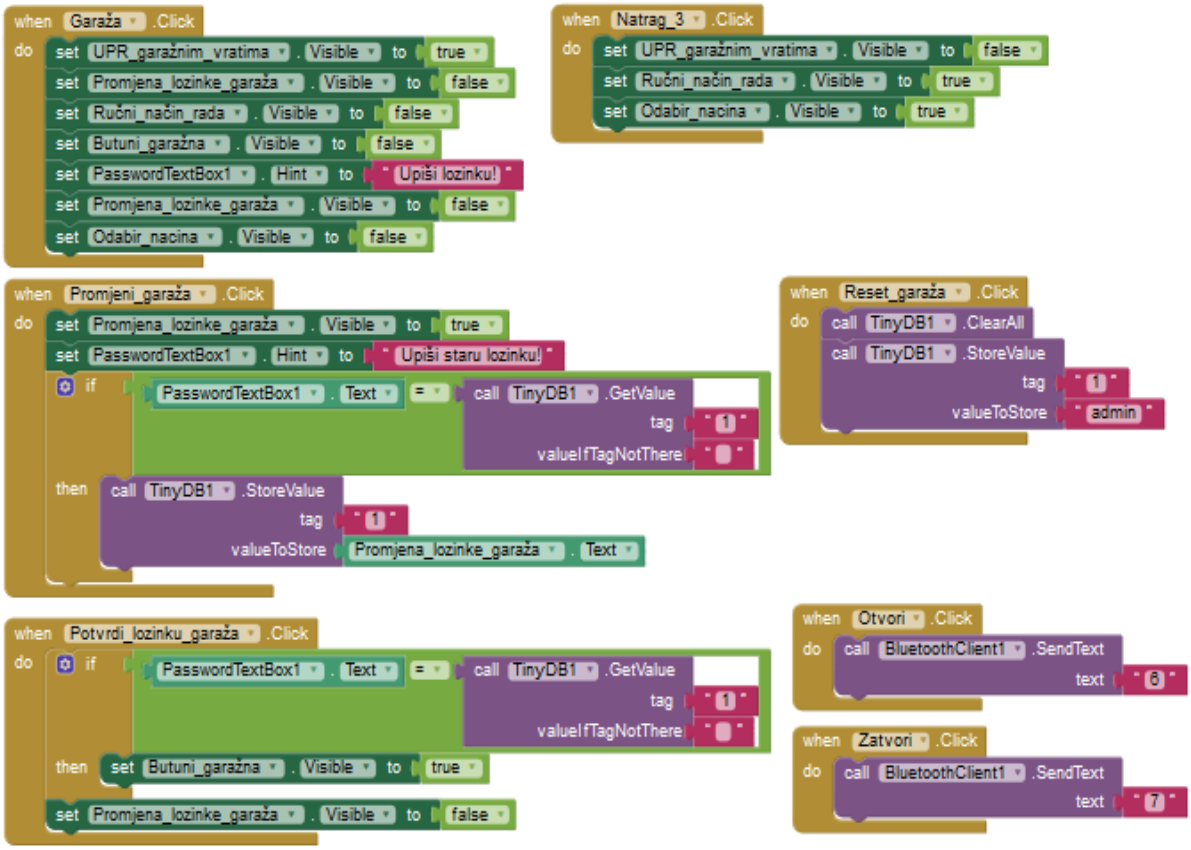
U ručnom načinu rada imamo četiri mogućnosti upravljanja, a jedna od njih je upravljanje rasvjetom. Na slici 64. prikazan je logički dio Upravljanja rasvjetom. Klikom u ručnom režimu rada na Upravljanje rasvjetom otvara nam novi prozor u kojem dosada nevidljivi blokovi postaju vidljivi. Definiran je gumb koji vraća na početni zaslon odnosno na ručni režim rada, a on je definiran tako što se vidljivi djelovi postavljaju u nevidljive. Za uključivanje rasvjete zadužen je gumb „Uključi rasvjetu“ koji je definiran tako da kada se klikne na njega *Bluetooth client1* šalje u tekstualnom obliku „1“, a za isključivanje rasvjete gumb „Isključi rasvjetu“ koji kod njegovog pritiska *Bluetooth client1* šalje u tekstualnom obliku „0“.



Slika 65. – Logički dio upravljanja klimatizacijom

Izvor: Autor

Isto kao i kod upravljanja rasvjetom, upravljanje klimatizacijom ima definirane gumbe za natrag na ručni način rada koji nas vraća pomoću stavljanja vidljivih dijelova u nevidljive. Razlika je ta u tome što u upravljanju klimatizacijom imamo dva moguća odabira a to je „Upravljanje hlađenjem“ i „Upravljanje grijanjem“, a to je definirano tako da odabirom na jednu od tih dviju ponuđenih opcija aplikacija postavlja vidljive dijelove u nevidljive ovisno o tome koju opciju želimo izabrati. Tako da ako izaberemo opciju „Upravljanje hlađenjem“ aplikacija postavlja gumbe „Uključi hlađenje“ i „Isključi hlađenje“ iz nevidljivih u vidljive. Odabirom na gumb „Uključi hlađenje“ *Bluetooth client1* šalje broj „2“, a odabirom na „Isključi hlađenje“ šalje broj „3“. Slika 65. prikazuje logički dio aplikacije za upravljanje klimatizacijom.

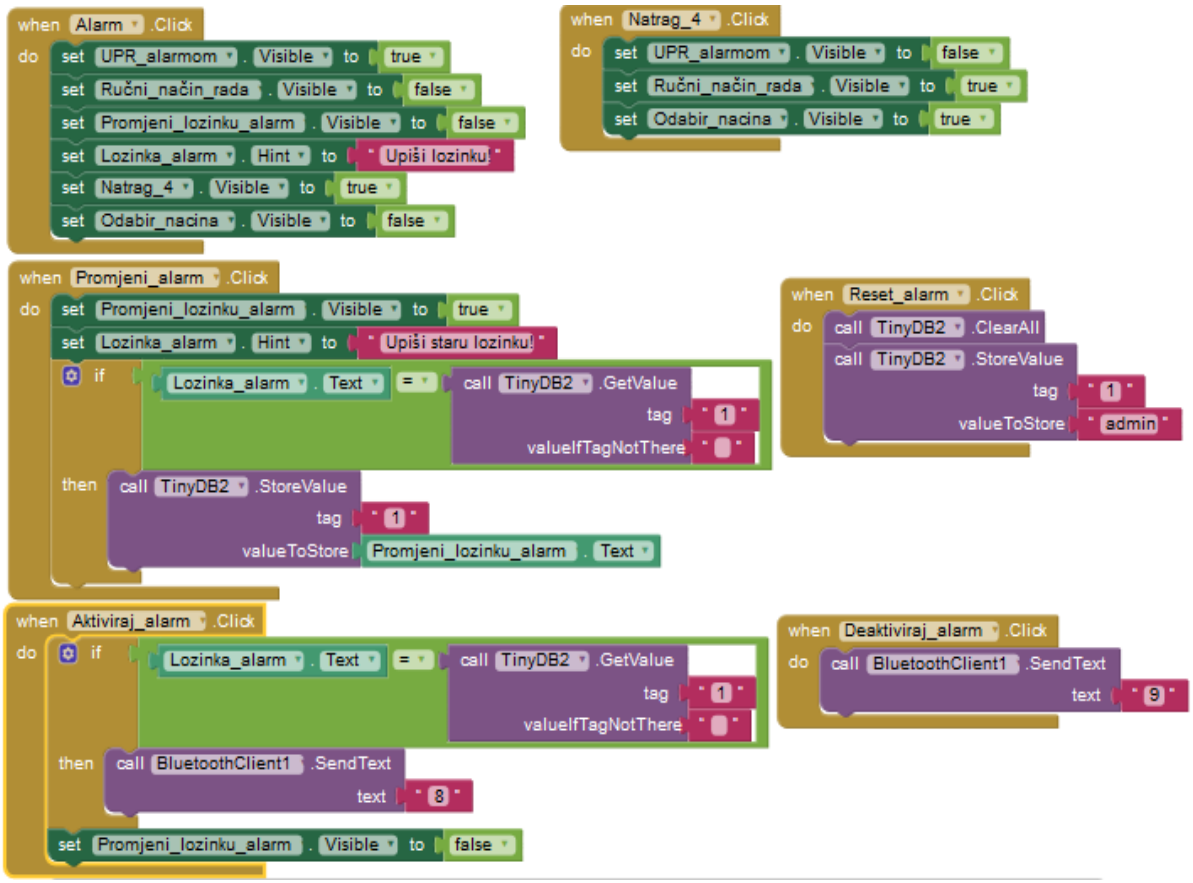


Slika 66. – Logički dio Upravljanja garažnim vratima

Izvor: Autor

Odabirom na „Upravljanje garažnim vratima“ dosada nevidljive opcije: tekstualnih okvira za lozinku, gumbova za potvrđivanje lozinke, resetiranja i promjeni lozinku postaju vidljive, te *hint* tekstualnog okvira za utipkavanje lozinke se postavlja u „Upiši lozinku“ što vidimo na slici 66.. Gumb za promjeni lozinku definiran je tako da pritiskom na njega tekstualni okvir za

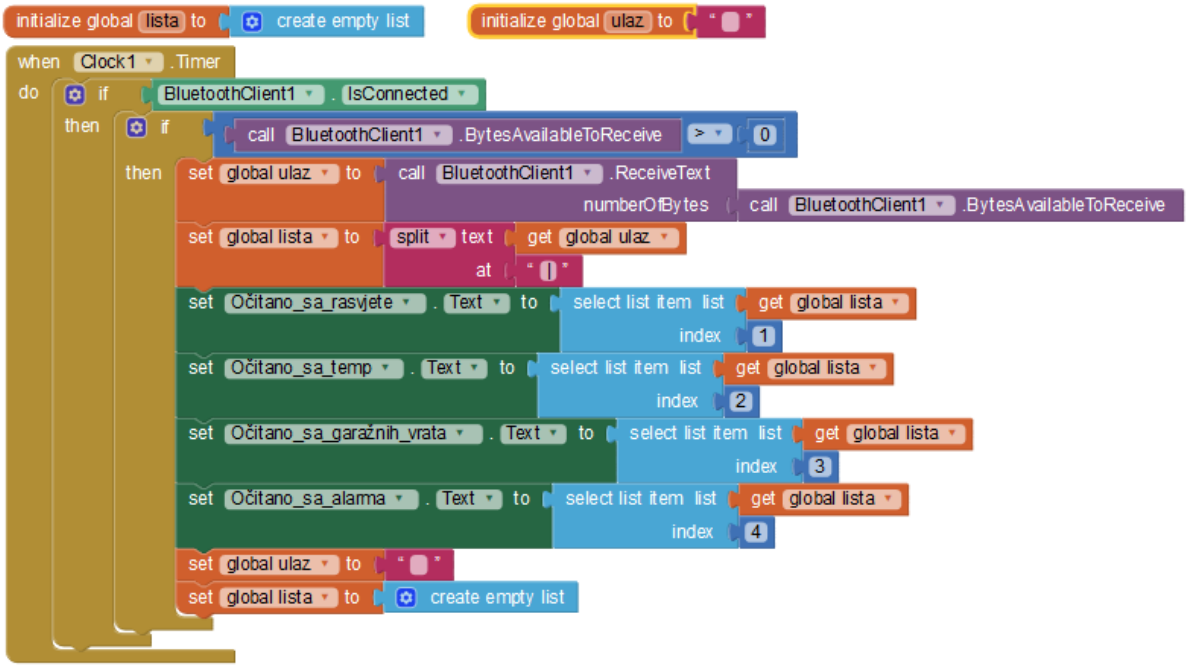
promjenu lozinke postane vidljiv, te se u njega upisuje željena nova lozinka, a u tekstualni okvir za lozinku poprima *hint* „Upiši staru lozinku“. Nakon upisivanja lozinka aplikacija provjerava dali se upisana stara lozinka stvarno podudara sa starom lozinkom, ako se podudara poziva se naredba koja novo upisanu lozinku sprema u *TinyDB* (memorija). Pritiskom na gumb reset koji je definiran tako da ako se pritisne zove naredbu koja briše sve iz *TinyDB*-a i unutra postavlja pred definiranu lozinku „admin“. Upisom lozinke bez promjene lozinke u odgovarajući tekstualni oblik i pritiskom na gumb „Potvrdi lozinku“ poziva se naredba koja uspoređuje upisanu lozinku sa stvarnom lozinkom, te ako se podudaraju, aplikacija postavi dva gumba „Otvori garažna vrata“ i „Zatvori garažna vrata“ u vidljive. Pritiskom na gumb „Otvori garažna vrata“ *Bluetooth client1* šalje broj „6“, a pritiskom na gumb „Zatvori garažna vrata“ šalje broj „7“.



Slika 67. – Logički dio Upravljanja alarmnim sustavom

Izvor: Autor

Na slici 67. je prikazan logički dio za upravljanje alarmnim sustavom, koji je gotovo isti kao i prijašnje opisani logički dio za upravljanje garažnim vratima. Gumbovi „Promjeni lozinku“ i „Resetiraj lozinku“ su isto definirani samo što se ne spremaju u isti *TinyDB* (memorija) i u ovom slučaju nema potvrđivanja lozinke nego samim jednostavnim utipkavanjem točne lozinke u tekstualni okvir za to zadužen i pritiskom na jedan od dva gumba „Aktiviraj alarm“ i „Deaktiviraj alarm“ *Bluetooth client1* šalje brojeve „8“ za aktiviranje alarma, odnosno broj „9“ za deaktiviranje alarma.



Slika 68. – Logički dio očitanja senzora u aplikaciji

Izvor: Autor

Slika 68. prikazuje definiran logički dio za spremanje i prikaz očitanja sa senzora. Na početku definirana je prazna lista pod nazivom „lista“ i jedan prazan tekstualni okvir nazvan „ulaz“. Za provjeru kada će se primati očitanja zadužen je *Clock* (sat). Kada *Clock* odbroji provjerava se dali ima povezanosti sa Bluetooth uređajima, ako ima povezanosti onda se poziva *Bluetooth client1* koji provjerava dali ima dolaznih podataka, ako ima dolaznih podataka, tada on prima tekstualni oblik i sprema ga u prazni tekstualni oblik. Nakon toga u prije definiranu praznu listu se spremaju tekstualni podatci koji se podjeljuju sa znakom „|“ i tako odvajaju na način da aplikacija upisuje podatke na pripadajuće mjesto. Kada sve to obavi kreće se na postavljanje teksta u pripadajuće tekstualne okvire sa blokom *select list item list indeks*. Nakon postavljanja na pripadajuće pozicije aplikacija opet postavlja praznu listu i prazan tekstualni okvir

3.5. Opis rada modela pametne kuće

Model pametne kuće koja se izradila ima dva načina rada jedan je automatski način rada, a drugi je ručni način rada. Promjena između ta dva načina rada se izvršava pritiskom na sklopku. Ako je sklopka u uključena sustav ulazi u automatski način rada, a ako je sklopka isključena sustav radi u ručnom načinu rada. Za upravljanje automatskim sustavom rada koriste se senzori koji uključuju i isključuju podsustave modela pametne kuće, a za ručni način rada koristi se Android uređaj pomoću Bluetooth tehnologije koji služi za upravljanje. Alarmni sustav je jedan od podsustava modela pametne kuće koji radi u oba dva načina rada identično. Sustav alarma se sastoji od ultrazvučnog senzor koji mjeri udaljenost od nekog objekta, te služi nam za otkrivanje detekcije prisutnosti u prostoru u kojem je smješten. Detekcijom na ultrazvučnom senzoru, uključuje se crvena led dioda koja zablinka dva puta i ostaje upaljena dok se alarmni sustav ne isključi, te zujalica koja nam predstavlja zvučni efekt kad se detektira nepoželjna kretnja.

3.5.1. Automatski način rada

Kao što je u uvodnom dijelu opisa modela pametne kuće napisano automatski režim rada se izvršava kada je sklopka uključena. Automatski način rada se sastoji od tri podsustava a to su: sustav za upravljanje rasvjetom, sustav za upravljanje klimatizacijom i sustav za upravljanje garažnim vratima. Sustav za upravljanje rasvjetom sastoji se od PIR senzora, dvije žute led diode te foto otpornik. On radi tako da se prvom detekcijom pokreta na PIR senzoru, uključuju se dvije led diode koje se postavljaju ovisno o stanju na foto otporniku, tj. ako je razina svijetlosti u prostoriji niska led diode svijetle jače, odnosno ako je razina svijetlosti u prostoriji visoka led diode svijetle slabije. Drugom detekcijom na PIR senzoru rasvjeta se isključuje, odnosno izlaskom iz prostorije gasi se rasvjeta. Za upravljanje sustavom klimatizacije u automatskom načinu rada koristi se DHT 11 senzor temperature, koji ovisno o stanju temperature pali dvije led diode, od kojih nam ona plave boje predstavlja sustav za hlađenje prostorije, ili drugu led diodu crvene boje koja nam predstavlja sustav za grijanje prostorije. Treći podsustav je sustav upravljanja garažnim vratima. On se sastoji od PIR senzora, koji prvom detekcijom pokreta otvara garažna vrata na modelu pametne kuće, a drugom detekcijom zatvara garažna vrata.

3.5.2. Ručni način rada

Kao što je prije spomenuto ručni način rada se izvršava kada je sklopka isključena. Za ručno upravljanje modelom pametne kuće koristi se Android uređaj sa aplikacijom koja naredbe prima pomoću Bluetooth tehnologije. U ručnom režimu rada moguće je upravljati sa sljedećim podsustavima: sustav rasvjete, sustav klimatizacije i sustav garažnih vrata. U ručnom načinu rada miruju svi senzori osim ultrazvučnog senzora koji je aktivan cijelo vrijeme zbog potrebe za alarmnim sustavom u ručnom i automatskom načinu rada. Rasvjetom se upravlja pomoću aplikacije instalirane na Android uređaju. Rasvjetu možemo uključiti i isključiti jednostavnim pritiskom na gumb, te pritiskom na gumb uključi uključujemo dvije led diode koje predstavljaju rasvjetna tijela, a pritiskom na gumb isključi isključujemo led diode. Za upravljanje klimatizacijom također se koristi aplikacija na Android uređaju, kojom izborom između dvije opcije, grijanje i hlađenje, možemo odabrati nama željeni izbor kojim u tom trenutku želimo izvršiti. Isto kao i u sustavu upravljanja rasvjetom i u sustavu upravljanja klimatizacijom ovisno o odabiru, ako želimo hlađenje možemo uključiti ili isključiti hlađenje pritiskom na gumb, te tako uključujemo i isključujemo plavu led diodu koja nam predstavlja sustav hlađenja, a ako želimo grijanje isto tako može se uključiti i isključiti pritiskom na gumb, i uključuje se ili isključuje crvena led dioda koja predstavlja sustav grijanja. Treći sustav kojim se ručno može upravljati je sustav garažnih vrata. Njime se upravlja tako da u aplikaciji na Android uređaju moramo upisati postavljenu lozinku za otvaranje ili zatvaranje garažnih vrata zbog sigurnosnih razloga. Pritiskom na gumb otvori sustav pokreće servo motor koji otvara garažna vrata, odnosno gumb zatvori pokreće servo motor koji zatvara garažna vrata.

3.6. Financijsko izvješće izrade modela pametne kuće

Za izradu ovog rada potrebno je kupiti potrebne komponente za funkcionalnost modela pametne kuće. Kao što je gore navedeno potrebne komponente će biti prikazane u tablici 6. sa svojom cijenom i rokom isporuke iz različitih zemalja kao usporedbu za troškove.

Tablica 6. – Korištene elektroničke komponente za izradu modela pametne kuće

Naziv komponente	Hrvatska			Kina		
	Količina	Cijena[44]	Rok isporuke	Količina	Cijena[45]	Rok isporuke
Arduino UNO	1 kom.	99,00	2 - 3 dana	1 kom.	17,19	20 – 30 dana
Bluetooth HC06	1 kom.	99,00	2 – 3 dana	1 kom.	16,13	20 – 30 dana
DHT 11	1 kom.	25,00	2 – 3 dana	1 kom.	6,19	20 – 30 dana
PIR senzor	2 kom.	25,00	2 – 3 dana	2 kom.	8,88	20 – 30 dana
Ultrazvučni senzor	1 kom.	25,00	2 – 3 dana	1 kom.	4,75	20 – 30 dana
Servo motor	1 kom.	35,00	2 – 3 dana	1 kom.	19,25	20 – 30 dana
Zujalica	1 kom.	5,00	2 – 3 dana	1 kom	4,00	20 – 30 dana
Foto otpornik	1 kom.	4,00	2 – 3 dana	20 kom.	3,49	20 – 30 dana
Led diode	13 kom.	9,00	2 – 3 dana	10 kom.	2,44	20 – 30 dana
Spojne žice	145 kom.	75,00	2 – 3 dana	120 kom.	8,57	20 – 30 dana
Eksperimentalna	1 kom.	19,00	2 – 3 dana	1 kom.	6,69	20 – 30 dana
Otpornik 220 Ω	25 kom.	2,00	2 – 3 dana	100 kom.	3,13	20 – 30 dana
Otpornik 10 kΩ	15 kom.	2,00	2 – 3 dana	100 kom.	3,32	20 – 30 dana
Sklopka	1 kom.	5,00	2 – 3 dana	1 kom	4,25	20 – 30 dana
Ukupno		774,00			108,28	

Izvor: Autor

Usporedbom komponenata koje se mogu kupiti u Hrvatskoj i u Kini velika je razlika u samoj cijeni i količini koja se dobije prilikom kupnje. Tako za kupovinu komponenata u Hrvatskoj potrebno je izdvojiti 774,00 kn što je navedeno u tablici 6., a za kupovinu van granica Republike Hrvatske, odnosno u Kini što iznosi znatno manje nego u Hrvatskoj a iznos je 108,28 kn. Što se tiče roka isporuke za ove komponente znatno je manji za kupovinu u Republici Hrvatskoj nego za kupovinu u Kini. Rok isporuke za Hrvatsku iznosi od 2 – 3 radna dana, dok za komponente naručene u Kini potrebno je čekati između 20 i 30 radnih dana ovisno o prodavaču.

[44] Cijena za Hrvatsku preuzeta sa : https://e-radionica.com/hr/ (15.09.2017.)
[45] Cijena za Kinu preuzeta sa : https://www.aliexpress.com/ (15.09.2017.)

Tablica 7. – Potreban materijal za izradu modela pametne kuće

Naziv materijala	Količina	Cijena[46]	Rok isporuke
Šperploča 4 mm	1 m^2	77,00	Odmah
Lijepljena ploča 400 x 800 x 18 mm	1 kom.	39,00	Odmah
Ljepilo za pištolj na vruću plastiku	10 kom.	20,00	Odmah
Žuta boja	1 kom.	15,00	Odmah
Bijela boja	1 kom.	15,00	Odmah
Ukupno		166,00	

Izvor: Autor

Za izradu konstrukcijskog dijela modela pametne kuće potrebni materijal je naveden u tablici 7.. Kao što se vidi iz priložene tablice potrebno je bilo izvršiti kupovinu šperploče debljine 4 mm koja se koristi za izradu stranica modela, lijepljene ploče dimenzija 400 x 800 x 18 mm za podlogu na kojoj će se nalaziti, te ljepilo za vrući pištolj za spajanje dijelova, te dvije različite boje za bojanje. Ukupna novčana cifra koja se potrošila je 166,00 kn, te rok isporuke je odmah zbog dostupnosti materijala u svakoj poslovnici sa građevinskim materijalom.

Tablica 8. – Troškovi izrade modela pametne kuće

Naziv	Trošak
Elektroničke komponente	108,28
Materijal potreban za izradu modela	166,00
Ukupno	274,28

Izvor: Autor

Za svrhu izrade modela pametne kuće, kupovina elektroničkih komponenata je obavljena pomoću online kupnje iz Kine, zbog znatno manjeg troška od kupovine u Republici Hrvatskoj. Sveukupni iznos troškova za model pametne kuće iznosi 274,28 kn.

[46] Cijena preuzeta u trgovini Bauhaus

4. ZAKLJUČAK

Pod pojmom pametna kuća podrazumijeva se potpuno automatiziran sustav upravljanja stambenim objektom. U današnje vrijeme se sve više susrećemo sa pojmom pametna kuća, ali u budućnosti će to biti uobičajeni termin zbog sve veće populariziranosti takvog sustava. Pametna kuća obuhvaća integraciju više podsustava u jedan cjeloviti sustav. Sustavi koji mogu sačinjavati pametnu kuću su: osvjetljenje, grijanje i ventilacijski sustav, sigurnosni sustav, multimedijski sustav, itd.. Integracijom navedenih sustava dobivamo jedinstveni sustav automatizacije stambenog objekta koji doprinosi korisniku potpuno zadovoljstvo, što se tiče uštede i učinkovitosti, a i povećava i sam komfor i udobnost korisnikova objekta. Implementacijom ovakvog sustava na već postojeću objekt ili izgradnjom novog stambenog objekta koji se sačinjava sa sustavom automatizacije možemo si olakšati život. Zbog toga što imamo sve podatke o potrošnji, gdje se što dešava i kako koristimo sustav na jednom mjestu.

Na temelju automatizacije kućanstva napravljen je i model koji prikazuje najosnovnije sustave: osvjetljenje, klimatizacija, sigurnost i upravljanje garažnim vratima. Izradom modela vidljivo je kako takav automatizirani sustav doprinosi korisnikovo zadovoljstvo svojom udobnošću i komforom zbog jednostavnosti upravljanja sa samo jednog mjesta, u ovom slučaju preko mobilnog uređaja.

LITERATURA

Stručni radovi:

1. Laboratorij za sustave i signale: **Sigurnost pametnih kuće**, Fakultet elektrotehnike i računarstva, Sveučilište u Zagrebu, Travanj 2012.
2. Jakovčić, A.; Marić, M. :**Pametna kuća**, Seminarski rad, Svibanj 2007., Zagreb
3. Matković, P. :**Norme i protokoli za automatizaciju kućanstva**, Završni rad, Lipanj 2013., Zagreb

Materijali sa predavanja:

4. Matas, K. : **Mrežni protokoli i norme**, Računalne mreže, Materijali sa predavanja

Internet izvori:

5. https://bs.wikipedia.org/wiki/Sistem_ku%C4%87ne_automatike (16.08.2017.)
6. http://www.eko.zagreb.hr/default.aspx?id=109 (16.08.2017.)
7. http://www.pametna-kuca.ba/index.php/pametna-kuca/mogucnosti-sistema (16.08.2017.)
8. http://www.knx.org/hr/knx/stambene-zgrade/ (16.08.2017.)
9. https://e-radionica.com/hr/blog/2015/08/19/kkm-ultrazvucni-modul-hc-sr04/ (16.08.2017.)
10. https://e-radionica.com/hr/blog/2015/08/19/kkm-hc-sr501/ (16.08.2017.)
11. https://e-radionica.com/hr/blog/2016/06/23/kkm-servo-motor-tower-pro-sg90/ (16.08.2017.)
12. https://e-radionica.com/hr/blog/2015/08/19/kkm-dht11/ (16.08.2017.)
13. http://e-radionica.com/wp/wp-content/uploads/2015/05/DSC_8529-1024x678.jpg (27.08.2017.)
14. http://pimami.hr/wp-content/uploads/2014/05/image7_hclite_eng.png (29.08.2017.)
15. http://www.hyperion.rs/images/fancybox/SmartHome.jpg (29.08.2017.)
16. http://www.hyperion.rs/images/fancybox/HyperiON_Concept.jpg (29.08.2017.)
17. http://www.ee.ic.ac.uk/leon.zhang11/yr2proj/img/sinewave.png (29.08.2017.)
18. https://www.bwired.nl/images/how/schema_Xanura.gif (29.08.2017.)
19. http://www.protoconvert.com/portals/0/DiagramsForIndividualPage/Lonworks%20to%20BACnet%20IP%20Scenario%201.jpg (29.08.2017.)

20. http://www.hometoys.com/htinews/aug97/articles/kwacks//img02.gif (29.08.2017.)
21. http://trade.amx.com/imgs/diagrams/NXB-KNX.Diagram.png (29.08.2017.)
22. http://plane-tehnika.hr/wp-content/uploads/2015/07/pametne-kuce.jpg (29.08.2017.)
23. https://hr.wikipedia.org/wiki/Arduino (29.08.2017)
24. https://storecdn.arduino.cc/uni/catalog/product/cache/1/image/1800x/ea1ef423b933d797cfca49bc5855eef6/A/0/A000066_front_2.jpg (29.08.2017.)
25. http://www.instructables.com/id/Add-bluetooth-to-your-Arduino-project-ArduinoHC-06/ (29.08.2017.)
26. https://store.arduino.cc/arduino-uno-rev3 (29.08.2017.)
27. https://e-radionica.com/hr/blog/2016/04/06/bluetooth/ (29.08.2017.)
28. https://hr.wikipedia.org/wiki/Android_(operacijski_sustav)#/media/File:Nexus_5_L.png (29.08.2017.)
29. https://hr.wikipedia.org/wiki/Android_(operacijski_sustav) (29.08.2017.)
30. http://e-radionica.com/wp/wp-content/uploads/2015/03/DSC_8467.jpg (29.08.2017.)
31. http://e-radionica.com/wp/wp-content/uploads/2015/03/Untitled-Sketch_bb.png (29.08.2017.)
32. http://e-radionica.com/wp/wp-content/uploads/2015/05/Screen-Shot-2015-05-18-at-16.24.48.png (29.08.2017.)
33. http://e-radionica.com/wp/wp-content/uploads/2015/05/Progetto-PIR-con-accensione-LED_bb.png (29.08.2017.)
34. https://e-radionica.com/wp/hrvatski/wp-content/uploads/sites/3/2016/06/dsc_4840-1024x678.jpg (29.08.2017.)
35. https://e-radionica.com/wp/hrvatski/wp-content/uploads/sites/3/2016/06/Untitled-Sketch_bb.png (29.08.2017.)
36. http://e-radionica.com/wp/wp-content/uploads/2015/03/dht11.jpg (29.08.2017.)
37. http://e-radionica.com/wp/wp-content/uploads/2015/03/Untitled-Sketch_bb2.png (29.08.2017.)
38. http://home.planet.nl/~lhendrix/x10_history.htm (29.08.2017.)
39. https://en.wikipedia.org/wiki/LonWorks (29.08.2017.)
40. https://en.wikipedia.org/wiki/KNX_(standard) (29.08.2017.)
41. https://en.wikipedia.org/wiki/CEBus (29.08.2017.)
42. http://buildyoursmarthome.co/home-automation/protocols/knx/ (29.08.2017.)

43. http://pimami.hr/vimarautomatika/?gclid=Cj0KEQjwwqXMBRCD1afDldbp5qEBEiQAP4BDaJipAxrcGP1m2nwZVpvh1wzDQCzzcwgIxTVedtNpvkgaAsh-8P8HAQ4 (29.08.2017.)
44. http://e-radionica.com/wp/wp-content/uploads/2015/05/x.png (31.08.2017.)
45. http://www.enciklopedija.hr/natuknica.aspx?ID=59111 (31.08.2017.)
46. http://3.bp.blogspot.com/-h4OVYkj6lZ4/Uelgt9ye01I/AAAAAAAAAU0/bzU-wyBM38Y/s1600/konstrukcija-otpornika.jpg (31.08.2017.)
47. http://osnove.tel.fer.hr/VJEZBEOE/DC_2.htm (31.08.2017.)
48. http://e-radionica.com/wp/wp-content/uploads/2015/05/DSC_8597-1024x678.jpg (31.08.2017.)
49. https://e-radionica.com/hr/blog/2015/08/19/buzzer/ (31.08.2107.)
50. http://www.saperel.com/image/cache/data/Proizvodi/Senzori/Foto%20otpornik%20mali/09088-02-L-500x500.jpg (31.08.2017.)
51. https://hr.wikipedia.org/wiki/Fotootpornik (31.08.2017.)
52. http://ai2.appinventor.mit.edu/?locale=en#4658944700841984 (31.08.2017.)
53. http://appinventor.mit.edu/explore/ (31.08.2017.)
54. https://www.veleri.hr/?q=node/1133# (31.08.2017.)
55. https://www.arduino.cc/en/Main/Software (31.08.2107.)
56. http://arduino.bloger.index.hr/default.aspx (13.09.2017.)
57. https://cdn.sparkfun.com/datasheets/Sensors/Proximity/HCSR04.pdf (15.09.2017.)
58. https://cdn-learn.adafruit.com/downloads/pdf/pir-passive-infrared-proximity-motion-sensor.pdf (15.09.2017.)
59. http://www.micropik.com/PDF/SG90Servo.pdf (15.09.2017.)
60. http://www.micropik.com/PDF/dht11.pdf (15.09.2017.)
61. https://e-radionica.com/hr/ (15.09.2017.)
62. https://www.aliexpress.com/ (15.09.2017.)

POPIS SLIKA

Slika 1. – Koncept pametne kuće 5
Slika 2. – Upravljačka jedinica pametne kuće 6
Slika 3. – Integrirani sustav pametne kuće 7
Slika 4. – Signal koji šalje X-10 protokol 12
Slika 5. – Povezivanje sa X-10 protokolom 13
Slika 6. – LonWorks model spajanja 15
Slika 7. – Primjer povezivanja sa CEBus protokolom 16
Slika 8. – Povezivanje pametne kuće sa Konnex protokolom 17
Slika 9. – Pametna kuća 20
Slika 10. – Arduino UNO mikrokontroler 23
Slika 11. - Bluetooth HC 06 modul 24
Slika 12. – Spajanje Bluetooth modula sa Arduino uređajem 25
Slika 13. - Android uređaj 26
Slika 14. – Ultrazvučni modul 27
Slika 15. – Povezivanje Ultrazvučnog modula sa Arduinom 28
Slika 16.– PIR senzor 29
Slika 17. – Princip rada PIR senzora 30
Slika 18. – Spajanje PIR senzora sa Arduino uređaj 31
Slika 19. - Servo motor 32
Slika 20. – Spajanje servo motora sa Arduino uređajem 33
Slika 21. – DHT 11 temperaturni senzor 34
Slika 22. – Povezivanje DHT 11 senzora sa Arduinom 35
Slika 23. -Svjetleća dioda 36
Slika 24.– Otpornik 37
Slika 25. – Zujalica 38
Slika 26. – Fotootpornik 39
Slika 27. – Nacrt unutrašnjeg sloja modela pametne kuće 40
Slika 28. – Prednji pogled na konstrukciju modela pametne kuće 41
Slika 29. – Stražnji pogled na konstrukciju modela pametne kuće 41
Slika 30. – Bočni pogled sa lijeve strane na model pametne kuće 42
Slika 31. – Bočni pogled sa desne strane na model pametne kuće 42
Slika 32. – Nacrt vanjskog sloja modela pametne kuće 43

Slika 33. – Nacrt smještaja elektroničkih komponenata na modelu pametne kuće 44
Slika 34. – Ocrtavanje linija za izrezivanje na šperploči 45
Slika 35. – Piljenje šperploče sa multifunkcijskim alatom 46
Slika 36. – Bojanje unutarnje strane modela pametne kuće 46
Slika 37. – Bojanje vanjske strane modela pametne kuće 47
Slika 38. – Lijepljenje šperploče 47
Slika 39. – Gotov unutarnji oblik modela pametne kuće 48
Slika 40. – Elektronički dio modela pametne kuće 49
Slika 41. – Spajanje elektroničkih komponenata na eksperimentalnu pločicu 50
Slika 42. – Otvoren gotov model pametne kuće 51
Slika 43. – Zatvoren gotov model pametne kuće 51
Slika 44. – Početni dio programa 52
Slika 45. – Drugi dio programa (Void setup) 53
Slika 46. – Početak glavne petlje 54
Slika 47. – Programski dio za automatski način rada 55
Slika 48. – Slanje očitanja senzora serijskom komunikacijom 56
Slika 50. – MIT App Inventor razvojno okruženje 59
Slika 51. – MIT AI2 Companion aplikacija za testiranje 60
Slika 52. – Početni zaslon aplikacije 61
Slika 53. – Odabir načina rada 62
Slika 54. – Automatski način rad 62
Slika 55. – Dizajn ručnog režima rada 63
Slika 56. – Dizajnerski dio upravljanja rasvjetom 64
Slika 58. – Dizajnerski dio za Upravljanje garažnim vratima 65
Slika 59. – Dizajnerski dio za Upravljanje alarmnim sustavom 66
Slika 60. – Logički dio u MIT App Inventoru 67
Slika 61. – Logički dio početne stranice 68
Slika 62. – Logički dio spajanja na Bluetooth uređaj 68
Slika 63. – Logički dio nakon uspješnog spajanja na Bluetooth uređaj 69
Slika 64. – Logički dio za Upravljanje rasvjetom 70
Slika 65. – Logički dio upravljanja klimatizacijom 70
Slika 66. – Logički dio Upravljanja garažnim vratima 71
Slika 67. – Logički dio Upravljanja alarmnim sustavom 72
Slika 68. – Logički dio očitanja senzora u aplikaciji 73

POPIS TABLICA

Tablica 1. – Karakteristike Arduino UNO uređaja 23
Tablica 2. – Karakteristike ultrazvučnog senzora HC-SR04 28
Tablica 3. – Karakteristike PIR senzora HC-SR501 30
Tablica 4. – Karakteristike Servo motora SG90 32
Tablica 5. – Karakteristike DHT 11 temperaturnog senzora..... 35
Tablica 6. – Korištene elektroničke komponente za izradu modela pametne kuće 76
Tablica 7. – Potreban materijal za izradu modela pametne kuće 77
Tablica 8. – Troškovi izrade modela pametne kuće..... 77

POPIS PRILOGA

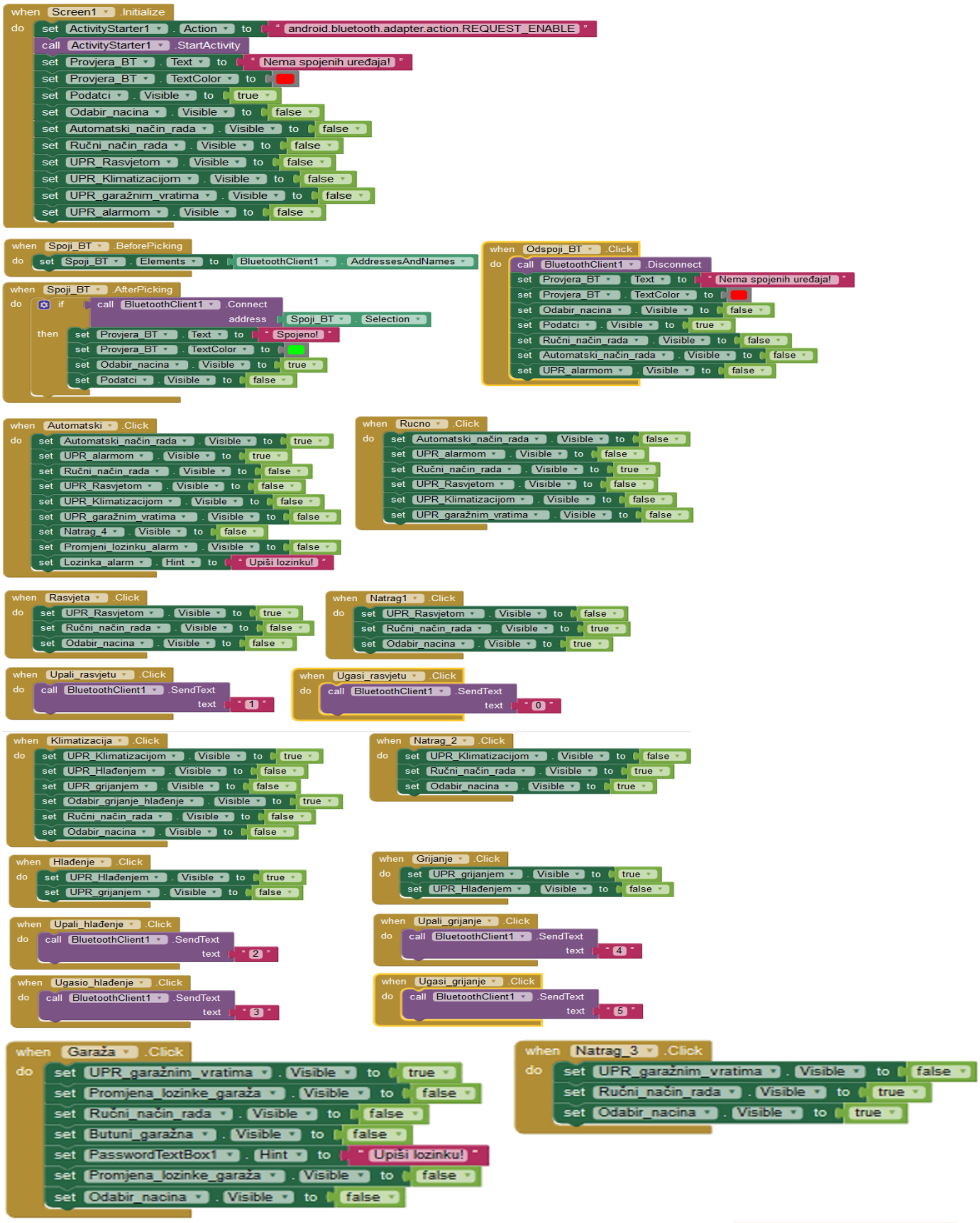
Prilog 1. – Program za upravljanje Arduino uređajem

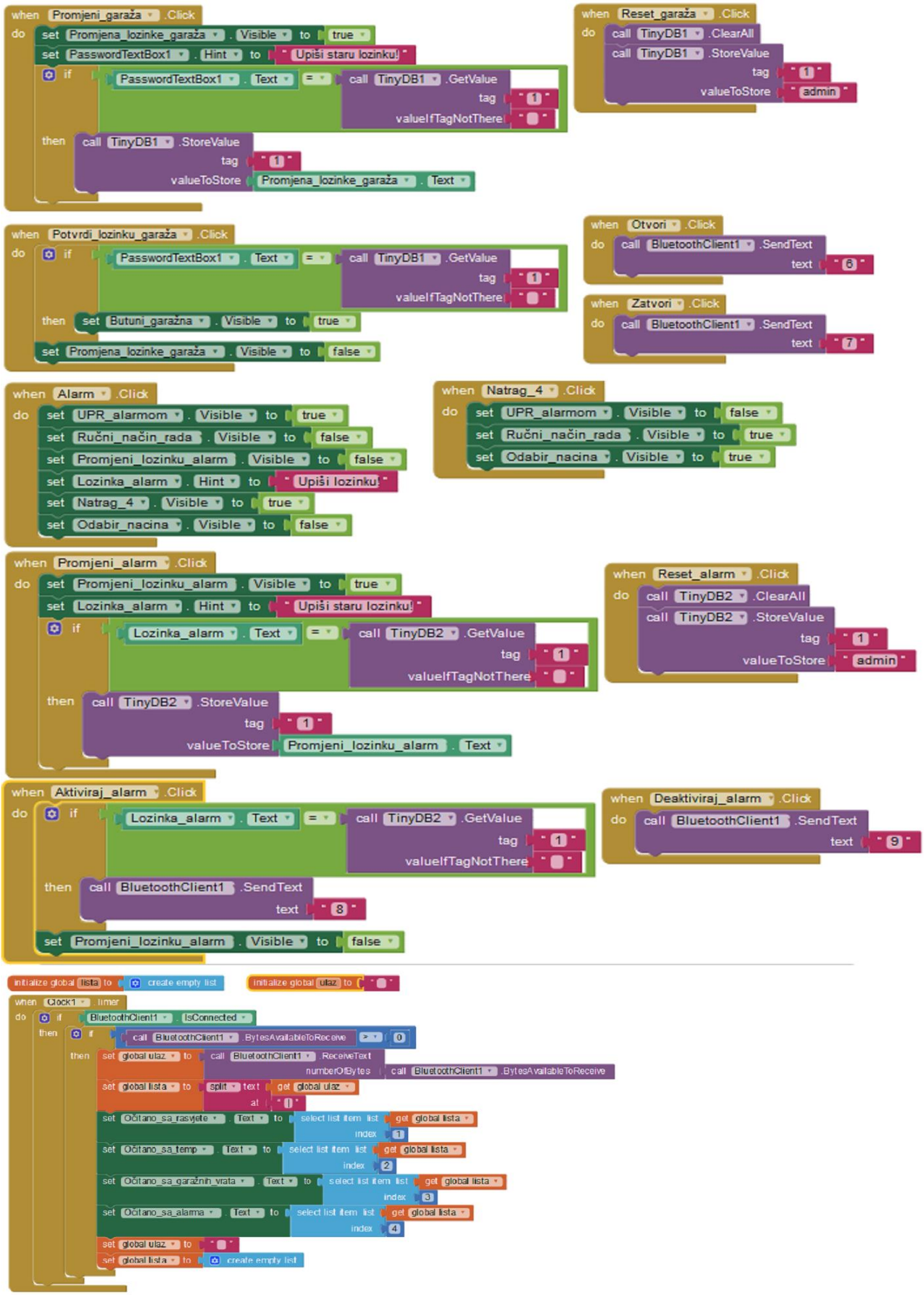
```
#include <Servo.h>                          // uključi library za servo motror
#include <dht.h>                            // uključi library za temperaturni senzor
#define dht_dpin A0                         // spoji temperaturni senzor na analogni ulaz A0
dht DHT;                                    // potrebno je defiinirat zbog librarya
Servo servo;                                // potrebno je defiinirat zbog librarya
int sklopka = 8;                            // sklopka je spojena na priključak 8
int poz = 0;                                // varijabla za for petlju za upravljanje servo motorom
int foto = A1;                              // foto otpornik spojen na priključak A1
int ocitanjefo;                             // varijabla za spremanje očitanja na fotootporniku
int jacinaled;                              // varijabla za pretvorbu očitanja na fotootporniku u jačinu led diode
int led_rasvjeta =3;                        // led_rasvjeta je spojena na priključak 3
int led_grijanje =4;                        // led_grijanje je spojena na priključak 4
int led_hladenje = 5;                       // led_hlađenje je spojena na priključak 5
int led_alarm = 6;                          // led_alarm je spojena na priključak 6
int zuj = 7;                                // zujalica je spojena na priključak 7
int echo = 12;                              // echo priključak sa ultrazvučnog modula je spojen na priključak 12
int trig = 13;                              // trig priključak sa ultrazvučnog modula je spojen na priključak 13
int ocitanjeUZV = 0;                        // varijabla za spremanje očitanja na ultrazvučnom modulu
int udaljenost;                             // varijabla za pretvorbu ocitanja na ultrazvučnom modulu u udaljenost od objekta
const int inRasvjeta = 10;                  // PIR senzor rasvjete spojen na priključak 10
const int PIRgaraza = 2;                    // PIR senzor garaznihvrata spojen na priključak 2
int ocitanjeIN;                             // varijabla za spremanje očitanja na PIR senzoru rasvjete
int ocitanjeG;                              // varijabla za spremanje očitanja na PIR senzoru garaznih vrata
char data = 0;                              // varijabla data za spremanje bluetooth podataka
int rasvjeta = 0;                           // varijabla za spremanje stanja rasvjete
int garaza = 0;                             // varijabla za spremanje stanja garaznih vrata
int alarm;                                  // varijabla za spremanje stanja alarma
int temperatura;                            // varijabla za spremanje ocitanja temperature
void setup() {
  servo.attach(9);                          // spajanje servo motora na priključak 9
  pinMode(led_rasvjeta,OUTPUT);             // definiranje led rasvjete kao izlazni signal
  pinMode(led_grijanje,OUTPUT);             // definiranje led grijanja kao izlazni signal
  pinMode(led_hladenje,OUTPUT);             // definiranje led hlađenja kao izlazni signal
  pinMode(led_alarm,OUTPUT);                // definiranje led alarma kao izlazni signal
  pinMode(inRasvjeta,INPUT);                // definiranje PIR senzora rasvjete kao ulazni signal
  pinMode(PIRgaraza,INPUT);                 // definiranje PIR senzora gražnih vrata kao ulazni signal
  pinMode(sklopka,INPUT_PULLUP);            // definiranje sklopke kao ulazni signal
  pinMode(zuj,OUTPUT);                      // definiranje zujalice kao izlazni signal
  pinMode(trig,OUTPUT);                     // definiranje trig kao izlazni signal
  pinMode(echo,INPUT);                      // definiranje echo kao ulazni signal
  digitalWrite(led_alarm,LOW);              // postavljanje led alarm u nisko stanje
  digitalWrite(led_rasvjeta,LOW);           // postavljanje led rasvjeta u nisko stanje
  digitalWrite(led_grijanje,LOW);           // postavljanje led grijanje u nisko stanje
  digitalWrite(led_hladenje,LOW);           // postavljanje led hlađenje u nisko stanje
  servo.write(0);                           // servo motor se postavlja u 0
  Serial.begin(9600);                       // započni serijsku komunikaciju na frekvenciji od 9600
}
 void loop() {
   //ALARMNI SUSTAV
   digitalWrite(trig,HIGH);                 // postavi priključak trig u visoko stanje
   delayMicroseconds(10);                   // čekaj 10 mikro sekundi
   digitalWrite(trig,LOW);                  // postavi priključak trig u nisko stanje
   delayMicroseconds(5);                    // čekaj 5 mikro sekundi
   ocitanjeUZV = pulseIn(echo,HIGH);        // primanje reflektivnog triga i spremanje u varijablu ocitanjeUZV
   udaljenost = (ocitanjeUZV/2)/29.1;       // pretvorba reflektivnog triga u udaljenost u cm
   delay(500);                              // čekaj 0,5 sekunde za stabilizaciju senzora
   if(data=='8'){                           // ako je varijabla data = 8
     alarm = 1;                             // postavi varijablu alarm u 1
     if((udaljenost<20)&&(udaljenost>0)){   // ako je udaljenost manja od 19 cm i veća od 0
       tone(zuj,1000);                      // postavi zujalicu da pišti
       digitalWrite(led_alarm,HIGH);        // postavi led_alarm u visoko stanje
       delay(500);                          // čekaj 0,5 sekunde
       digitalWrite(led_alarm,LOW);         // postavi led alarm u nisko stanje
       delay(500);                          // čekaj 0,5 sekunde
       digitalWrite(led_alarm,HIGH);        // postavi led alarm u visoko stanje
       }
       }
       else if(data == '9'){                // inače ako je varijabla data = 9
         alarm = 0;                         // postavi varijablu alarm u 0
         digitalWrite(led_alarm,LOW);       // postavi led alarm u nisko stanje
         noTone(zuj);                       // ugasi pištanje zujalice
         }
```

```
// AUTOMATSKI NAČIN RADA
if(digitalRead(sklopka)==LOW){                  // ako je sklopka u niskom stanju izvršava se automatski režim rada
  //AUTOMATSKO UPRAVLJANJE GARAŽNIM VRATIMA
  ocitanjeG = digitalRead(PIRgaraza);           // očitano stanje na PIR senzoru garažnih vrata postavi u varijablu ocitanj
  if(ocitanjeG == HIGH){                        // ako je očitanje na PIR senzoru garažnih vrata u visokom stanju
    if(garaza == 0){                            // ako je stanje garaza = 0
      garaza = 1;                               // postavi varijablu garaza u 1
      for(poz = 0; poz < 90; poz +=1){          // OTVARA GARAŽNA VRATA
      servo.write(poz);
      delay(15);
    }
    }
    else{                                       // inače
      garaza= 0;                                // garaza je 0
      for(poz = 90; poz >=1; poz -=1){          // ZATVARA GARAŽNA VRATA
      servo.write(poz);
      delay(15);
    }
    }
    }
    // AUTOMATSKO UPRAVLJANJE RASVJETOM
 ocitanjefo = analogRead(foto);                 // očitano stanje na fotootporniku postavi u varijablu ocitanjefo
 jacinaled = map(ocitanjefo, 0,1023, 255,0);    // pretvorba stanja na fotootporniku u jacinu led diode
 ocitanjeIN = digitalRead(inRasvjeta);          // očitano stanje na PIR senzoru rasvjete postavi u varijablu ocitanjeIN
  if(ocitanjeIN == HIGH){                       // ako je varijabla ocitanjeIN u visokom stanju
    if(rasvjeta==0){                            // ako je rasvjeta = 0
      rasvjeta = 1;                             // postavi varijablu rasvjeta u 1
      analogWrite(led_rasvjeta,jacinaled);      // postavi led rasvjeta u stanje ovisno o pretvorenom očitanju sa fotootpornika
      delay(3000);                              // čekaj 3 sekunde radi delaya PIR senzora
    }
    else {                                      // inače
      rasvjeta = 0;                             // rasvjeta = 0
      analogWrite(led_rasvjeta,0);              // postavi led rasvjetu u 0
      delay(3000);                              // čekaj 3 sekunde radi delaya PIR senzora
    }
  }
  // AUTOMATSKO UPRAVLJANJE KLIMATIZACIJOM
 DHT.readl1(dht_dpin);                          // očitaj stanje na temperaturnom senzoru
 temperatura = DHT.temperature;
 if(DHT.temperature > 25){                      // ako je očitana temperatura veća od 25 stupnjeva
  digitalWrite(led_hladenje,HIGH);              // postavi led hlađenje u visoko stanje
  digitalWrite(led_grijanje,LOW);               // postavi led grijanje u nisko stanje
    }
 else if(DHT.temperature < 20){                 // inače ako je očitana temperatura manja od 20 stupnjeva
  digitalWrite(led_grijanje,HIGH);              // postavi led grijanje u visoko stanje
  digitalWrite(led_hladenje,LOW);               // postavi led hlađenje u nisko stanje
    }
  // SLANJE OČITANJA NA ANDROID UREĐAJ
 if(rasvjeta == 1){                             // ako je varijabla stanje u 1
  Serial.print("Uključena rasvjeta");           // serijskom komunikacijom šalji da je rasvjeta upaljena
  Serial.print("|");                            // potrebno radi razdvajanja i pozicioniranja teksta na android uređaju
    }
 else if(rasvjeta ==0){                         // inače ako je stanje u 0
  Serial.print("Isključena rasvjeta");          // serijskom komunikacijom šalji da je rasvjeta ugašena
  Serial.print("|");                            // potrebno radi razdvajanja i pozicioniranja teksta na android uređaju
    }
  Serial.print(temperatura);                    // serijskom komunikacijom šalji očitanje temperaturnog otpornika
  Serial.print("°C");                           // serijskom komunikacijom šalji mjernu jedinicu
  Serial.print("|");                            // potrebno radi razdvajanja i pozicioniranja teksta na android uređaju

 if(garaza==1){                                 // ako je varijabla garaza u 1
  Serial.print("Otvorena garažna vrata");       // serijskom komunikacijom šalji da su garažna vrata otovrena
  Serial.print("|");                            // potrebno radi razdvajanja i pozicioniranja teksta na android uređaju
    }
 else if(garaza==0){                            // inače ako je varijabla garaza u 0
  Serial.print("Zatvorena garažna vrata");      // serijskom komunikacijom šalji da su garažna vrata zatvorena
  Serial.print("|");                            // potrebno radi razdvajanja i pozicioniranja teksta na android uređaju
    }
 if(alarm==1){                                  // ako je varijabla alarm u 1
  Serial.print("Aktiviran alarm");              // serijskom komunikacijom šalji da je alarm upaljen
  Serial.print("|");                            // potrebno radi razdvajanja i pozicioniranja teksta na android uređaju
    }
 else if(alarm == 0){                           // inače ako je varijabla alarm u 0
   Serial.print("Deaktiviran alarm");           // serijskom komunikacijom šalji da je alarm ugašen
   Serial.print("|");                           // potrebno radi razdvajanja i pozicioniranja teksta na android uređaju
    }
   delay(3000);                                 // čeka se 3 sekunde radi slanja očitanja svakih 3 sekunde
```

```
if(Serial.available()>0){                    // ako postoji serijska komunikacija i veća je od 0
data = Serial.read();                        // u varijablu data spremaj podatke primljene serijskom komunikacijom

// RUČNI REŽIM RADA
if(digitalRead(sklopka)==HIGH){              // ako je sklopka u visokom stanju izvršava se ručni režim rada
 // RUČNO UPRAVLJANJE RASVJETOM
ocitanjefo = analogRead(foto);               // očitano stanje na fotootporniku postavi u varijablu ocitanjefo
jacinaled = map(ocitanjefo, 0,1023, 255,0);  // pretvorba stanja na fotootporniku u jacinu led diode
if(data == '1'){                             // ako je varijabla data = 1
   analogWrite(led_rasvjeta,jacinaled);      // postavi led rasvjeta u stanje ovisno o pretvorenom očitanju sa fotootpornika
 }
 else if (data == '0'){                      // inače ako je varijabla data = 0
   analogWrite(led_rasvjeta,0);              // postavi led rasvjeta u 0
 }
 // RUČNO UPARAVLJANJE KLIMATIZACIJOM
 if(data == '2'){                            // ako je varijabla data = 2
   DHT.read11(dht_dpin);                     // očitaj stanje na temperaturnom senzoru
   digitalWrite(led_hladenje,HIGH);          // postavi led hlađenje u visoko stanje
   digitalWrite(led_grijanje,LOW);           // postavi led hladenje u nisko stanje
 }
 else if (data == '3'){                      // inače ako je varijabla data = 3
   digitalWrite(led_hladenje,LOW);           // postavi led hlađenje u nisko stanje
 }
 if(data == '4'){                            // ako je varijabla data = 4
   DHT.read11(dht_dpin);                     // očitaj stanje na temperaturnom senzoru
   digitalWrite(led_grijanje,HIGH);          // postavi led grijanje u visoko stanje
   digitalWrite(led_hladenje,LOW);           // postavi led hladenje u nisko stanje
 }
 else if (data == '5'){                      // inače ako je varijabla data = 5
   digitalWrite(led_grijanje,LOW);           // postavi led grijanje u nisko stanje
 }
 // RUČNO UPRAVLJANJE GARAŽNIM VRATIMA
 if(data == '6'){                            // ako je varijabla data = 6
   for(poz = 0; poz < 90; poz +=1){          // OTVORI GARAŽNA VRATA
     servo.write(poz);
     delay(15);
   }
 }
```

Prilog 2. – Logički dio aplikacije za upravljanje modelom pametne kuće




when Promjeni_garaža .Click
do set Promjena_lozinke_garaža . Visible to true
set PasswordTextBox1 . Hint to "Upiši staru lozinku!"
if PasswordTextBox1 . Text = call TinyDB1 .GetValue tag "1" valueIfTagNotThere ""
then call TinyDB1 .StoreValue tag "1" valueToStore Promjena_lozinke_garaža . Text
when Reset_garaža .Click
do call TinyDB1 .ClearAll
call TinyDB1 .StoreValue tag "1" valueToStore "admin"
when Potvrdi_lozinku_garaža .Click
do if PasswordTextBox1 . Text = call TinyDB1 .GetValue tag "1" valueIfTagNotThere ""
then set Butuni_garažna . Visible to true
set Promjena_lozinke_garaža . Visible to false
when Otvori .Click
do call BluetoothClient1 .SendText text "6"
when Zatvori .Click
do call BluetoothClient1 .SendText text "7"
when Alarm .Click
do set UPR_alarmom . Visible to true
set Ručni_način_rada . Visible to false
set Promjeni_lozinku_alarm . Visible to false
set Lozinka_alarm . Hint to "Upiši lozinku!"
set Natrag_4 . Visible to true
set Odabir_nacina . Visible to false
when Natrag_4 .Click
do set UPR_alarmom . Visible to false
set Ručni_način_rada . Visible to true
set Odabir_nacina . Visible to true
when Promjeni_alarm .Click
do set Promjeni_lozinku_alarm . Visible to true
set Lozinka_alarm . Hint to "Upiši staru lozinku!"
if Lozinka_alarm . Text = call TinyDB2 .GetValue tag "1" valueIfTagNotThere ""
then call TinyDB2 .StoreValue tag "1" valueToStore Promjeni_lozinku_alarm . Text
when Reset_alarm .Click
do call TinyDB2 .ClearAll
call TinyDB2 .StoreValue tag "1" valueToStore "admin"
when Aktiviraj_alarm .Click
do if Lozinka_alarm . Text = call TinyDB2 .GetValue tag "1" valueIfTagNotThere ""
then call BluetoothClient1 .SendText text "8"
set Promjeni_lozinku_alarm . Visible to false
when Deaktiviraj_alarm .Click
do call BluetoothClient1 .SendText text "9"
initialize global lista to create empty list
initialize global ulaz to ""
when Clock1 .Timer
do if BluetoothClient1 .IsConnected
then if call BluetoothClient1 .BytesAvailableToReceive > 0
then set global ulaz to call BluetoothClient1 .ReceiveText numberOfBytes call BluetoothClient1 .BytesAvailableToReceive
set global lista to split text get global ulaz at "|"
set Očitano_sa_rasvjete . Text to select list item list get global lista index 1
set Očitano_sa_temp . Text to select list item list get global lista index 2
set Očitano_sa_garažnih_vrata . Text to select list item list get global lista index 3
set Očitano_sa_alarma . Text to select list item list get global lista index 4
set global ulaz to ""
set global lista to create empty list