SUSTAV NADZORA I UPRAVLJANJA PAMETNOM KUĆOM

Jugović, Luka

Undergraduate thesis / Završni rad

2022

Degree Grantor / Ustanova koja je dodijelila akademski / stručni stupanj: **University of Rijeka, Faculty of Engineering / Sveučilište u Rijeci, Tehnički fakultet**

Permanent link / Trajna poveznica: https://urn.nsk.hr/urn:nbn:hr:190:809174

Rights / Prava: Attribution 4.0 International/Imenovanje 4.0 međunarodna

Download date / Datum preuzimanja: **2024-05-23**

Repository / Repozitorij:

Repository of the University of Rijeka, Faculty of Engineering

SVEUČILIŠTE U RIJECI

TEHNIČKI FAKULTET

Preddiplomski stručni studij elektrotehnike

Završni rad

SUSTAV NADZORA I UPRAVLJANJA PAMETNOM KUĆOM

Rijeka, rujan, 2022.

Luka Jugović

0069071567

SVEUČILIŠTE U RIJECI

TEHNIČKI FAKULTET

Preddiplomski stručni studij elektrotehnike

Završni rad

SUSTAV NADZORA I UPRAVLJANJA PAMETNOM KUĆOM

Mentor: prof. dr. sc. Dario Matika

Rijeka, rujan, 2022.

Luka Jugović

0069071567

SVEUČILIŠTE U RIJECI
TEHNIČKI FAKULTET
POVJERENSTVO ZA ZAVRŠNE ISPITE

Rijeka, 16. ožujka 2022.

Zavod: **Zavod za automatiku i elektroniku**
Predmet: **Osnove automatske regulacije**
Grana: **2.03.06 automatizacija i robotika**

ZADATAK ZA ZAVRŠNI RAD

Pristupnik: **Luka Jugović (0069071567)**
Studij: Preddiplomski stručni studij elektrotehnike

Zadatak: **Sustav nadzora i upravljanja pametnom kućom/Smart home monitoring and control system**

Opis zadatka:

U završnom rau obraditi obraditi internet stvari IoT (Internet of things) i dati prijedlog za korištenje IoT-a u "pametnoj kući" sa trenutno dostupnim tehnologijama i mogućnostima njihove implementacije u Hrvatskoj. Aanalizirati protokol, komunikacijske tehnologije i sigurnosti sustava.

Rad mora biti napisan prema Uputama za pisanje diplomskih / završnih radova koje su objavljene na mrežnim stranicama studija.

Zadatak uručen pristupniku: 21. ožujka 2022.

Mentor:

Prof. dr. sc. Dario Matika

Predsjednik povjerenstva za završni ispit:

Prof. dr. sc. Saša Vlahinić

IZJAVA

Sukladno članku 8. Pravilnika o završnom radu, završnom ispitu i završetku stručnih studija Tehničkog fakulteta u Rijeci od 1. veljače 2020., izjavljujem da sam samostalno izradio završni rad prema zadatku preuzetom dana 21. ožujka 2022.

Rijeka, 20. rujna 2022. _____

Luka Jugović

SAŽETAK

Nove sigurnosne ranjivosti i zakrpe istih u IoT, kao i u drugim informatičkim branšama, će se uvijek pojavljivati. Sigurnost i privatnost najviše ovise o tome koliko truda uložimo u iste da bismo smanjili prijetnje. Današnji protokoli su dovoljno sigurni da zadovolje sigurnost prosječne pametne kuće. Bitno je da su tehnologije otvorenog koda zbog potrebe privatnosti. MQTT protokol je značajno prihvatljiviji od HTTP-a za IoT svrhe. openHAB je više nego dovoljan kao server za upravljanje pametnom kućom.
KLJUČNE RIJEČI: MQTT, pametna kuća, IoT, Internet of Things, openHAB

SUMMARY

New security vulnerabilities and patches for them in IoT, as well as in other IT industries, will always appear. Security and privacy mostly depends on how much effort we put into them to reduce threats. Today's protocols are secure enough to satisfy the security of the average smart home. It is essential that the technologies are open source due to the need for privacy. The MQTT protocol is significantly more acceptable than HTTP for IoT purposes. openHAB is more than enough as a smart home management server.
KEY WORDS: MQTT, smart house, IoT, Internet of things, openHAB

SADRŽAJ

1. UVOD 1
2. INTERNET STVARI (IOT) 2
2.1 Primjena IoT-a 2
3. PAMETNA KUĆA 3
3.1. Povijest pametne kuće 4
3.2. Primjena pametne kuće 5
4. BLUETOOTH 6
5. BEŽIČNA MREŽA WI-FI 8
5.1. Komponente Wi-Fi mreže 8
5.1.1 Modem 8
5.1.2 Usmjerivač (ruter) 9
5.1.3 Mrežna kartica 10
5.1.4 Pristupna točka (access point) 11
5.1.5 Antene 11
5.1.6 Repetitori 12
6. ODABIR TEHNOLOGIJE UPRAVALJANJA PAMETNOM KUĆOM 13
6.1 Instalacija openHAB programa na Raspberry-Pi 14
7. MQTT PROTOKOL 15
7.1. MQTT posrednici 15
7.2 MQTT autentifikacija 16
7.2.1 Identifikacijska šifra klijenta 16
7.2.2 Korisničko ime i zaporka 16
7.2.3 Certifikati klijenata 16
7.3 MQTT sigurnost 16
7.3.1 TLS enkripcija 17
7.3.2 Tekstualna enkripcija 17
7.4 MQTT protokol u odnosu na HTTP protokol 17
8. IZRAĐIVANJE UREĐAJA ZA SENZOR POKRETA, VLAŽNOSTI I TEMPERATURE ZRAKA I DIMA 18
8.1 Senzor dima 18
8.2 Senzor vlažnosti i temperature zraka 18
8.3 Senzor pokreta 19

8.4 Shema spajanja svih komponenti na D1 Wemos mini 20
8.5 Spajanje pametnih uređaja na mrežu 21
8.6 Upute za dodavanje pametne žarulje Hue marke Philips 21
8.7. Dodavanje uređaja za senzor pokreta, vlažnosti i temperature zraka i dima na openHAB. 24
8.8 Uključivanje i isključivanje uređaja u određeno vrijeme 25
8.9 Uključivanje Svjetla kad senzor pokreta detektira pokret 26
8.10 Dodavanje uređaja za senzor pokreta, vlažnosti i temperature zraka i dima na openHAB. 27
8.11 Ranjivosti korištenja glasovnih naredbi za upravljanje 27
8.12 Odvajanje pametnih uređaja od interneta 27
8.13 Korištenje Amazon Alexe lokalno 28
8.14 Postavljanje zasebne virtualne lokalne mreže na dd-wrt ruteru 29
9. ZAKLJUČAK 30
10. POPIS IZVORA 31
PRILOZI 33
OPENHAB RJEČNIK 33
Programski kod u Arduino (C++) jeziku 34

1. UVOD

U današnje vrijeme svjedočimo porastu brzine interneta zbog povećane upotrebe podataka jer je potrebna kako bi se što veća količina podataka mogla prenijeti, što rezultira nastankom novih i kompleksnijih aplikacija koje se koriste u modernom svijetu za zabavu ali i za pomoć u industriji i kućanstvima.

Danas se dostupna tehnologija koristi za automatizaciju određenih elemenata u kućama, stoga te kuće nazivamo pametnim kućama. Omogućava se automatizacija upravljanja što korisniku ove vrste sustava olakšava upravljanje raznim uređajima u kućanstvu.

Daljinskim upravljanjem omogućava se kontrola raznih sustava pametne kuće poput video nadzora, senzora pokreta, regulacija grijanja i mnoštvo drugih.

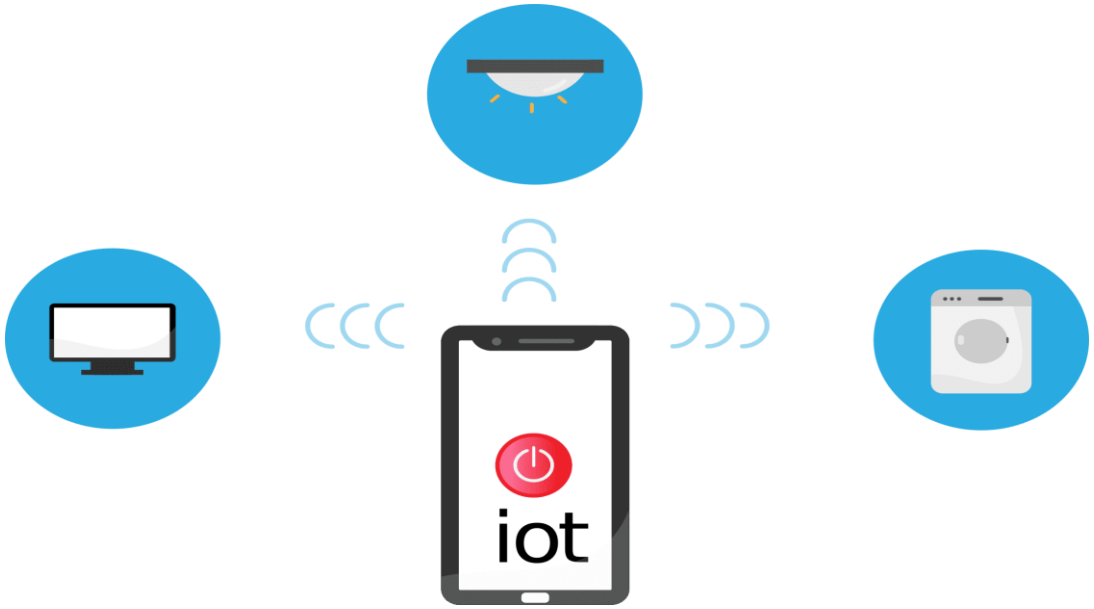
Modernizacija tehnologije znatno olakšava podešavanje raznih uređaja u sustavu pametne kuće. Tako na primjer u sustavu uređaji za regulaciju grijanja putem termostata ujutro i navečer zamijenjeni su uređajima poput tableta i pametnih telefona. Elementima pametne kuće možemo upravljati već postojećim aplikacijama koje su povezive putem Wi-Fi, Bluetooth, mobilne mreže i drugih protokola ili gotovim modulima koje možemo kupiti online ili u trgovinama. Sve je više uređaja koji imaju mogućnost povezivosti i upravljanja na daljinu kako u kućanstvima tako i u industriji.

Prvo poglavlje završnog rada referirati će se na Iot tehnologiji , njezinoj definiciji, ali i primjeni u sustavu pametne kuće. Naredno poglavlje pojašnjava definiciju pametne kuće, njezinu izgradnju tokom godina i primjenu. Nadalje, iduća poglavlja bazirati će se na sustave Bluetooth i Wi-Fi mreže koje se primjenjuju u pametnim kućama, ali i na odabir tehnologije upravljanje pametnom kućom koja koristi program openHub. Mqtt protokol i izrađivanje uređaja senzora pokreta, vlažnosti i temperatura zraka i dima su preostala poglavlja koja će se objasniti u ovome završnom radu.

2. INTERNET STVARI (IOT)

„Internet stvari“ možemo definirati kao veliku mrežu koja se bežičnom povezanosti proteže u bilo kojem proizvodu.[1] Pojam obuhvaća uređaje čije se karakteristike ističu u domeni povezivosti uređaja u jednu veliku cjelinu. Time je omogućena maksimalna komunikacija sa korisnikom. Pomoću pametne tehnologija uređaji poput pisača, hladnjaka, perilice rublja, klima uređaja koriste se u kućanstvima, te postaju sve više komercijalno dostupni i cijena im je s godinama sve pristupačnija.

Iot mreža pomoću koje se mogu regulirati razni kućanski uređaji poput perilice rublja, rasvjete i televizora je prikazana na slici 2.1.



Slika 2.1. IoT mreža[2]

2.1 Primjena IoT-a

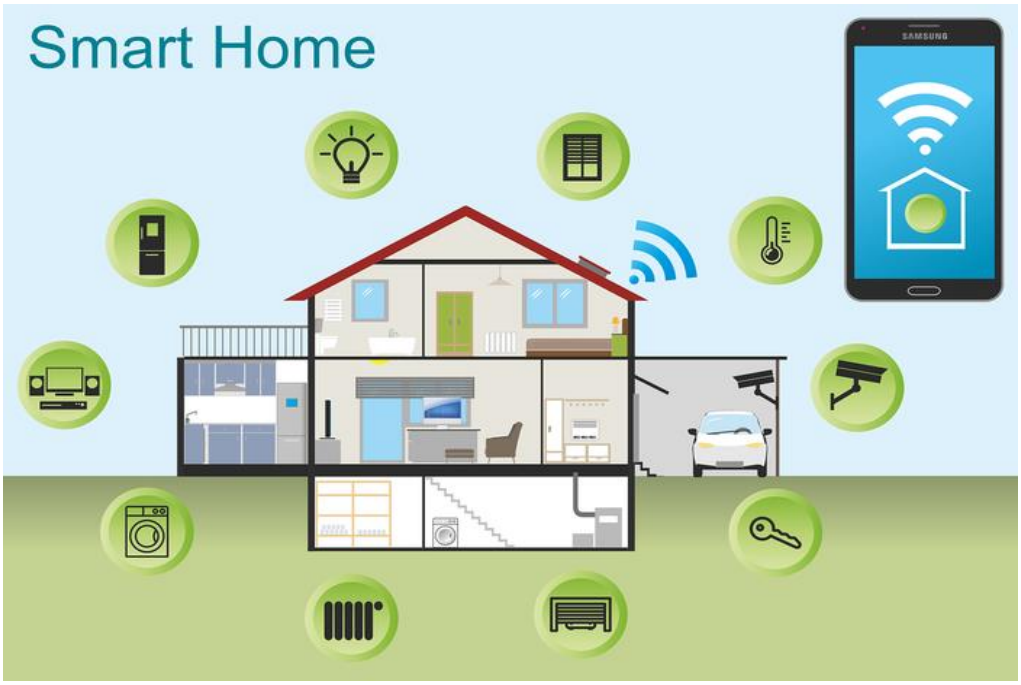
Najzastupljeniji je među privatnim potrošačima, te čini veliki postotak uređaja dostupnih na trenutnoj mreži. Ova vrsta sustava u pametnoj kući je najčešće primjenjena kod senzora pokreta, video nadzora, sustava upravljanja centralnim grijanjem i rasvjetom, otvaranja vrata/prozora, te video nadzora. Sustav kada je uključen u slučaju detekcije pokreta obavještava korisnika o tome.

[1] https://aws.amazon.com/what-is/iot/ (preuzeto 17.09.2022.)

[2] https://www.pngwing.com/en/free-png-vwfwr/ (preuzeto 13.09.2022.)

3. PAMETNA KUĆA

Pojam pametne kuće definira se kao sustav koji obuhvaća podatkovnu mrežu koja spaja značajne električne uređaje. Centralni sustav pametne kuće omogućuje povezivanje više sustava poput sigurnosti, rasvjete, grijanja, sustava hlađenja, potrošnju tople vode. Pametna kuća sadrži inteligentni sustav koji upravlja svojim segmentima samostalno, te samim time odsustvom osobe i dalje rade ispravno. Na slici 3.1 prikazani su sustavi pametne kuće povezani mobilnim telefon putem bežične mreže.



Slika 3.1. Pametna kuća[3]

Sustav kao takav omogućava kontrolu sve dok su svi elementi inteligentne kuće spojeni na mrežu. Tako na primjer za paljenje rasvjete ispred kuće osoba nije nužno fizički potrebna, nego uz nekoliko klikova na aplikaciji koja se nalazi na njezinom pametnom telefonu određuje funkciju koju želi. Sustav pametne kuće ima niz prednosti radi konstantne povezanosti s internetom. Osnovna instalacija lako je programirana kako bi ugodila svim potrebama pojedinca.

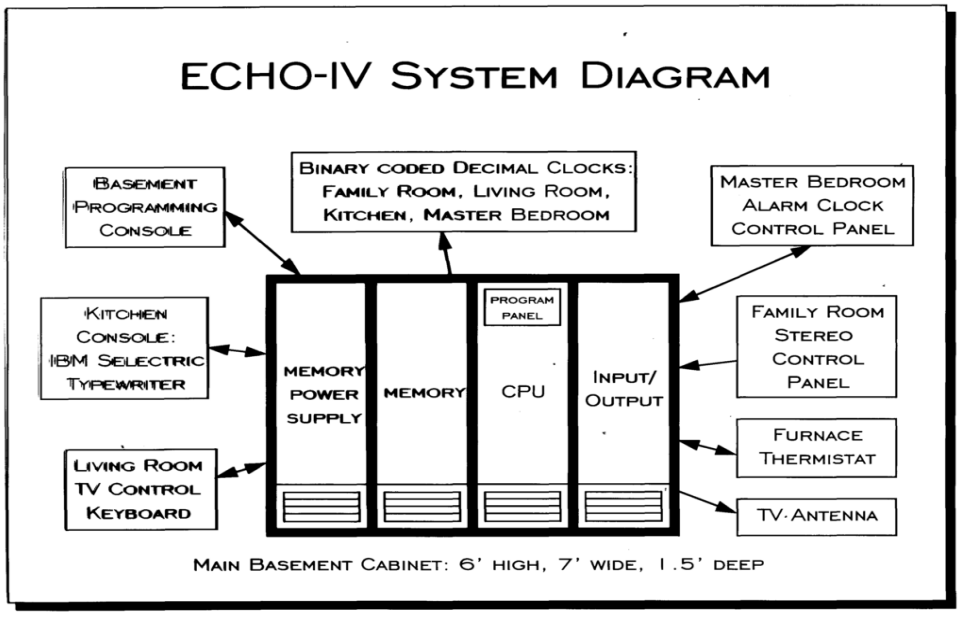
[3] https://www.nabava.net/clanci/savjeti/smarthome-sto-znaci-imati-pametnu-kucu-1545t6 (preuzeto 13.09.2022.)

3.1. Povijest pametne kuće

Sedamdesetih godina prošlog stoljeća nastaje pojam pametne kuće. Do tada se energetski učinkovito sagrađena kuća smatrala pametnom. Od tada se izraz pametne kuće mijenja uz napredak tehnologije u određenom razdoblju. Tijekom osamdesetih godina inteligentna kuća je bila ona u kojoj se preko osobnog računala upravljalo određenim sustavima u njoj.

Izum kućnih uređaja početkom dvadesetog stoljeća bio je značajno postignuće. 1901. godine napravljen je prvi usisivač koji se pokretao na motor. Nekoliko godina nakon, 1907. na tržištu se pojavljuje bolji i jednostavniji usisivač koji radi na struju.

James Sutherland 1966. godine kreira Echo IV (Electronic Computing Home Operator) koji je tada bio prvi uređaj osmišljen na principu kućne automatizacije. Uređaj je imao mogućnost upravljanja raznim uređajima u njegovom domu poput klime, digitalnih satova i televizora. Takav uređaj tih godina bio je daleko ispred tadašnje tehnologije, no Sutherland nije imao namjeru uređaj prodavati. Slika 3.2. prikazuje Echo IV sustav.



Slika 3.2. Glavni Echo IV sustav[4]

Najveći procvat kućne automatizacije dolazi krajem devedesetih razvojem novih tehnologija, te postaju pristupačnije većem broju ljudi.

[4] https://computerhistory.org/blog/the-echo-iv-home-computer-50-years-later/ *(preuzeto 13.09.2022.)*

3.2. Primjena pametne kuće

Sustav pametne kuće sastoji se od uređaja koji se dijele u veći broj podsustava koji mogu raditi zasebno i zajedno čine sustav inteligentne kuće. Protokol X-10 najčešće se koristi u takvim sustavima zbog jednostavne primjene. Osnovna oprema pametne kuće sastavljena je od LCD monitora koji su osjetljivi na podražaje, senzora, daljinskog upravljača, mikro modula i tipkala. LCD monitor u sebi sadrži modem koji omogućava komunikaciju sa telefonskom centralom. Time je omogućeno upravljanje putem pametnog telefona.

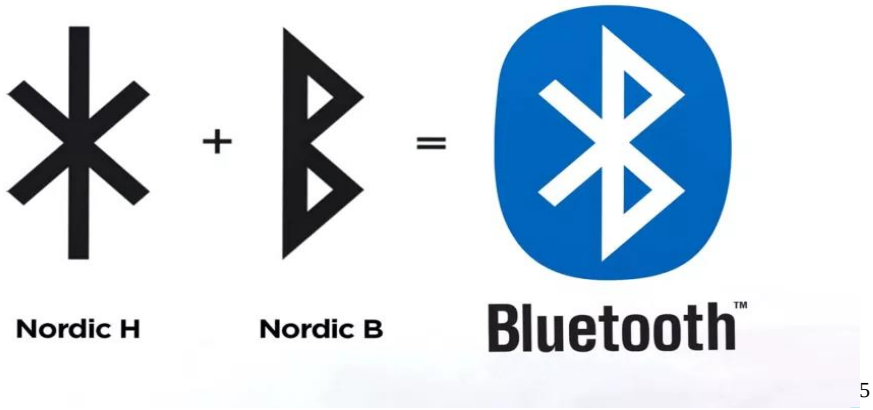
Centralna konzola također je osjetljiva na dodir, a sastoji se od primopredajnika kojemu je osnovna zadaća primanje radio frekventivnih signala preko daljinskog upravljača, te ih pretvara u X-10 signale. Kućna automatika i sigurnosni sustav spojeni su na centralnu konzolu. Software koji je u njoj programiran znatno olakšava programiranje naredbi i postavljanje ikona. Također na konzolu su spojene sigurnosne kamere na osobni zahtjev korisnika. Senzori imaju značajnu ulogu zbog smanjene mogućnosti nastanka poplave i požara, ali i provala. Također se koriste senzori za detekciju pokreta pri ulasku ili izlasku iz prostorije.

Daljinski upravljač jedan je od najvažnijih uređaja. Omogućava praktičnost upravljanja jer se njime može upravljati iz bilo kojeg dijela kuće. Mirko moduli imaju veliku važnost prilikom spajanja na gradsku mrežu, te se tako određuje adresa uređaja. Ugrađuju se strujne utičnice, no koriste se i kod standardnih prekidača za rasvjetu. U X-10 protokolu najčešće u standardiziranom paketu dolaze i tri filtera, te zauzimaju šest mjesta u razvodnoj kutiji. Na tržištu možemo pronaći opremu i protokole koji su normirani po europskim, ali i svjetskim kriterijima (ISO, IEC, EN…) što svakome korisniku može osigurati sigurnost, ali i pouzdanost.

4. BLUETOOTH

Bluetooth sustav dobio je naziv po kralju Haroldu Gormssonu koji je 958. godine ujedinio Dansku i Norvešku. Njegov „mrtvi“ zub poprima tamnoplavu boju, te sukladno tome kralj dobije nadimak Bluetooth koji se danas koristi za vrstu bežične mreže.

Godine 1994. tvrtka Ericsson sa sjedištem u Švedskoj definira ovu vrstu sustava. Bluetooth Special Interest Group je kompanija koja ima glavnu ulogu u razvitku i održavanju sustava. Tvrtka se sastoji od desetak tisuća firmi zbog primjene Bluetooth-a u velikom broju proizvoda elektroničke industrije. Na slici 4.1 objašnjen je logotip Bluetooth mreže koji je kombinacija runa „H“ (Hagall) i „B“ (Berkanan), Haraldovi inicijali.



[5]

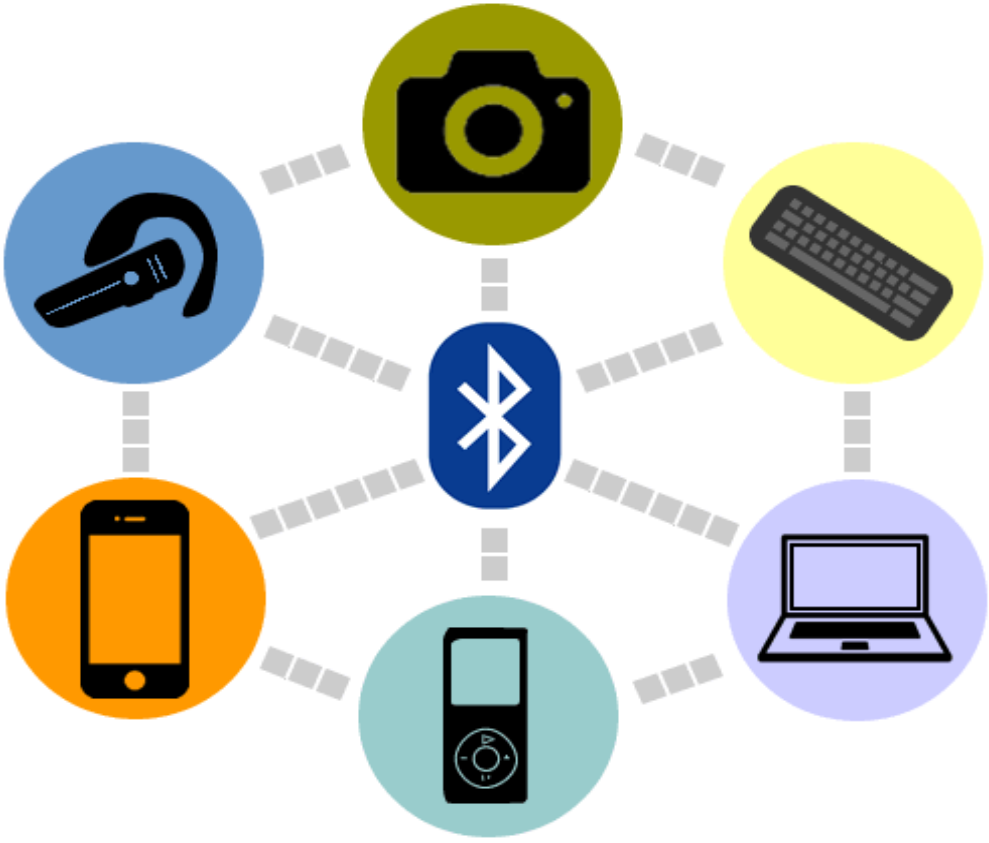
Slika 4.1. Bluetooth logo[6]

Ova vrsta sustava osigurava prijenos informacija unutar uređaja manjih udaljenosti. Uključena Bluetooth mreža sadrži adapter u kojemu se nalaze odašiljač i prijemnik. Najmanja udaljenost definirana je na 6 metara, a snaga njezinog odašiljača iznosi 1 mili-Wat. To je sasvim dovoljno za normalan rad mreže. Često se navode i drugi dosezi koji se mogu protezati i do 100 metara snage odašiljača 100 mili-Wata, što je i najveća snaga. Stoga se Bluetooth mreža još naziva i mrežom osobnog prostora. 2,45 GHz je frekventni pojas u kojemu radi Bluetooth mreža.

Njezino korištenje ne zahtjeva potrebnu licencu i doseže propusnost od 2,1 Mbps. Propusnost mreže je više nego zadovoljavajuća za poslove koji ju koriste. Primjena tehnologije bežičnog prijenosa ima širok obujam uporabe za razne uređaje, ali i sustave. Također se za ovu vrstu bežične mreže može reći da zamjenjuje USB vezu.

[5] https://www.creativebloq.com/news/bluetooth-logo-secret (preuzeto 13.09.2022.)

Bluetooth sustav osigurava vezu na principu veze podataka. Prenosi određenu strukturu podataka između raznovrsnih uređaja. Veza uključuje do osam aktivnih uređaja u isto vrijeme. Slika 4.2. pokazuje uređaje koji su spojivi putem bluetooth veze.



Slika 4.2. Bluetooth povezivanje[7]

Bluetoothom spajamo uređaje poput audio uređaja (prijenosni bluetooth zvučnici, bluetooth slušalice,…), fotoaparata koji imaju mogućnost povezivanja bluetoothom, bežične tipkovnice i bežičnog miša,…

Razvojem mreža tijekom godina, ZigBee je noviji sustav koji radi na principu Bluetooth-a, te mu u određenim dijelovima konkurira. Ova vrsta sustava pogodna je manjoj propusnosti veze i potrošnja energije je smanjena. Njezina najveća prednost je jednostavnost, ali i niska cijena cijena koja ju čini pristupačnom za veliki broj naprava.

[7] https://www.dignited.com/50134/the-updated-guide-to-bluetooth/ (preuzeto 17.09.2022.)

5. BEŽIČNA MREŽA WI-FI

„Naziv Wi-Fi označava jedan skup mreža koje se međusobno razlikuju u raznim elementima i koje su raznim varijantama standarda 802.11.“[8] Bežična lokalna mreža kojoj je namjena uporaba u prostorima poput poslovnih zgrada, te obuhvaća više poslovnih prostora u jednom mjestu. Omogućuje dobru mobilnost uređaja, bežično povezivanje računala ili drugih uređaja na mrežu. Usprkos tome mobilnost bežične mreže nije velika i u prosijeku iznosi oko 100 metara.

Prilikom spajanja na Wi-Fi mrežu potreban je adapter za standard 802.11. on uključuje transfer (odašiljač i prijamnik) za elektromagnetske signale. Prijenosna računala mogu sadržavati već ugrađene adaptere. Postoje slučajevi u kojima računalo nema potrebni ugrađeni adapter, tada se može ugraditi ili priključiti na neki USB port računala.

Wi-Fi Alliance je udruženje čiji je zaštitni znak upravo Wi-Fi. Udruženje okuplja nešto više od 300 tvrtki koje se bave proizvodnjom opreme za lokalne bežične mreže. 1991. godine NCR korporacija AT&T u Nizozemskoj izumila je Wi-Fi mrežu. Prva mreža nosila je naziv Wawe. Wawe LAN i radila brzinom od 1 do 2 Mbit/s.

5.1. Komponente Wi-Fi mreže

Komponente bežične Wi-Fi mreže sadrže modem, mrežnu karticu, usmjerivač, pristupnu točku, repetitore i antene.

5.1.1 Modem

Uređaj hardverskog umrežavanja koji pretvorene podatke u signal lako šalje i prima putem telefonske linije, satelitske veze ili kabela. Nekada su radili na principu analogne telefonske linije, te modem pretvara podatke u realnom vremenu za dvosmjernu mrežnu komunikaciju. Dovodi Internet u kuću i održava vezu s pružateljem usluge, te omogućava pristup internetu.

Računalni signali su digitalni, dok su internetski analogni. Modem demodulira analogne signale s interneta u digitalne da bi ih računalo moglo razumjeti i obrnuto. Stoga riječ modem znači *modulator-demodulator*.

Postoje razni tipovi modema, no najzastupljeniji su oni žičani i DSL modemi. Žičani su povezani pomoću koaksijalnog kabela koji je najčešće osiguran od strane pružatelja kablovske televizije (Hrvatski Telekom, A1…).

[8](M,Radovan,2008,str.273)

Izgled modema prikazan je na slici 5.1.

Slika 5.1. Modem [32]

Putem telefonske linije povezani su DSL modemi. Klasične telefonske veze danas imaju brzinu od 56 000 bit/s što je omogućeno današnjim normama. Mogu biti izrađeni u obliku elektroničke kartice koja se ugrađuje u računalo, te samostalne vanjske jedinice.

- Faks modem-prijenos računalnih podataka, slika i teksta faks uređajem.
- ADSL-prijenos podataka preko telefonske mreže i mreže kablovske televizije

5.1.2 Usmjerivač (ruter)

Funkcionira na sličan princip kao i standardni usmjerivač za žičane Ethernet mreže. Bežični usmjerivač potreban je kako bi se uspostavila bežična mreža. 802.11ac pruža dobru reprodukciju videozapisa ili igranje video igara. Usmjerivač je prikazan na slici 5.1.2.

Slika 5.2. Usmjerivač[9]

[9] https://www.links.hr/hr/router-asus-rt-ac65p-dualband-ac1750-wan-1-port-gigabit-4-port-switch-3-antene-1x-usb-bezicni-060800022 (preuzeto 13.09.2022.)

5.1.3 Mrežna kartica

Ima veliku važnost za uređaje na bežičnoj mreži, te su potrebni za svaki takav uređaj. Njegova zadaća je priključivanje uređaja na lokalnu mrežu. Pametni telefoni, tableti, prijenosna računala imaju mogućnost povezivanja bežičnim putem. Mrežna kartica za priključivanje na bežičnu mrežu prikazana je na slici 4.1.3.1.

Slika 5.3. Mrežna kartica[10]

Postoji nekoliko priključaka mrežne kartice, a to su:

- Tanki koaksijalni-koaksijalni kabel povezan je direktno na sabirnicu. Prednost ovog načina povezanosti je taj što je jeftino rješenje, dok se za nedostatak može navesti da prilikom oštećenja ili prekida mrežne kartice dolazi do potpunog prekida povezanosti mreže.
- Usukane parice-osmo žitni kabel, mogu biti STP i UTP tipa tj. otklopljenje i neotklopljene. Prednosti su jeftinija cijena vodova i aktivnih mrežnih uređaja, koja je dovela do povećanja popularnosti. Također je i lakša za instalaciju i znatno pouzdanija od prethodno navedene.
- Debeli koaksijalni-koaksijalni kabel se više ne koristi. Prekid ili oštećenje dovodi do potpunog prekida mreže. Uređaj se povezuje na mrežnu karticu pomoću 15 pinskog AUI (Attachmet Unit Interface) konektora.

[10] https://www.hgspot.hr/mrezna-kartica-tp-link-tl-wn781nd-150-mbps-wireless-n-pcie (preuzeto 13.09.2022.)

5.1.4 Pristupna točka (access point)

Koristi se kao dopuštenje bežičnoj mreži da se spoji na postojeću mrežu. Točka pristupa olakšava pristup spajanju Wi-Fi-ja ako u nekim dijelovima kuće ili poslovne zgrade postoji slaba mreža. Na slici 5.4. prikazana je pristupna točka.

Slika 5.4. Pristupna točka[11]

5.1.5 Antene

Antene funkcioniraju tako da šalju radio valove na frekvencije gdje slušni uređaji mogu primiti. Radijski odašiljači i prijemnici ugrađeni su u opremu. Na slici 5.5. je prikazana antena.

Slika 5.5. Antena[12]

[11] https://www.nabava.net/pojacivaci-wifi-signala/pristupna-tocka-tp-link-nswpac0290-eap115-5w-12v-1-fast-ethernet-rj-45-2424835-ghz-bela-cijena-57896211(preuzeto 13.09.2022.)

[12] https://www.kupindo.com/Oprema-i-delovi/70961693_2-4GHz-i-5-8GHz-ANTENA-za-hvatanje-WiFi-signala-NOVO (preuzeto 13.09.2022.)

Antene prikupljaju dolazne signale ili zračenje odlaznih signala. Na usmjerivačima Wi-Fi antene se mogu ugraditi iznova, dok su ostale ugrađene unutar kućišta uređaja.

5.1.6 Repetitori

Repetitori se koriste za proširenje pokrivenosti Wi-Fi mreže. Rade na principu da nakon primitka postojećih Wi-Fi signala pojačava signal i prenosi pojačan signal. Repetitor na jednostavan način može udvostručiti pokrivenost Wi-Fi mreže. Kod same instalacije mora biti izravno povezan sa usmjerivačem. Slika 5.6. prikazuje repetitor.

Slika 5.6. Repetitor[13]

[13] https://www.mall.hr/pojacalo-wi-fi-signala/emos-wifi-repetitor-300m-u25 (skinuto 13.09.2022.)

6. ODABIR TEHNOLOGIJE UPRAVALJANJA PAMETNOM KUĆOM

Program openHUB je otvorenog koda za automatizirano upravljanje pametnom kućom. Logo openHAB programa prikazan je na slici 6.1.

Slika 6.1. openHAB logo[14]

Podržava preko 200 različitih tehnologija i sustava. Moguće je podesiti pravila, vremenski i događajno okinuti algoritme, skripte, akcije i obavijesti i upravljati glasom. openHAB se pokreće na Javi, stoga je podržan na plaformama kao što su Linux, macOS, Windows, Raspberry Pi, Docker, Synology i ostale platforme koje podržavaju Javu.

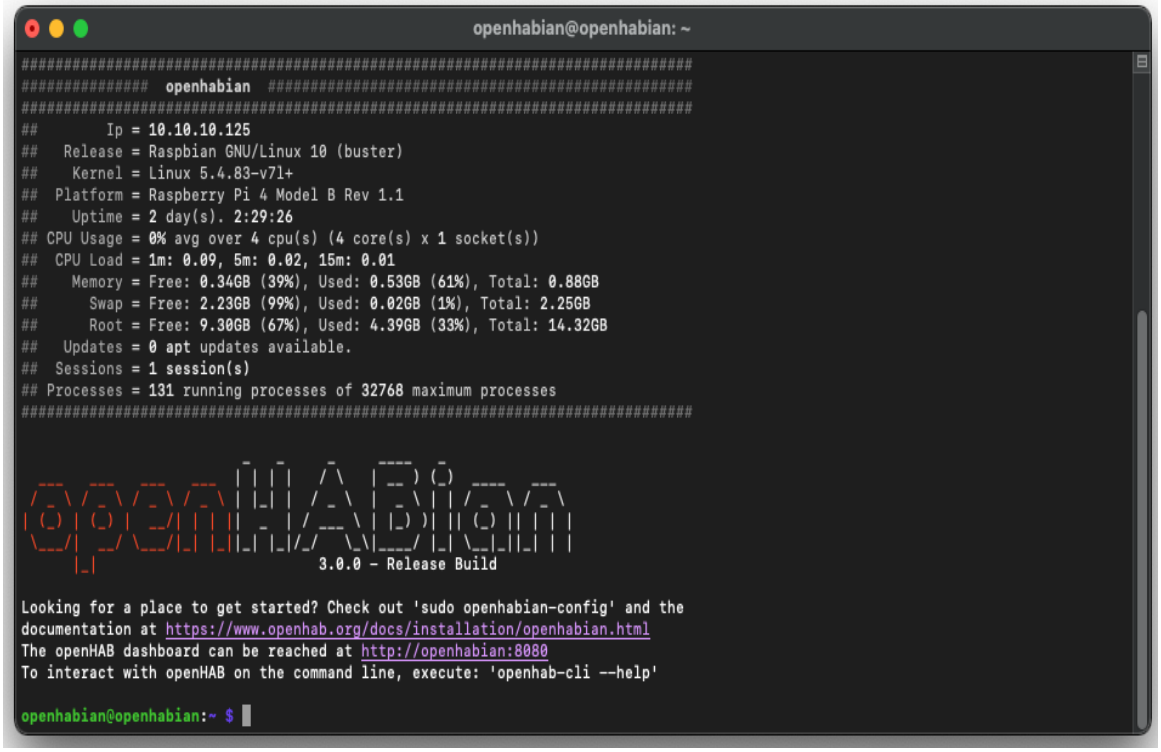
Moguće je prisupiti grafičkom sučelju preko internet preglednika, Androida i iOS-a. Idealan je za primjenu u ovom projektu zato što je besplatan, otvorenog koda, lagan za postavljanje i podržava dodavanje ručno napravljenih pametnih uređaja pomoću Arduino sklopovlja. Ova odabrana tehnologija je istovremeno lagana za naučiti, skalabilna je na veće sustave i istovremeno korisniku pomaže shvatiti princip na koji IoT sustavi rade i upoznavanje IoT protokola.

Moguće je dodati gotovo sve pametne uređaje, a i ručno izraditi podršku za one uređaje koji još nisu dodani. Kad jedan korisnik doda podršku za još nepostojeći uređaj u bazi, može distribuirati podršku za nj tako da to mogu svi ostali korisnici iskoristiti. openHAB nudi podršku za MQTT protokol koji će biti kasnije spomenut.

[14] https://www.openhab.org/ (preuzeto 13.09.2022.)

6.1 Instalacija openHAB programa na Raspberry-Pi

Za instalaciju openHAB-a na RaspberryPi potrebna je SD kartica koja se formatira i na koju se snimi openHABian slika koja je temeljena na RaspberryPi OS Lite operacijskom sustavu. Na slici 6.2. možemo vidjeti izgled openHAB početnog zaslona.



Slika 6.2. Izgled openHABian početnog zaslona[15]

Nakon snimanja slike na SD karticu, ubacuje se SD kartica u RaspberryPi i uključuje se. Tad se povezuje na openHAB korisničko sučelje. Za to je potrebna lokalna IP adresa RaspberryPi-a ili pak se može lokalno povezati unutar RaspberryPi-a za što se treba priključiti monitor, miš i tipkovnicu na RaspberryPi. Da bi se doznala lokalna IP adresu openHAB servera, odnosno RaspberryPi-a, postoje dva rješenja:

- priključiti monitor, miš i tipkovnicu na RaspberryPi i tada vidjeti konzolu sa svim potrebnim informacijama uključujući lokalnu IP adresu
- prijaviti se u ruter i tamo vidjeti koja je lokalna IP adresa dodijeljena RaspberryPi-u

[15] https://www.openhab.org/docs/installation/openhabian.html#raspberry-pi-prepackaged-sd-card-image (skinuto 17.09.2022.)

7. MQTT PROTOKOL

MQTT je standardni protokol za prijenos poruka za IoT uređaje. MQTT je skraćenica od Message Queuing Telemetry Transport. Radi na način izdavač/pretplatnik. Danas je korišten u različite svrhe pa čak i u industrijama kao što su automobilska, proizvođačka, telekomunikacijska i naftna. Za primanje samo željenih informacija koriste se teme. Informacije koje izdavači objavljuju su uvijek vezane uz neku temu. Pretplatitelj odlučuje na koje teme će se pretplatiti.

7.1. MQTT posrednici

MQTT posrednici su serveri koji su zaduženi za primanje poruka, pretplaćivanje i odjavljivanje tema s MQTT klijenata. Oni su važni zato što osiguravaju kvalitetu usluge. Osiguravaju prijenos poruke u slučajevima kad primatelj poruke nije na liniji u trenutku kad izdavač šalje poruku ili su smetnje na vezi. Radi tako da izdavač šalje ažuriranja na posrednik i tada posrednik prosljeđuje poruke pretplatnicima.

Postoje tri razine kvalitete usluge (engl. Quality of Services):

- najviše jednom - poslanu poruka koja je primljena je nemoguće poslati još jednom, ukoliko se primaoc sruši tijekom probavljanja poruke, poruka je izgubljena
- najmanje jednom - osigurava da će poruka stići minimalno jednom, no može stići više puta
- točno jednom - osigurava da je poruka zaprimljena točno jednom od strane primaoca, vrlo teško realizirati u praksi

Slika 7.1. prikazuje MQTT – izdavač pretplatnik model gdje se šalje informacija potvrde da je poruka izdana prema posredniku od kojeg dalje informacija putuje prema pretplatnicima

Slika 7.1. MQTT - Izdavač pretplatnik model

MQTT je nemoguće koristiti bez posrednika. Poruke na koje se nitko ne pretplati budu odbačene. Jednom kada posrednik pošalje poruku svim pretplatnicima, on zaboravlja poruku. Moguće je uključiti opciju da posrednik drži u memoriji zadnje poslanu poruku.

7.2 MQTT autentifikacija

MQTT nudi autentifikaciju na tri načina:

1. identifikacijska šifra klijenta
2. korisničko ime i zaporka
3. certifikati klijenata

7.2.1 Identifikacijska šifra klijenta

Svi MQTT klijenti moraju sadržavati identifikacijsku šifru. To je najniža razina autentifikacije. Moguće je odrediti koji prefiksi identifikacijskih šifri su dozvoljene. Posrednik tada provjerava identifikacijsku šifru kod spajanja i veže identifikacijsku šifru uz TCP vezu. Nakon što je pretplatitelj pretplaćen, posrednik zapamti pretplatitelja tako da veže identifikacijsku šifru i teme na koje se pretplatio.

7.2.2 Korisničko ime i zaporka

Posrednik traži korisničko ime i zaporku od klijenta prije nego dozvoli spajanje. Kombinacija korisničkog imena i zaporke je poslana u tekstualnom formatu bez enkripcije. To omogućava hakerima korištenje metode Man In The Middle da saznaju korisničko ime i zaporku. Na posredniku se traži datoteka s kombinacijama korisničkih imena i zaporki koja će služiti autentifikaciji. Moguće je podesiti autorizaciju tako da se teme ograniče samo na neka korisnička imena.

7.2.3 Certifikati klijenata

Ovo je najsigurnija metoda autentifikacije klijenta, ali i najteža zato što se mora upravljati certifikatima na više klijenata. Ova vrsta zaštite je jedino primjerena na manjoj količini klijenata i to na onim koji zahtjevaju najvišu razinu sigurnosti.

7.3 MQTT sigurnost

MQTT nudi dva načina enkripcije poruka, a to su TLS i tekstualna enkripcija.

7.3.1 TLS enkripcija

Poznata kao i SSL enkripcija koja je korištena na webu. Sve poruke će biti enkriptirane, no za implementaciju TLS enkripcije potreban je klijent koji to može omogućiti. Jednostavnijim klijentima je to problem.

7.3.2 Tekstualna enkripcija

Nudi enkripciju s jednog kraja do drugog kraja (end-to-end). Skriva informacije od posrednika i bilo koga drugoga tko nije pošiljatelj ili primaoc takve poruke. Ova metoda je dosta popularna zato što ne zahtjeva konfiguraciju posrednika.

Da bi se tekstualna enkripcija izvela, moguće je koristiti dijeljeni ključ (simetrična enkripcija) ili certifikate (asimetrična enkripcija). Korištenje dijeljenih ključeva je jednostavnije za implementirati, no postoji opasnost da tko god sazna dijeljeni ključ može čitati poruke enkriptirane tim dijeljenim ključem dok je enkripciju certifikatima nemoguće probiti.[16]

Moguće je koristiti TLS enkripciju zajedno s tekstualnom enkripcijom. Za osiguravanje integriteta podataka koristi se digitalno potpisivanje koje zahtjeva certifikate stoga jednostavniji klijenti poput običnih senzora neće moći implementirati tu funkcionalnost.[17]

7.4 MQTT protokol u odnosu na HTTP protokol

HTTP protokol radi na način zahtjev-odgovor, dok MQTT protokol radi na način izdavač-pretplatnik. MQTT protokol nudi oporuku i testament (engl. Last will & Testament) funkcionalnost koja u slučaju odspajanja izdavača i posrednika, posrednik šalje informaciju o odspajanju svim pretplatnicima na neku od tema odspojenog izdavača.

HTTP koristi zaglavlja puno veća od MQTT zaglavlja - najmanja moguća MQTT poruka sadrži 2 bajta. Puno je lakše čitati MQTT zaglavlja zato što je moguće samo par tipova poruka - CONNECT, PUBLISH, SUBSCRIBE, UNSUBSCRIBE i DISCONNECT, odnosno na hrvatskom SPOJI, OBJAVI, PRETPLATI, OTKAŽI PRETPLATU i ODSPOJI, dok HTTP zagljavlja sadrže 63 moguća tipa. Testovi na 3G mrežama pokazuju da je propusnost MQTT poruka 93 puta učinkovitija od HTTP poruka.[18]

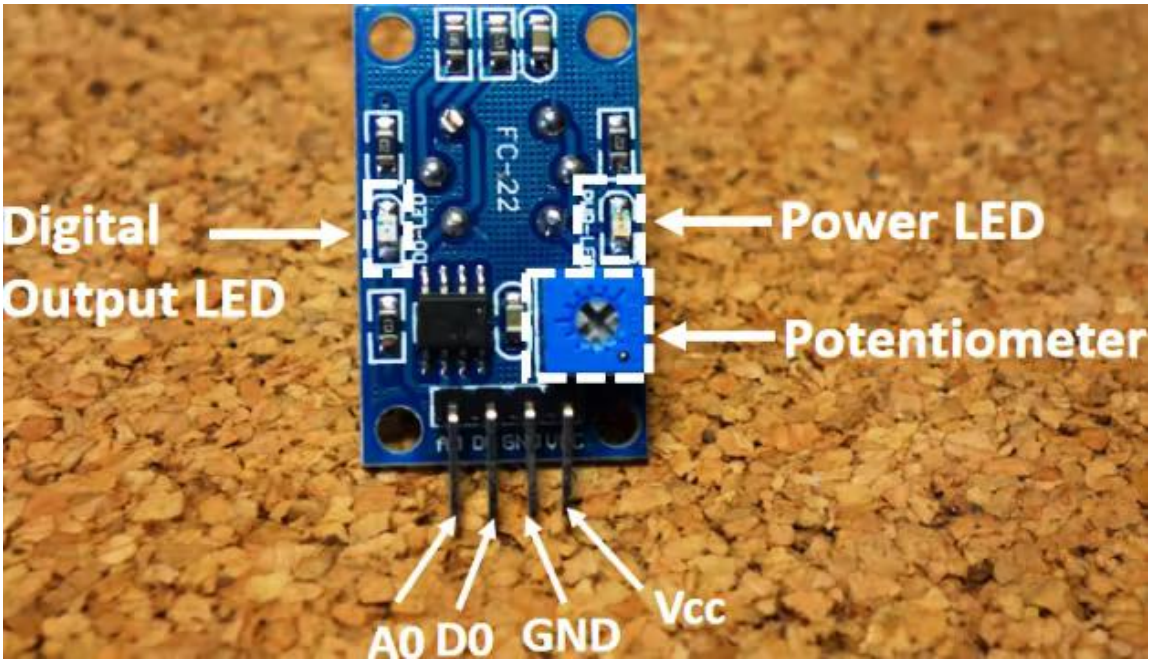
[16] https://docs.oracle.com/cd/E19047-01/sunscreen151/806-5397/howskipworks-4/index.html (preuzeto 13.09.2022.)
[17] http://www.steves-internet-guide.com/mqtt/ (preuzeto 13.09.2022.)
[18] https://medium.com/mqtt-buddy/mqtt-vs-http-which-one-is-the-best-for-iot-c868169b3105 (preuzeto 13.09.2022.)

8. IZRAĐIVANJE UREĐAJA ZA SENZOR POKRETA, VLAŽNOSTI I TEMPERATURE ZRAKA I DIMA

Wemos D1 mini je WiFi Arduino kompatibilna pločica bazirana na ESP-8266EX SoC-u. Ima flash memoriju od 4 megabajta, 11 digitalnih ulazno izlaznih jedinica i 1 analognu ulazno izlaznu jedinicu.

8.1 Senzor dima

Koristimo senzor MQ-2 senzor dima koji uz dim može i detektirati ostale zapaljive plinove kao što su: LPG, Butan, Propan, Metan, Alkohol, Vodik. Ima ugrađen potenciometar koji služi da podešavanje osjetljivosti na plinove. Različiti plinovi imaju različite pragove detekcije. Prisutnost zapaljivih tvari u zraku izaziva visok napon na digitalnom izlazu senzora, dok niska prisutnost ili odsutnost izaziva niski napon. Na slici 8.1. prikazan je MQ-2 senzor za dim.



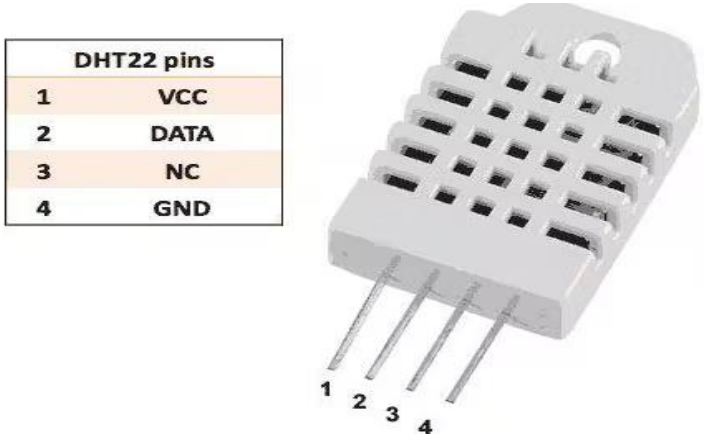
Slika 8.1. Prikaz izlaza MQ-2 senzora[19]

8.2 Senzor vlažnosti i temperature zraka

Korišten je senzor DHT22 koji omogućava detekciju vlažnosti i temperature zraka i digitalni izlaz.

[19] https://create.arduino.cc/projecthub/Aritro/smoke-detection-using-mq-2-gas-sensor-79c54a (preuzeto 13.09.2022.)

Na slici 8.2. prikazani su izlazi DHT22 senzora vlažnosti i temperature zraka s digitalnim izlazom.



Slika 8.2. Izlazi iz DHT22 senzora[20]

Tehnički podaci DHT22 senzora:

1. 5V napajanje
2. digitalni izlaz podataka
3. izlaz bez funkcije
4. uzemljenje

8.3 Senzor pokreta

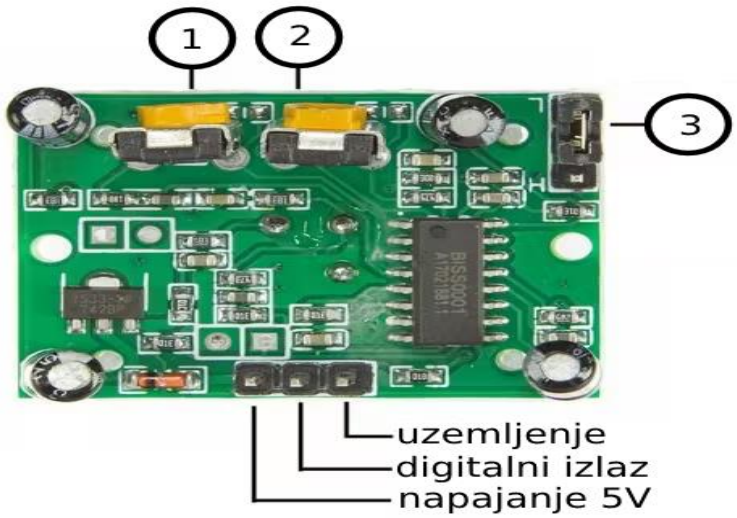
Za senzor pokreta korišten je ElectroPeak HC-SR501 PIR senzor.

Ima sljedeće mogućnosti koje je moguće podesiti na mjestima gdje su odgovarajuće brojke na slici 8.3.:

1. određivanja trajanja okidača (koliko dugo traje visok napon na izlazu nakon okidanja senzora)
2. određivanja osjetljivosti na pokret
3. određivanja hoće li se kod uzastupnih pokreta okidač držati uključen ili će se okidač više puta paliti i isključivati

[20] https://create.arduino.cc/projecthub/electropeak/pir-motion-sensor-how-to-use-pirs-w-arduino-raspberry-pi-18d7fa *(preuzeto 13.09.2022.)*

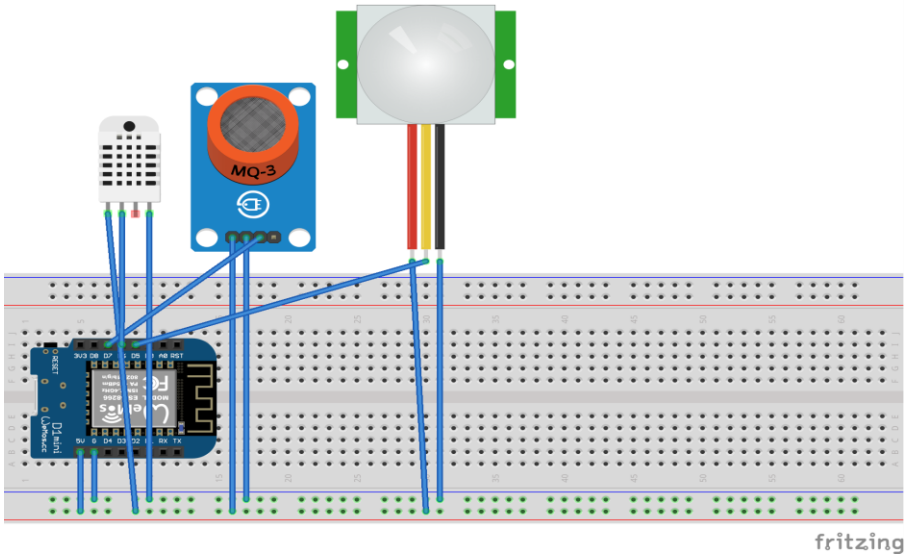
Slika 8.3. prikazuje izlaze iz PiR senzora pokreta.



Slika 8.3. Izlazi iz PIR senzora[21]

8.4 Shema spajanja svih komponenti na D1 Wemos mini

Slika 8.4. prikazuje shemu spajanja senzora dima, temperature i vlažnosti i senzora pokreta spojenih na Arduino pločicu



Slika 8.4. Shema spajanja svih komponenti[22]

[21] https://create.arduino.cc/projecthub/electropeak/pir-motion-sensor-how-to-use-pirs-w-arduino-raspberry-pi-18d7fa *(preuzeto 13.09.2022.)*
[22] https://create.arduino.cc/projecthub/electropeak/pir-motion-sensor-how-to-use-pirs-w-arduino-raspberry-pi-18d7fa *(preuzeto 13.09.2022.)*

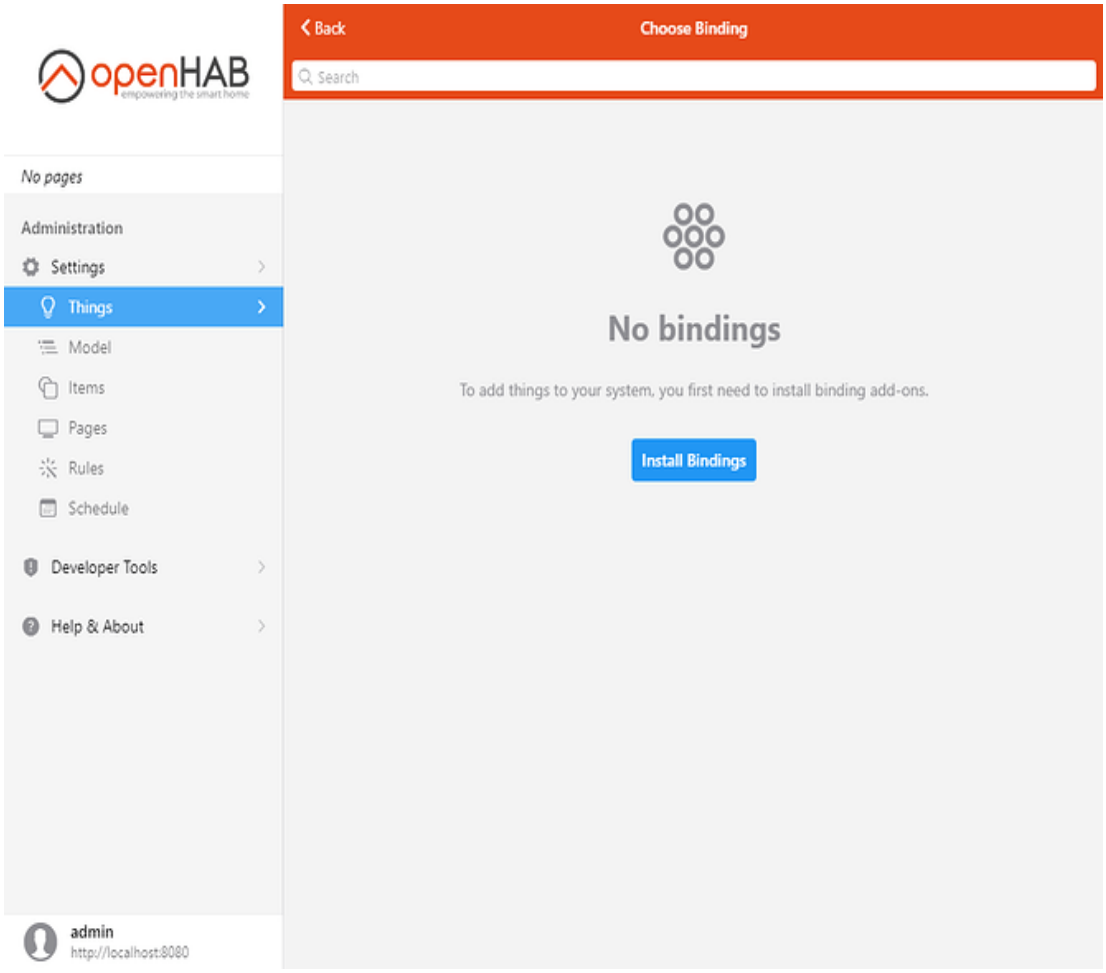
8.5 Spajanje pametnih uređaja na mrežu

openHAB instrukcije nam pokazuju kako spojiti gotove pametne uređaje na openHAB mrežu i kako spojiti ručno dodani Arduino uređaj.

Također, kako podesiti vremenske i događajne okidače, kako kontrolirati udaljeno preko interneta, itd.

8.6 Upute za dodavanje pametne žarulje Hue marke Philips

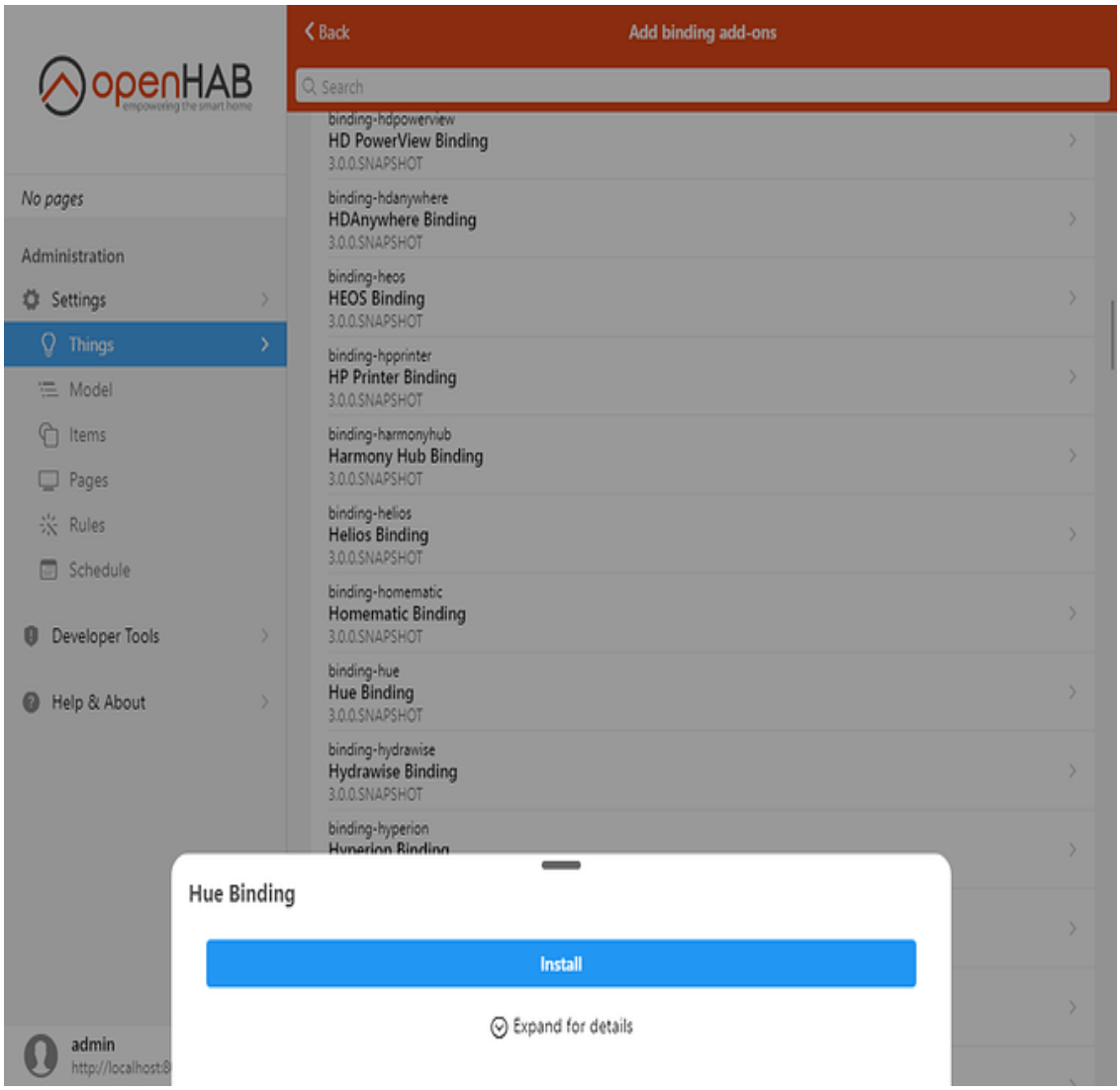
U openHAB sučelju kliknuti na Things (Stvari) kao što je prikazano na slici 8.5.



Slika 8.5. openHAB sučelje[23]

[23] https://www.openhab.org/docs/tutorial/things_simple.html *(preuzeto 13.09.2022.)*

1. Kliknuti na Install Bindings (instaliraj spajanja) i zatim iz popisa svih spajanja izabrati Hue Binding (Hue spajanje) kao na slici 8.7.



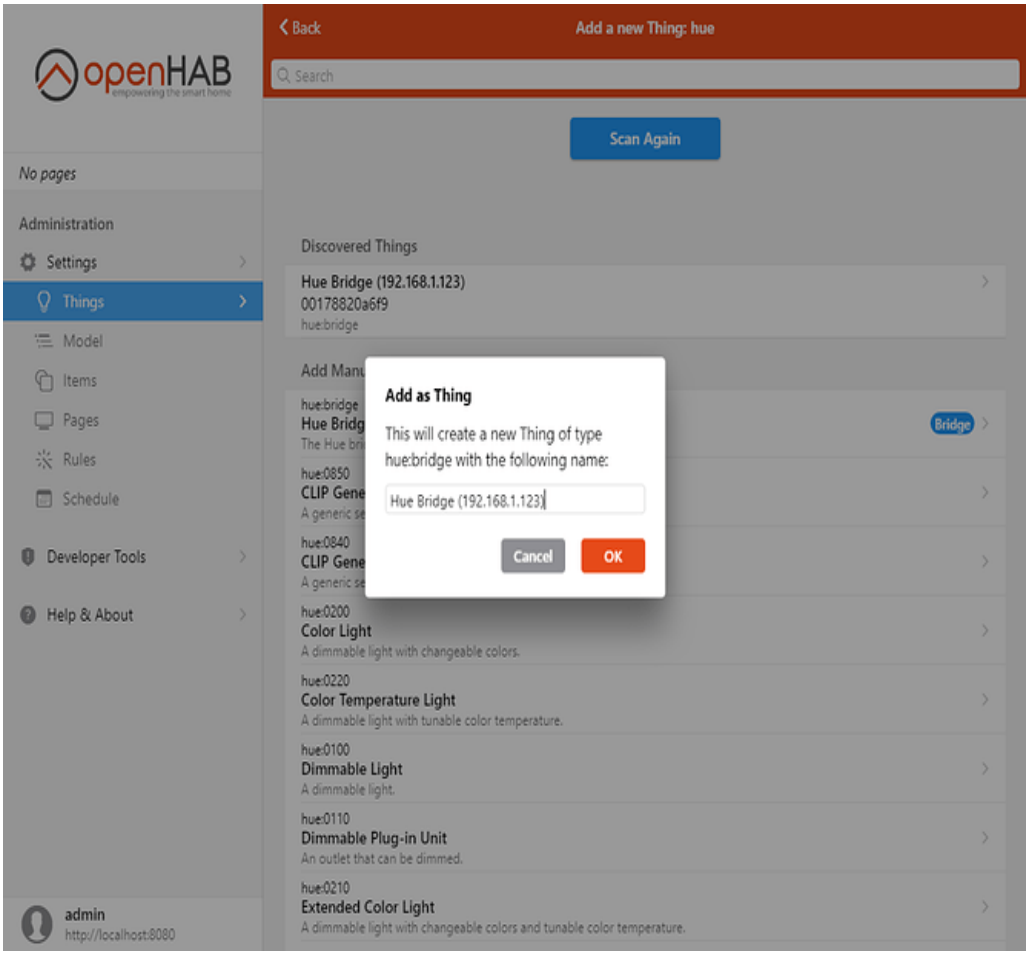
Slika 8.5. openHAB sučelje[24]

2. Nakon što se spajanje instalira, bit će prikazani svi uređaji na mreži koji su podržani od Hue spajanja, a to su sve Philips Hue žarulje koje su povezane na WiFi mrežu.

   Tada treba kliknuti na žarulju koju se želi dodati, dati joj ime po želji i kliknuti tipku OK.

[24] https://www.openhab.org/docs/tutorial/things_simple.html (preuzeto 13.09.2022.)

Prikazano na slici 8.7.



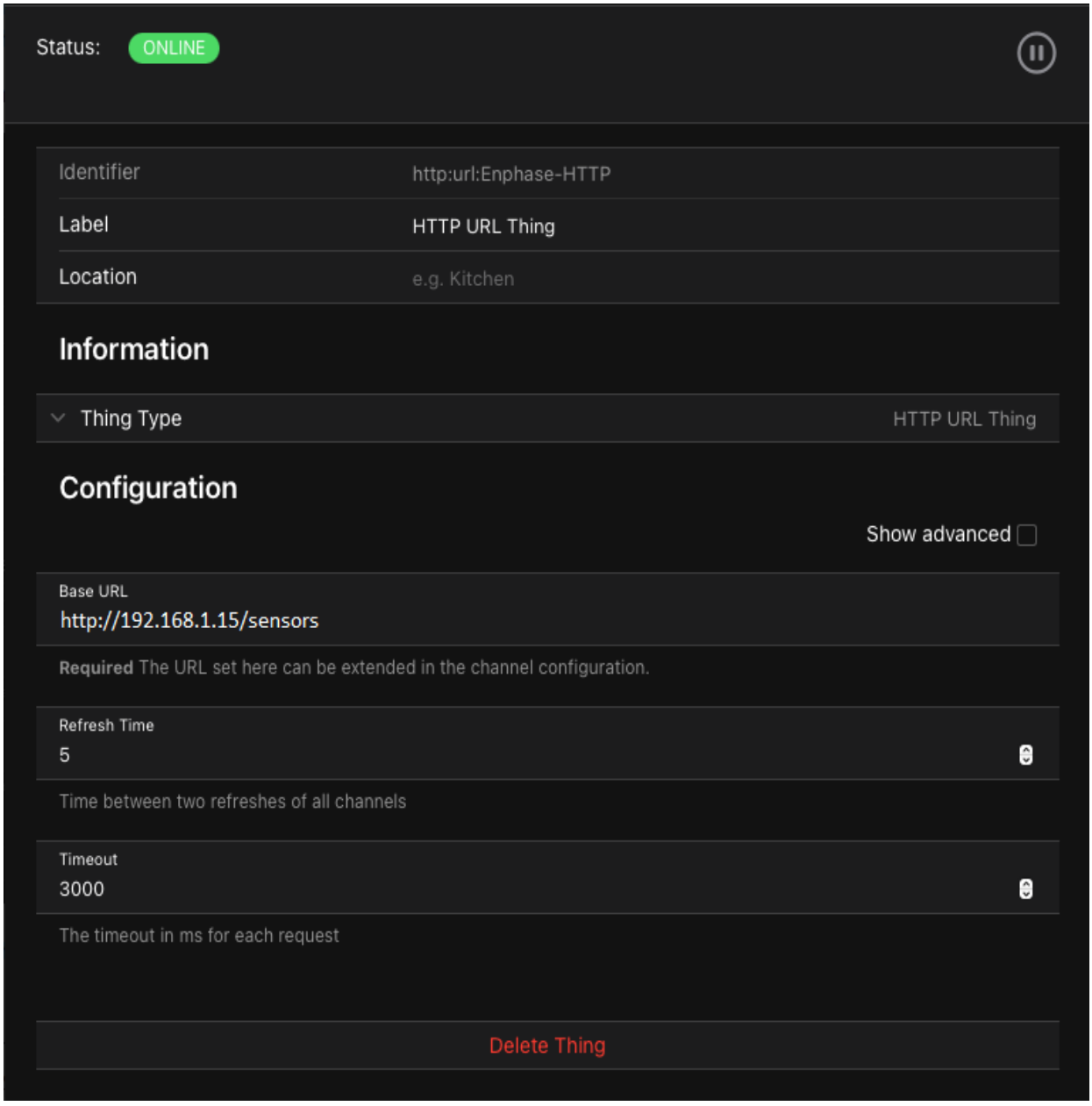
Slika 8.7. openHAB sučelje[25]

3. Nakon toga je potrebno pritisnuti fizičku tipku na Hue žarulji da bi se dozvolilo povezivanje. Tada nakon par sekundi uređaj bi trebao biti spojen i moguće ga je kontrolirati pomoću openHAB-a. Cijeli postupak dodavanja Hue žarulje, kao i ostalih Stvari moguće je potražiti na službenoj openHAB stranici.

[25] https://www.openhab.org/docs/tutorial/things_simple.html (preuzeto 13.09.2022.)

8.7. Dodavanje uređaja za senzor pokreta, vlažnosti i temperature zraka i dima na openHAB

Da bi openHAB mogao pročitati JSON odgovor koji mu uređaj pošalje, na openHAB je potrebno instalirati dodatak JSONPath Transformation. Tada je potrebno kreirati HTTP Stvar. Kod HTTP Stvari pod Base URL potrebno je upisati lokalnu IP adresu Arduino uređaja. Tu IP adresu možemo saznati pomoću rutera. Na slici 8.8. prikazan je popunjen formular za dodavanje HTTP Stvari.



Slika 8.8. Popunjen formular dodavanja HTTP stvari

Arduino nam vraća podatke u obliku JSON formata. JSON format je skraćenica za JavaScript Object Notation i on se koristi za strukturiranje podataka.

JSON format koji nam Arduino vraća izgleda ovako:

```
{
        “mq2”: 2,
        “pir”: false,
        “humidity”: 95,
        “temperature”: 23
}
```

Zatim je potrebno dodati Kanale na toj Stvari koji će iskoristiti JSON podatke. Za to koristimo JSONPath Transformation dodatak da bi se podaci iz JSON formata transformirali u brojčanu vrijednost koju će kanal uzimati. Za vlažnost zraka Transformacija Stanja (State Transformation) kanala treba izgledati kako je prikazano slikom 8.9.

Slika 8.9.. Transformacija Stanja za vlažnost zraka

Kada se to ponovi za svako od polja - mq2, pir i temperature, spojit će se Arduino uređaj koji šalje podatke preko HTTP protokola u JSON formatu na openHAB koji čita te podatke i ima mogućnost prikaza tih podataka korisniku.

8.8 Uključivanje i isključivanje uređaja u određeno vrijeme

Pravila u openHAB-u su skripte koje reagiraju na neke podražaje, vanjske ili unutarnje, i obavljaju određene akcije s obzirom na to koja pravila koja postavimo.

Ako nazovemo naš uređaj imenom Svjetlo, tada možemo dodati sljedeće pravilo koje će upaliti uređaj Svjetlo ujutro u 9 sati i isključiti ga za 15 minuta, odnosno u 9:15 ujutro.

```
import org.openhab.core.library.types.*
import org.openhab.core.persistence.*
import org.openhab.model.script.actions.*
import org.java.math.*
import org.joda.time.*

rule "Uključi Svjetlo"
when
Time cron "0 0 9 ? * *"
then
Svjetlo.sendCommand(ON)
end

rule "Isključi Svjetlo"
when
Time cron "0 15 9 ? * *"
then
Svjetlo.sendCommand(OFF)
end
```

Slika 8.10. Isječak pravila za paljenje i gašenje svjetla s obzirom na vrijeme

8.9 Uključivanje Svjetla kad senzor pokreta detektira pokret

Da bi se uključilo svjetlo na podražaj senzora pokreta, koristi se okidači bazirani na Stvarima. Ako se senzor pokreta nazove SenzorPokreta, skripta pravila izgleda sljedeće:

rule “Uključi Svjetlo na senzor pokreta”

when

Thing SenzorPokreta changed [from false] [to true]

then

Svjetlo.sendCommand(ON)

end

8.10 Dodavanje uređaja za senzor pokreta, vlažnosti i temperature zraka i dima na openHAB

Da bi openHAB mogao pročitati JSON odgovor koji mu uređaj pošalje, na openHAB je potrebno instalirati dodatak JSONPath Transformation. Zatim je potrebno dodati MQTT Posrednik (MQTT Broker).

8.11 Ranjivosti korištenja glasovnih naredbi za upravljanje

Svakodnevno se pojavljuju nove ranjivosti, a s tim i novi rizici korištenja tehnologija. Jedna od ranjivosti upravljanja glasovnim naredbama jest da se može glasovno upravljati uređajem kroz staklo pomoću lasera. Kad laser svijetli direktno na mikrofon, dijagragma mikrofona se povećava, a kad svjetlo prestane, dijafragma se suzi. Na taj način mogu se simulirati akustični valovi u sobi u kojoj se nalazi prepoznavatelj govora. Udaljenost napada ovisi o preciznosti i jačini lasera, a zabilježeno je da je napad bio uspješan na udaljenosti od čak 100 metara.[26] [27]

Druga ranjivost, koja ne isključuje nužno prvu ranjivost, jest davanje glasovnih naredbi ljudski nečujnim zvučnim valovima odnosno psihoakustično skrivanje (engl. psychoacoustic hiding). Takvi zvučni valovi su dizajnirani da ih ljudsko uho ne čuje, a na pametnim uređajima okidaju prepoznavanje govora. Budući da su prepoznavatelji govora kompleksni algoritmi koji čak mogu djelomično uključivati strojno učenje, teško ih je razumjeti i zakrpati ranjivosti. Oni daju najbolje od sebe da dekodiraju moguće zvučne valove pa stoga nije čudno da su se pojavile ovakvi napadi. Video koji govori na tu temu je na poveznici.[28]

Uz dvije navedene ranjivosti rizik upravljanja sustavom zaključavanja ili sustavom alarma glasovnim naredbama se povećava, no ukoliko se uređaj prepoznavanja govora drži na mjestu nedostupnom direktnom svjetlu lasera izvana i ukoliko su svi zvučni uređaji u kući (uključujući mobitele) isključeni, praktički nema rizika.

8.12 Odvajanje pametnih uređaja od interneta

Za najveću razinu privatnosti, budući da su komercijalni pametni uređaji zatvorenog koda i pod rizikom nadziranja pomoću službenih ažuriranja i pod rizikom pristupa i upravljanja od strane hakera, poželjno je odvojiti pametne uređaje u lokalnu mrežu tako da se smanji rizik neželjenog pristupa.

---

[26] https://www.youtube.com/watch?v=OQHJhUVJGeo (preuzeto 13.09.2022.)

[27] https://www.forbes.com/sites/thomasbrewster/2019/11/05/amazon-alexa-google-home-hacked-with-a-laser/(preuzeto 13.09.2022.)

[28] https://www.youtube.com/watch?v=lvxwFpk27-Y(preuzeto 13.09.2022.)

Za to je potreban ili ruter s mogućnošću virtualnih lokalnih mreža (engl. VLAN) ili dodatan ruter koji nema vezu s internetom. Tad će komercijalne mobilne aplikacije za upravljanje pametnim uređajima postati beskorisne, no i dalje možemo upravljati uređajima lokalno, pa čak i korištenjem glasovnih naredbi Amazon Alexe lokalno. Međutim, samo su novije generacije Amazon Alexe u mogućnosti koristiti glasovne naredbe lokalno, bez pristupa internetu.

8.13 Korištenje Amazon Alexe lokalno

Korištenje Amazon Alexe lokalno moguće je samo uz novije Amazon Echo uređaje. Uređaji koji to podržavaju su Echo Plus (prva i druga generacija) i Echo Show (druga generacija). Preporučeno je korištenje Amazon Alexe, kao i svih pametnih IoT uređaja, lokalno zbog zaštite privatnosti. Tada će biti moguće koristiti Alexu samo za akcije pametnih uređaja, postavljanje jačine zvuka, provjeravanje vremena ili otkazati postojeće alarme i podsjetnike.

Obavezno je napraviti korisnički račun za Amazon Alexu. Zatim je potrebno upariti sve pametne uređaje s Alexom. Nakon toga se u Alexa mobilnoj aplikaciji uključi funkcija Local Voice Control. Kad je to uključeno, Alexa će raditi čak i kad nije spojena na internet, no ako je ručno ne odspojimo preko rutera, i dalje će ostati spojena. [29]

Slika 8.11. prikazuje izgled AMAZON Echo Dot 3rd Generation bluetooth prijenosnog zvučnika.

Slika 8.11. Bluetooth zvučnik AMAZON Echo Dot 3rd Generation[30]

[29] https://www.howtogeek.com/732835/how-to-use-alexa-offline-with-smart-home-devices-using-echo/

[30] https://vacom.hr/proizvod/bluetooth-zvucnik-amazon-echo-dot-3rd-generation-sivi/ (preuzeto 13.09.2022.)

Druga solucija je dopuštenje komercijalnim pametnim uređajima vezu s internetom, no svejedno odvojiti lokalne uređaje poput računala, mobitela od pametnih uređaja. Ranjivost će biti nešto veća nego kod prve varijante, no svejedno značajno manja nego bez ovih mjera opreza.

Jedan od ruter firmware-a koji dopuštaju odvajanje uređaja u zasebne virtualne lokalne mreže je dd-wrt. Baziran je na linuxu, otvorenog je koda i podržava velik broj rutera.

8.14 Postavljanje zasebne virtualne lokalne mreže na dd-wrt ruteru

Da bi se instalirao dd-wrt firmware na ruter, potrebno je slijediti uputstva prikazana na službenoj stranici DD-WRT-a čije su upute za svaki ruter različite.

Postoje tri načina za to:

1) WebGUI
2) TFTP
3) Command Line

Detaljne upute se mogu naći na poveznici: https://forum.dd-wrt.com/wiki/index.php/Installation

Nakon postavljanja dd-wrt firmware-a na ruter, potrebno je slijediti upute prikazane na wikipediji na sljedećoj poveznici:

https://wiki.ddwrt.com/wiki/index.php/VLAN_Detached_Networks_(Separate_Networks_With_Internet)

Slika 8.12.. Linksys WRT3200ACM usmjerivač[31]

[31] https://www.amazon.com/gp/product/B01JOXW3YE/ref=as_li_tl?ie=UTF8&tag=techsphinx02-20&camp=1789&creative=9325&linkCode=as2&creativeASIN=B01JOXW3YE&linkId=749fa936853b9526adf07108bc3699e9 (preuzeto 13.09.2022.)

9. ZAKLJUČAK

Protokoli i tehnologije vezane uz IoT napreduju i olakšavaju izvedbu pametnih sustava i povezivanje uređaja. Sigurnost i privatnost su vrlo bitne stvari stoga je poželjno da razvoj novih tehnologija za IoT bude otvorenog koda, da krajnji korisnici imaju mogućnost mijenjanja izvornog koda i dozvolu za redistribuiranje izmijenjenih programa. Za veću sigurnost potrebno je uložiti više truda i vremena u postavljanje IoT uređaja, sve ovisi o potrebama projekta. openHAB kao program s velikom potporom zajednice i dovoljno mogućnosti da zadovolji potrebe najzahtjevnijih privatnih korisnika je idealan za izradu domaćih projekata pametne kuće. Pojam pametne kuće u današnje vrijeme je sve više populariziran. Ova vrsta automatiziranog sustava korisniku omogućava potpunu kontrolu nad svakim sustavom u pametnoj kući putem pametnog telefona. Implementacijom ove vrste sustava dobivamo poseban automatizacijski mehanizam pametne kuće koji značajno dovodi do uštede i djelotvornosti, ali i potpunog užitka korisnika u njegovom domu.

10. POPIS IZVORA

[1] Bukarica, Vesna, Priručnik za energetske savjetnike: poticanje energetske efikasnosti u Hrvatskoj, Zagreb, program Ujedinjenih naroda za razvoj (UNDP), 2008. (preuzeto 13.09.2022.)

[2] Radovan, M, računalne mreže (1), Rijeka, Digital point tiskara, 2010. (preuzeto 13.09.2022.)

[3] Arun, (2013). What ih Wi-Fi, Scribd, https://www.scribd.com/document/192846073/Wifi (preuzeto 13.09.2022.)

[4] Govlnsider (2019.) The future od WiFi is here. It will transform cities forever, Govlnsider https://govinsider.asia/connected-gov/the-future-of-wifi-is-here-it-will-trqnsform-cities-forever/ (preuzeto 13.09.2022.)

[5] Mitchel, B., (2019.). The range od typical Wi-Fi Network, Lifewire. https://www.lifewire.com/range-of-typical-wifi-neteork-816564 (preuzeto 13.09.2022.)

[6] openHAB Community and the openHAB Foundation e.V., (2022.). openHABian - Hassle-free openHAB setup. Dostupno na https://www.openhab.org/docs/installation/openhabian.html#raspberry-pi-prepackaged-sd-card-image (preuzeto 14.9.2022.)

[7] Oracle, Shared Key and Public Key Encryption, 2010. Dostupno na https://docs.oracle.com/cd/E19047-01/sunscreen151/806-5397/howskipworks-4/index.html (preuzeto 13.9.2022.)

[8] Steve Cope, Beginners Guide To The MQTT Protocol, 2018. Dostupno na http://www.steves-internet-guide.com/mqtt/ (preuzeto 14.9.2022.)

[9] Marina Serozhenko, MQTT vs. HTTP: which one is the best for IoT?, 2017. Dostupno na https://medium.com/mqtt-buddy/mqtt-vs-http-which-one-is-the-best-for-iot-c868169b3105 (preuzeto 14.9.2022.)

[10] The Action Lab, Hacking into an Amazon Echo Using Laser Light to "Speak" to Alexa, 2019. Dostupno na https://www.youtube.com/watch?v=OQHJhUVJGeo (preuzeto 14.09.2022.)

[11] Thomas Brewster, Amazon Alexa Can Be Hacked By A Laser From 100 Meters—Is It Time To Hide Your Echo?, 2019. Dostupno na https://www.forbes.com/sites/thomasbrewster/2019/11/05/amazon-alexa-google-home-hacked-with-a-laser/?sh=4924a09527d2 (preuzeto 14.09.2022.)

[12] The Hated One, The Science of Inaudible Voice Hacking, 2022. Dostupno na https://www.youtube.com/watch?v=lvxwFpk27-Y (preuzeto 14.09.2022.)

[13] Samir Makwana, How to Use Alexa Offline with Smart Home Devices Using Echo, 2021. Dostupno na https://www.howtogeek.com/732835/how-to-use-alexa-offline-with-smart-home-devices-using-echo/ (preuzeto 14.09.2022.)

[14] Knok, Željko, dostupno na: https://hrcak.srce.hr/66261 (preuzeto 13.09.2022)

[15] https://www.bluetooth.com/ (preuzeto 15.09.2022.)

[16] https://www.openhab.org/docs/installation/openhabian.html#raspberry-pi-prepackaged-sd-card-image (preuzeto 15.09.2022.)

[17] http://www.steves-internet-guide.com/mqtt/ (preuzeto 15.09.2022.)

[18] https://create.arduino.cc/projecthub/Aritro/smoke-detection-using-mq-2-gas-sensor-79c54a (preuzeto 15.09.2022.)

[19] https://create.arduino.cc/projecthub/sachendra003/arduino-humidity-sensor-288146 (preuzeto 15.09.2022.)

[20] https://repozitorij.unipu.hr/islandora/object/unipu%3A3907/datastream/PDF/view (preuzeto 14.09.2022.)

[21] https://forum.dd-wrt.com/wiki/index.php/Installation (preuzeto 15.09.2022.)

[22] https://medium.com/mqtt-buddy/mqtt-vs-http-which-one-is-the-best-for-iot-c868169b3105 (preuzeto 15.09.2022.)

[23] https://create.arduino.cc/projecthub/electropeak/pir-motion-sensor-how-to-use-pirs-w-arduino-raspberry-pi-18d7fa (preuzeto 15.09.2022.)

[24] https://www.openhab.org/docs/tutorial/things_simple.html (preuzeto 15.09.2022.)

[25] https://www.forbes.com/sites/thomasbrewster/2019/11/05/amazon-alexa-google-home-hacked-with-a-laser/ (preuzeto 15.09.2022.)

[26] https://www.youtube.com/watch?v=lvxwFpk27-Y (preuzeto 15.09.2022.)

[27] https://www.howtogeek.com/732835/how-to-use-alexa-offline-with-smart-home-devices-using-echo/ (preuzeto 15.09.2022.)

[28] https://forum.dd-wrt.com/wiki/index.php/Installation (preuzeto 15.09.2022.)

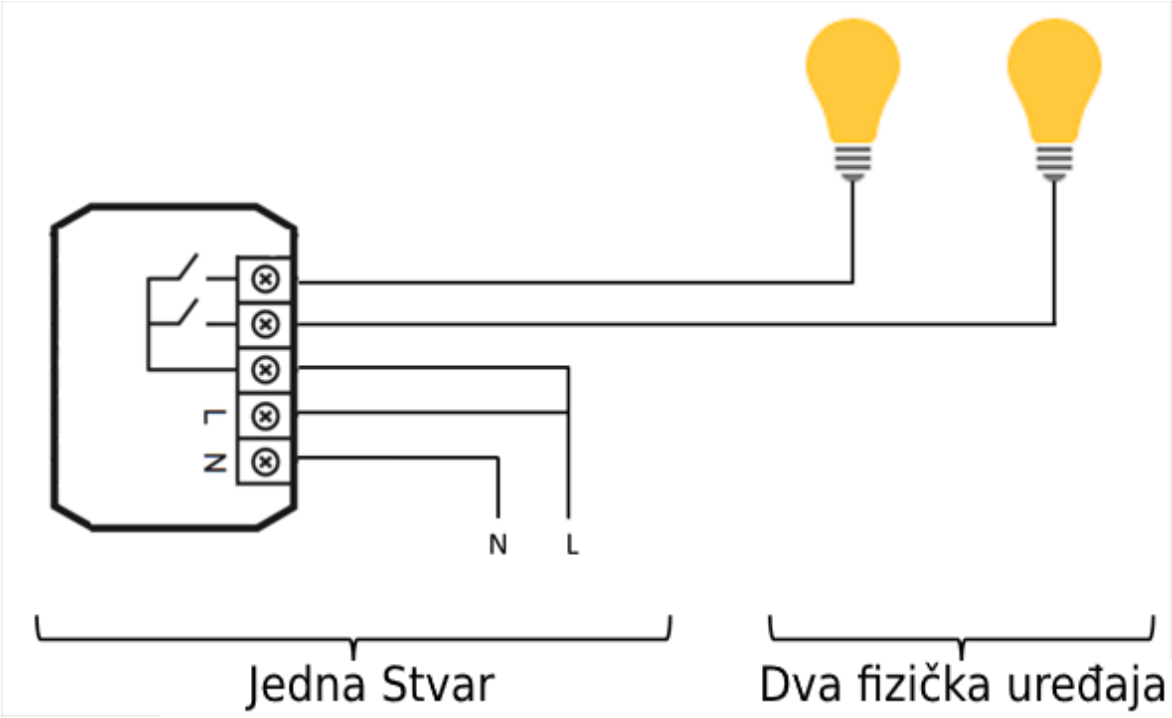
[29] https://wiki.dd-wrt.com/wiki/index.php/VLAN_Detached_Networks_(Separate_Networks_With_Internet) (preuzeto 15.09.2022.)

[30] https://www.istockphoto.com/search/search-by-asset?affiliateredirect=true&assetid=495732397&assettype=image&cid=IS&clickid=TsnxvHzlFxyNRpcxCQUTgxq3UkDWqAw9CWpf0Q0&irgwc=1&irpid=1404368&utm_campaign=&utm_content=258824&utm_medium=affiliate&utm_source=Du%C5%A1an+Bi%C4%8Danski+Pr+Digitalnio&utm_term= (preuzeto 13.9.2022.)

PRILOZI

OPENHAB RJEČNIK

- Stvari (engl. Things): Stvari su entiteti koji se fizički mogu dodati sustavu i pružaju mnoge funkcionalnosti. Stvari ne moraju biti uređaji, već mogu predstavljati i web servis ili bilo koji drugi upravljivi izvor informacija i funkcionalnosti. Stvari mogu odgovarati jedan na jedan fizičkim uređajima, no ne moraju. Moguće je da jedna Stvar sadrži više fizičkih uređaja kao što je prikazano na slici 8.13.



Slika 8.13. Primjer korištenja jedne Stvari s dva fizička uređaja [16]

- Kanali (engl. Channels): Stvari pružaju "Kanale" koji predstavljaju različite funkcije koje Stvar pruža. Ako je Stvar fizički entitet ili izvor informacija, Kanal je konkretna funkcija ove Stvari. Svaki od kanala ima svoja svojstva i mogućnosti. Na primjer, fizička žarulja može imati kanal temperature boje i kanal boje. Za izvore informacija Stvar može biti senzor s različitim kanalima poput temperature, tlaka i vlažnosti.
- Mostovi (engl. Bridges): Posebna vrsta Stvari su Mostovi. Mostovi su Stvari koje je potrebno dodati u sustav kako bi se dobio pristup drugim Stvarima. Tipičan primjer mosta je IP pristupnik za sustav kućne automatizacije koji se ne temelji na IP-u.
- Otkrivanje (engl. Discovery): Budući da se mnoge stvari mogu automatski otkriti, dostupni su različiti mehanizmi koji se bave dodavanjem automatski otkrivajućih Stvari.

Programski kod u Arduino (C++) jeziku

```
#include <Arduino.h>

#include <ESP8266WiFi.h>

#include <ESP8266HTTPClient.h>

#include <WiFiClient.h>

#include <ESP8266WebServer.h>


#include <DHT.h>;


#ifndef STASSID

#define STASSID "ssid"

#define STAPSK  "password"

#define MOTION_DETECTED 0

#define SMOKE_THRESHOLD_REACHED 1

#endif


const char* ssid = STASSID;

const char* password = STAPSK;


ESP8266WebServer server(80);


const int senzor_mq2_pin = 13; // d7

const int senzor_dht22_pin = 12; // d6

const int senzor_pir_pin = 14; // d5
```

```
int smoke_threshold = 1024;

int smoke_value = 0;

int temp_value = 0;

int humid_value = 0;

bool pir_value = 0;

String server_ip = "http://192.168.1.2/";


DHT dht(senzor_dht22_pin, DHT22);


void handleNotFound() {

  String message = "File Not Found\n\n";

  message += "URI: ";

  message += server.uri();

  message += "\nMethod: ";

  message += (server.method() == HTTP_GET) ? "GET" : "POST";

  message += "\nArguments: ";

  message += server.args();

  message += "\n";

  for (uint8_t i = 0; i < server.args(); i++) {

    message += " " + server.argName(i) + ": " + server.arg(i) + "\n";

  }

  server.send(404, "text/plain", message);
```

```
}

void setup(void) {
  Serial.begin(115200);
  WiFi.mode(WIFI_STA);
  WiFi.begin(ssid, password);
  Serial.println("");

  dht.begin();

  pinMode(senzor_mq2_pin, INPUT);
  pinMode(senzor_dht22_pin, INPUT);
  pinMode(senzor_pir_pin, INPUT);

  // Wait for connection
  while (WiFi.status() != WL_CONNECTED) {
    delay(500);
    Serial.print(".");
  }
  Serial.println("");
  Serial.print("Connected to ");
  Serial.println(ssid);
  Serial.print("IP address: ");
```

```
  Serial.println(WiFi.localIP());


  server.on("/sensors", []() {
    smoke_value = analogRead(senzor_mq2_pin);
    pir_value = digitalRead(senzor_pir_pin);
    humid_value = dht.readHumidity();
    temp_value = dht.readTemperature();

    String response = "{'mq2': '" + String(smoke_value) + "', 'pir': '" + String(pir_value) + "', 'humidity':
'" + String(humid_value) + "', 'temperature': '" + String(temp_value) + "'}";
    server.send(200, "text/plain", response);
  });


  server.on("/setSmokeThreshold", []() {
    if(server.args() > 0){
      if(server.argName(0) == "smokeThreshold"){
        smoke_threshold = server.arg(0).toInt();
      } else server.send(400, "text/plain", "Arguments: integer (0-1024) as 'smokeThreshold'");
    } else server.send(400, "text/plain", "Arguments: integer (0-1024) as 'smokeThreshold'");

    server.send(200, "text/plain", "Threshold set to " + String(smoke_threshold));
  });


  server.onNotFound(handleNotFound);
```

```
  server.begin();
  Serial.println("HTTP server started");
}

void loop(void) {
  server.handleClient();

  if(digitalRead(senzor_pir_pin) == HIGH && pir_value == false){
    sendNotification(MOTION_DETECTED);
    Serial.println("Motion detected!");
  }

  if(analogRead(senzor_mq2_pin) > smoke_threshold && smoke_value <= smoke_threshold){
    sendNotification(SMOKE_THRESHOLD_REACHED);
    Serial.println("Smoke threshold reached!");
  }

  smoke_value = analogRead(senzor_mq2_pin);
  pir_value = digitalRead(senzor_pir_pin);

  delay(500);
}
```

```
void sendNotification(int notification_type){
  String saljem = "";
  switch(notification_type){
    case MOTION_DETECTED:
      saljem = server_ip + "motionDetected";
      break;
    case SMOKE_THRESHOLD_REACHED:
      saljem = server_ip + "smokeThresholdReached";
      break;
  }
  Serial.print("Šaljem ");
  Serial.println(saljem);

  int pokusaj = 0;
  bool poslano = false;
  while(!poslano && pokusaj < 10){
    // wait for WiFi connection
    if ((WiFi.status() == WL_CONNECTED)) {
      WiFiClient client;
      HTTPClient http;
      Serial.print("[HTTP] begin...\n");
      if (http.begin(client, saljem)) {
```

```
    Serial.print("[HTTP] GET...\n");
    // start connection and send HTTP header
    int httpCode = http.GET();
    // httpCode will be negative on error
    if (httpCode > 0) {
      // HTTP header has been send and Server response header has been handled
      Serial.printf("[HTTP] GET... code: %d\n", httpCode);
      if(httpCode == HTTP_CODE_NOT_FOUND){
        poslano = true;
        Serial.println("not found");
      }
      if (httpCode == HTTP_CODE_OK || httpCode == HTTP_CODE_MOVED_PERMANENTLY) {
        poslano = true;
        String payload = http.getString();
        Serial.println(payload);
      }
    } else {
      Serial.printf("[HTTP] GET... failed, error: %s\n", http.errorToString(httpCode).c_str());
    }
    http.end();
  } else {
    Serial.printf("[HTTP} Unable to connect\n");
  }
```

```
    }
    pokusaj++;
    if(!poslano) delay(1000);
  }
}
```