

Univerzitet u Sarajevu
Elektrotehnički fakultet
Internet stvari (IoT) 2023/24

Projektna dokumentacija

Integracija RF senzora u openHAB sistem

Studenti:
Ehlimana Ćesić / 19040
Selma Gološ / 19041
Anela Mutilović / 19156

Sarajevo, februar 2024.g.

Sadržaj

1	Uvod	2
1.1	Postavka zadatka	2
1.2	Predviđeni za realizaciju projekta	2
1.3	Jasan pregled strukture dokumentacije	2
1.4	Osnovne smjernice i upute	2
2	Tehničke specifikacije	3
2.1	Hardverske komponente	3
2.2	Softverske komponente	6
2.3	Mrežna arhitektura	6
3	Postavljanje razvojnog okruženja, implementacija i integracija	7
3.1	Dodavanje i konfigurisanje krajnjih uređaja unutar openHAB sistema	7
3.2	Konfiguracija implementiranog openHAB pravila	9
4	Korištenje i upravljanje	11
5	Zaključak	12
6	Literatura	13

1 Uvod

1.1 Postavka zadatka

Potrebno je povezati jednostavni RF senzor (PIR senzor) sa openHAB sistemom. Za povezivanje RF senzora se preporučuje korištenje OpenMQTTGateway-a, ali je dozvoljeno koristiti i neki drugi sistem.

1.2 Preduvjeti za realizaciju projekta

Za realizaciju projekta potrebno je poznavanje rada sa mikrokontrolerom ESP32, rada sa MQTT brokerom i rada sa OpenHAB platformom pomoću Raspberry Pi.

Potrebne hardverske komponente:

- mikrokontroler ESP32
- Raspberry Pi
- RF 433MHz receiver modul
- RF PIR senzor

Potrebne softverske komponente:

- MQTT broker
- OpenHAB
- OpenMQTTGateway
- MQTTSX klijent
- Putty

1.3 Jasan pregled strukture dokumentacije

Poglavlje 2, Tehničke specifikacije, namijenjeno je za upoznavanje korisnika sa korištenim hardverskim i softverskim komponentama i samoj mrežnoj arhitekturi realizacije. Radi lakše vizualizacije, dati su shematski prikaz povezivanja hardverskih komponenti, kao i kompletna blokovska struktura realiziranog sistema.

Poglavlje 3, Postavljanje razvojnog okruženja, implementacija i integracija, koncipirano je na način da korisniku nudi detaljan opis koraka potrebnih za integraciju RF senzora u openHAB sistem. Ovo poglavlje služi kao vodič korisniku za samostalnu realizaciju projektnog zadatka. Od korisnika se za razumijevanje ovog dijela dokumentacije očekuje predznanje iz oblasti obrađenih na predmetu Internet stvari.

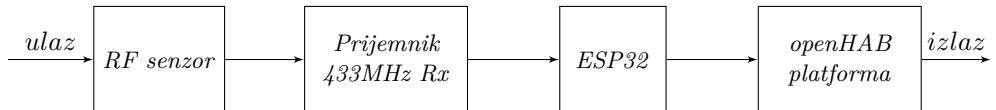
Poglavlje 4, Korištenje i upravljanje, namijenjeno je za krajnjeg korisnika sistema, odnosno predstavlja jednostavno uputsvo za korištenje finalnog proizvoda.

1.4 Osnovne smjernice i upute

Ukoliko je korisnik zainteresovan za samostalnu implementaciju zadatka, pročitati poglavlja 2 i 3. Za krajnjeg korisnika, relevantno je poglavlje 4 koje nudi jasne smjernice za uspješno korištenje razvijenog sistema.

2 Tehničke specifikacije

Blokovski prikaz kompletognog realiziranog sistema je prikazan u nastavku.



Slika 1: Blokovski prikaz realiziranog sistema

Ulas u sistem predstavlja pokret, koji je detektovan pomoću RF PIR senzora i u obliku radio talasa frekvencije 433MHz poslan na prijemnik. Prijemnik je povezan na mikrokontroler ESP32, koji pomoću instaliranog firmware-a vrši konverziju primljene informacije u MQTT poruku. Informacija se u ovom obliku šalje na openHAB platformu gdje će dalje biti prezentirana korisniku kao finalni izlaz opisanog sistema.

2.1 Hardverske komponente

Sistem detekcije prisustva realiziran je pomoću RF PIR senzora. Bežična komunikacija između RF senzora i ostatka sistema omogućena je putem RF 433MHz receiver-a obzirom da je to frekvencija koju koristi predajnik RF senzora. Prijemnik je povezan na mikrokontroler ESP32, na koji je instaliran modul esp32dev-rf sa OpenMQTTGateway platforme, dok je sam mikrokontroler povezan na računar. Korišteni RF PIR senzor je prikazan na slici 2.



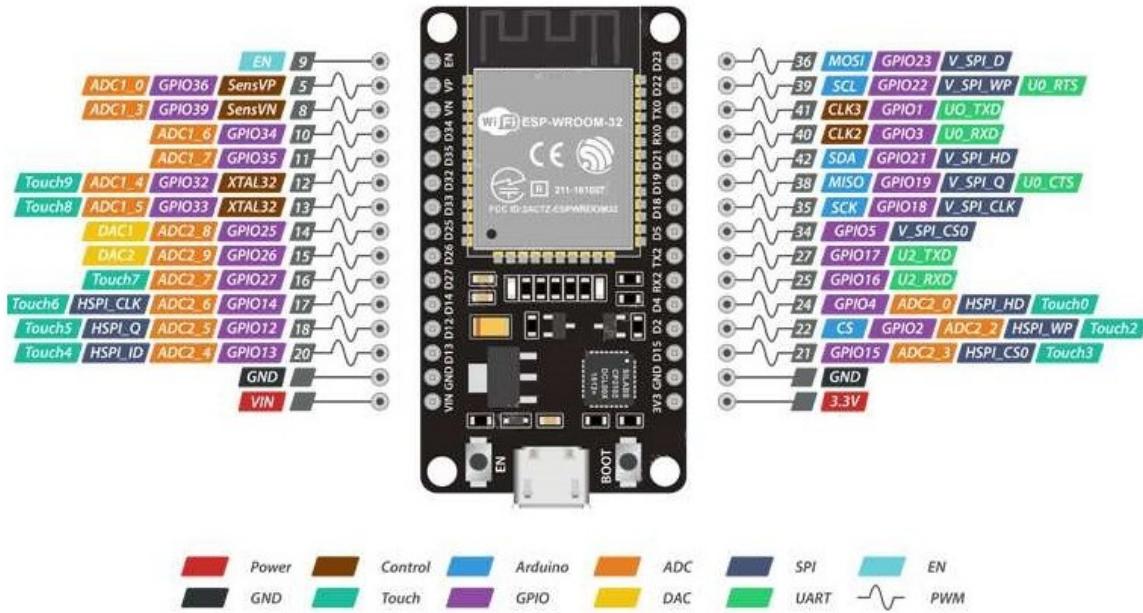
Slika 2: RF PIR senzor

OpenHAB platforma instalirana je na računar Raspberry Pi. Raspberry Pi je jako popularan računar baziran na ARM SoC-U. Konkretan model upotrebljen u realizaciji Raspberry Pi 2B.



Slika 3: Raspberry Pi 2B

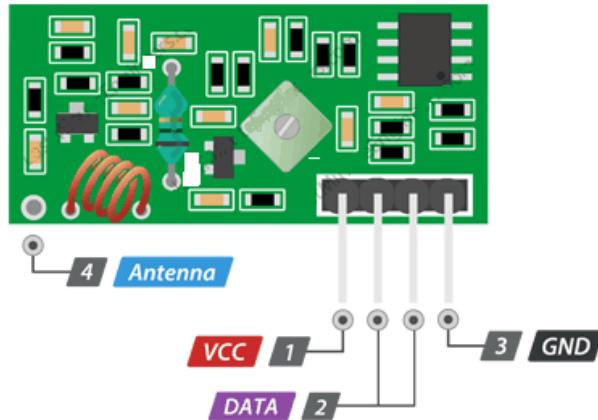
Na slici 4 je predstavljen raspored funkcija po pinovima za modul DevKit. Obzirom da se modul napaja direktno sa USB porta, nema potrebe za njegovim priključivanjem na vanjski izvor napajanja. Modul se programira preko USB porta. Raspored pinova RF prijemnika nalazi se na slici 5.



ESP32 Dev. Board / Pinout

Slika 4: Raspored pinova na modulu DevKit sa mikrokontrolerom ESP32

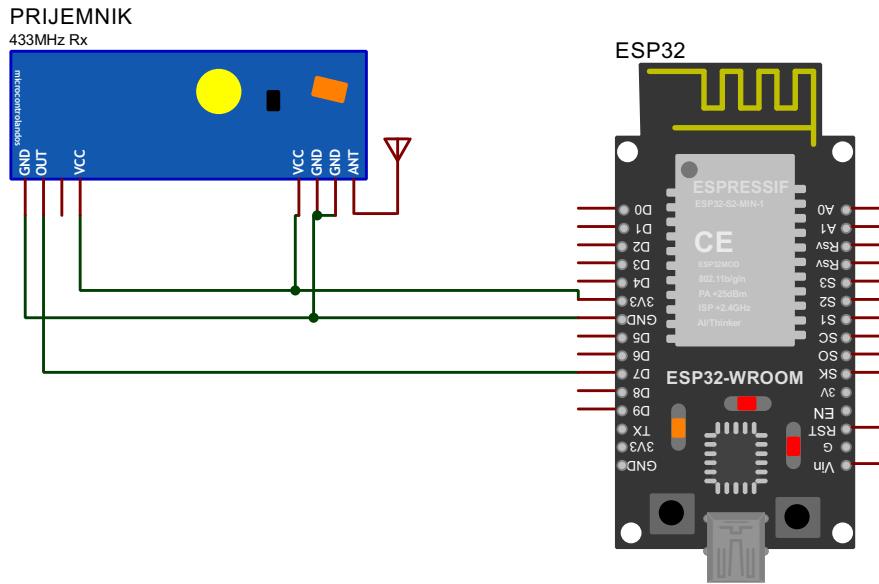
U nastavku je priložen raspored pinova korištenog RF prijemnika. Ovaj prijemnik predstavlja jednostavan modul i kao takvog lahko ga je inkorporirati u razne aplikacije.



433MHz Rx / Pinout

Slika 5: Raspored pinova RF prijemnika

Za uspješno ostvarivanje komunikacije neophodno je izlazne pinove RF prijemnika povezati sa pinovima na ESP32 mikrokontroleru. Ovi pinovi su pin za podatke (DATA), te pinovi za napajanje (VCC) i uzemljenje (GND). Pin za napajanje se povezuje sa pinom 3V3 na mikrokontroleru ESP32, dok se pin za uzemljenje povezuje na GND. Pin GPIO27 mikrokontrolera ESP32 služi za komunikaciju, te se iz tog razloga DATA pin sa prijemnika povezuje na isti. Shema povezivanja hardverskih komponenti je prikazana u nastavku.



Slika 6: Shematski prikaz povezivanja hardverskih komponenti

2.2 Softverske komponente

Raspberry Pi podržava različite operativne sisteme. Najčešće se koristi Raspberry Pi OS, te je isti upotrebljen za realizaciju ovog zadatka. Radi se o Linux distribuciji temeljenoj na Debianu, prilagođenoj specifičnostima Raspberry Pi računara. Aplikacija openHAB je pokrenuta na računaru u formi Docker kontejnera.

Nakon uspješne instalacije openHAB platforme¹, istoj je moguće pristupiti putem browsera na adresi <http://192.168.0.118:8080>, gdje 192.168.0.118 odgovara ip adresi korištenog Raspberry Pi.

Pri detekciji prisustva RF PIR senzorom, senzor pomoću svog ugrađenog predajnika šalje poruku prijemniku. Poruka koju prima integrirani 433MHz prijemnik nije adekvatna za dalje korištenje u sistemu, te ju je potrebno konvertovati u MQTT poruku. Ovo je omogućeno MQTT Gateway-om, tačnije modulom esp32dev-rf koji je instaliran na ESP32 mikrokontroler sa OpenMQTTGateway platforme. MQTT Gateway ima ulogu mosta između uređaja i MQTT broker-a tako što konverteuje ne-MQTT protokole u MQTT, čime se omogućava mrežna komunikacija. Konvertovanu MQTT poruku je moguće očitati na računaru pomoću Putty terminala ili MQTTX klijenta.

Nakon što RF PIR senzor detektira pokret, šalje odgovarajuću poruku koja se za dalje korištenje u sistemu obrađuje na već definiran način. Ova poruka signalizira sistemu da je senzor aktivan, te se ažurira stanje senzora na openHAB platformi u OPEN. Međutim, nakon detektiranog pokreta i aktivacije senzora, sistem ne dobiva nikakvu povratnu informaciju potrebnu za promjenu njegovog stanja iz OPEN u CLOSED, te bi po defaultu stanje senzora na openHAB platformi zauvijek ostalo OPEN. Kako bi se ovo korigovalo, implementirano je pravilo unutar openHAB platforme koje diktira sistemu da promijeni stanje senzora u CLOSED 2 sekunde nakon detekcije pokreta.

2.3 Mrežna arhitektura

U dатој aplikацији се implementiraju само они слојеви² за којима постоји потреба (слика 7).

Fizički sloj - Пренос података у облику радио таласа од RF PIR сензора до RF пријемника.

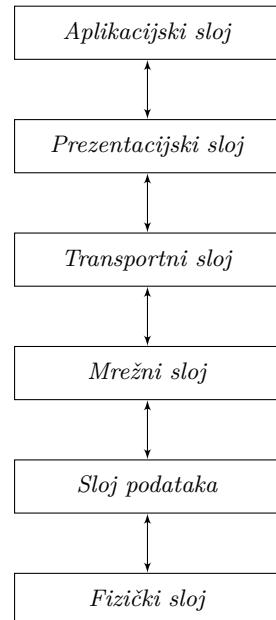
Sлој података - ESP32, пovezan на RF пријемник, обавља трансформацију примљених радио таласа у дигиталне податке адекватне за интерпретирање од стране дигиталног рачунара.

Mrežni sloj - Raspberry Pi пovezan са ESP32 преко WiFi мреже обавља рутирање пакета података од EPS32 до рачунара на којем је инсталiran openHAB. Одговарајући је за управљање IP адресама и организацију путање података унутар мреже.

Transportni sloj - Комunikација између ESP32 и Raspberry Pi-a.

Prezentacijski sloj - Односи се на синтаксу примљених информација. Интерпретирање стања сензора (on/off) помоћу синтакси OPEN и CLOSED респективно.

Aplikacijski sloj - Раčunar на којем је инсталiran openHAB, mqtt протокол за комуникацију са уредајима и контролом система.



Slika 7: Mrežna arhitektura

¹Detaljne upute za instalaciju i pokretanje openHAB platforme na Raspberry Pi računaru moguće je naći u sklopu Laboratorijske vježbe 8 iz predmeta Internet stvari (ak. 2023/2024)

²Dodata objašnjenja o odabiru slojeva koji čine mrežnu arhitekturu moguće je naći u sklopu predavanja Komunikacija u IoT sistemima iz predmeta Internet stvari (ak. 2023/2024)

3 Postavljanje razvojnog okruženja, implementacija i integracija

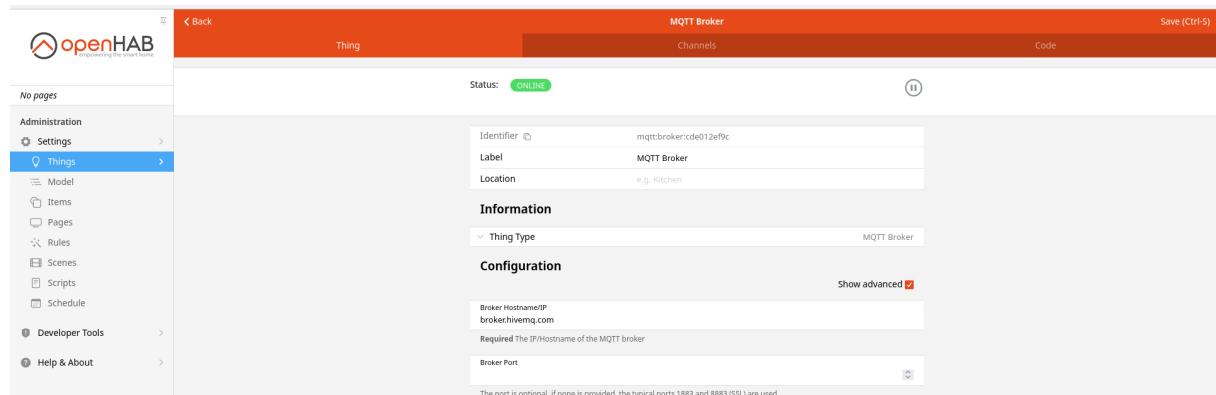
Integracija sistema razvijena je kroz openHAB platformu, koja je odabrana kao glavni okvir za upravljanje korištenim uređajima. Implementirano je pravilo koje osigurava ispravan rad sistema, odnosno kojim se omogućava kontrola rada korištenog RF PIR senzora. Nužnost za ovim pravilom ogleda se u činjenici da RF senzor ne šalje nikakvu poruku kada je potrebno stanje uređaja promijeniti u CLOSED. Napisanim pravilom, omogućena je ova promjena stanja 2 sekunde nakon detekcije prisustva, sve do ponovne detekcije istog.

3.1 Dodavanje i konfiguriranje krajinjih uređaja unutar openHAB sistema

OpenHAB razlikuje nekoliko pojmove vezanih za povezivanje i korištenje krajinjih uređaja:

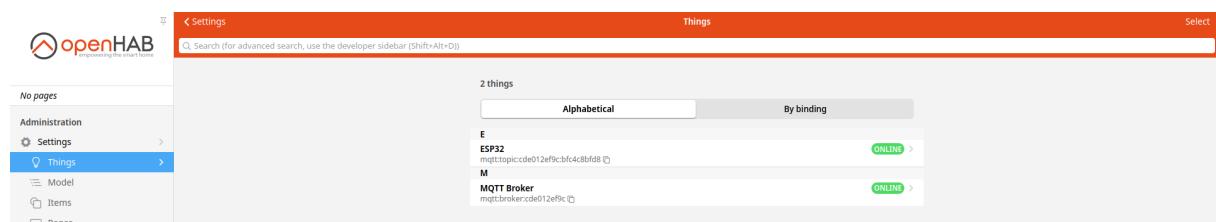
- **Binding** - omogućava definiranje načina povezivanja krajnog uređaja.
- **Thing** - predstavlja cijelokupni uređaj (hardverski ili softverski).
- **Channel** - fizički krajni uređaji obično imaju više "kanala".
- **Item** – logička reprezentacija kanala, svaki kanal mora imati svoj Item.

Kao Thing potrebno je dodati MQTT Broker i konfigurisati adresu i ostale parametre koji omogućavaju pristup brokeru. U konkretnom slučaju korišten je broker.hivemq.com. Za potrebe uspostavljanja komunikacije sa ESP32 modulom, neophodno je dodati Generic MQTT Thing. Ovaj thing je nazvan ESP32.



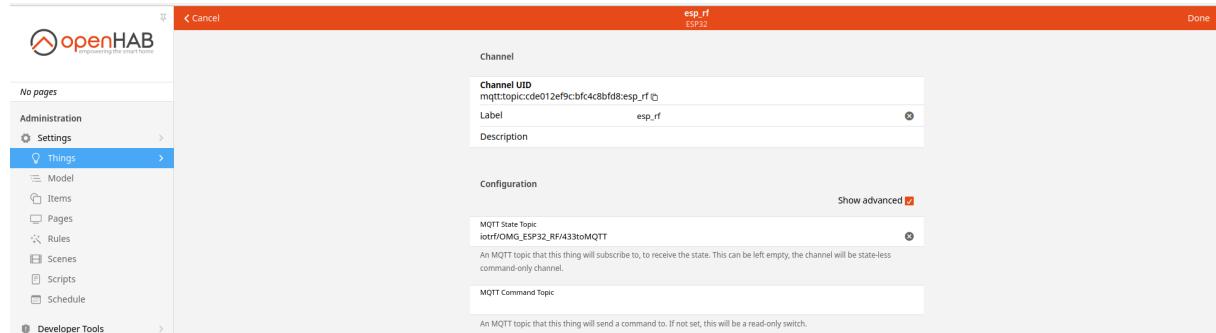
Slika 8: Prikaz konfiguracije Thing-a

Nakon uspješnog dodavanja ESP32 modula i MQTT brokera, bit će prikazana dva Thing-a (slika 9).



Slika 9: Prikaz nakon dodavanja MQTT brokera i modula ESP32

Od interesa je pokazivanje stanja jednog senzora, pa je potrebno kreirati jedan kanal u okviru ESP32 Thing-a. Definiran je kanal `esp_rf`, za koji se dalje dodaje Item, kako bi se mogao koristiti u okviru korisničkog interfejsa ili u okviru pravila.



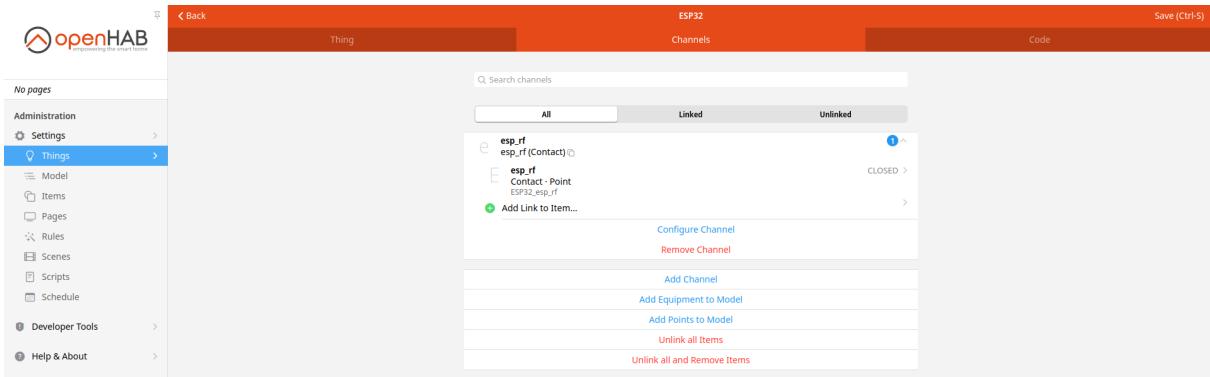
Slika 10: Prikaz konfiguracije kanala



Slika 11: Prikaz teme na koju je pretplaćen Thing unutar MQTTX-a

U dijelu za konfiguraciju kanala dodat je naziv teme u polje MQTT State Topic, na koju će se razmatrani *Thing* pretplatiti. Tema na koju je kanal pretplaćen je `iotrf/OMG_ESP32_RF/433toMQTT`, do koje se došlo pomoću MQTTX klijenta (slika 11). Na ovu temu se šalje svaka promjena stanja senzora. Ukoliko se izostavi ovaj dio kanal će biti proglašen kao *state-less command-only channel*.

Pri konfiguraciji Thing-a ESP32 potrebno je unijeti kao poseban slučaj prepoznavanja stanja **ON**, odnosno OPEN, ID koji je primljen kao **value** dio MQTT poruke. Ova informacija je pročitana unutar Putty terminala, a istu je moguće pročitati i u MQTTX klijentu. Dakle, nakon uspostavljanja veze putem Putty-a moguće je čitati podatke koje RF senzor šalje i interpretirati ih kako bi se utvrdilo stanje senzora. Kodirano stanje OPEN nalazi se pod oznakom **value** kao broj 2878985.



Slika 12: Prikaz implementiranog kanala unutar *Thing-a* ESP32

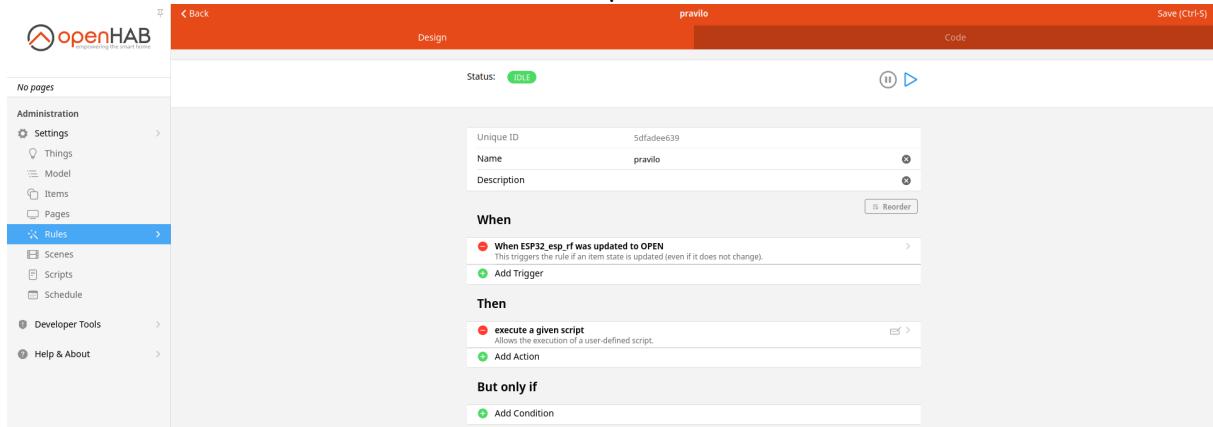
```
N: Send on /433toMQTT msg {"value":2878985,"protocol":1,"length":24,"delay":342,"tre_state":"-","binary":"001010111101100001001","raw":"10647,307,1011,334,1011,1036,294,331,1012,1035,297,328,1018,1032,298,1031,307,1031,302,1026,307,1024,310,317,1038,1013,311,1022,315,1017,315,311,1048,305,1044,304,1043,305,1043,305,1045,1007,321,311,1044,304,1044,1009"}  
N: [ MQTT->OMG ]: {"value":2878985,"protocol":1,"length":24,"delay":342,"tre_state":"-","binary":"001010111101100001001","raw":"10647,307,1011,334,1011,1036,294,331,1012,1035,297,328,1018,1032,298,1031,307,1031,302,1026,307,1024,310,317,1038,1013,311,1022,315,1017,315,311,1048,305,1044,304,1043,305,1043,305,1045,1007,321,311,1044,304,1044,1009"}  
N: Send on /RFtoMQTT msg {"active":2}  
N: Send on /SYSstoMQTT msg {"uptime":6011,"version":"v1.7.0","disc":false,"ohdisc":false,"env": "esp32dev-rf","freemem":211432,"mqttp": "1883","mqtts": false,"msgpcre": 16,"msgblk": 0,"maxq": 3,"minmem": 146044,"tempc": 56.67,"frestck": 2332,"eth": false,"rssi": -48,"SSID": "Lab220","BSSID": "00:0C:F6:21:29:91","ip": "192.168.0.110","mac": "0C:B8:15:C5:05:78","modules": ["WebUI", "RF"]}
```

Slika 13: Prikaz terminala Putty

3.2 Konfiguracija implementiranog openHAB pravila

Uredaj ispravno detektuje prisustvo, međutim za krajnju realizaciju potrebno je da se omogući kontrola njegovog upravljanja, na način da se automatski postavi na CLOSED 2 sekunde nakon detektovanja. Dakle, potrebno je izvršiti automatizaciju ponašanja RF senzora.

Ovakvo ponašanje sistema postignuto je definisanjem pravila u openHAB-u. Za početak, potrebno je definisati kada će se pravilo aktivirati. U ovom slučaju u polje **When** postavlja se slučaj kada je Item `ESP32_esp_rf` ažuriran u stanje OPEN.

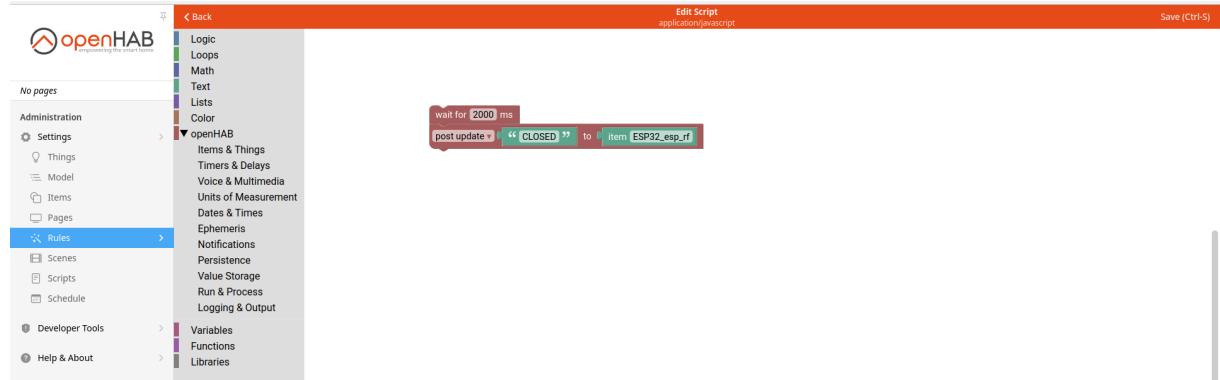


Slika 14: Detalji implementacije pravila

Polje **Then** indicira koju radnju je potrebno poduzeti u slučaju da je ostvaren uslov zadat u polju **When**. Za pisanje ovog dijela pravila odabrana je opcija upotrebe skripti. Potrebna skripta napisana je pomoću alata Blockly³ kojim se skripta komponira vizuelno sastavljanjem odgovarajućih blokova. Napisana skripta je jako jednostavna i sastoји se od dva bloka. Unutar polja *Timers & Delays*

³Osim navedenog načina, u openHAB-u moguće je definisati pravila i pomoću DSL-a (*Domain Specific Language*), te ECMAScript 5.1

odabran je blok *wait for* pomoću kojeg je zadato vrijeme koje treba proći od trenutka trigerovanja pravila do izvršavanja idućeg dijela koda. Ovo vrijeme je postavljeno na 2 sekunde. Drugi blok je iz polja *Items & Things*, *post update* unutar kojeg je odabrano da se aplikaciji diktira da ažurira stanje itema `ESP_esp_rf`, odnosno senzora u CLOSED, naravno nakon što je ispunjen prvi uslov, tj. nakon što su prošle 2 sekunde od aktiviranja senzora. Detalji o implementaciji pravila i skripte koja definiše strukturu pravila nalaze se na slikama 14 i 15.



Slika 15: Prikaz realizane skripte

Na ovaj način definisano je tačno kada će se pravilo aktivirati, koje akcije će se izvršiti i pod kojim uslovima. Omogućen je scenarij automatizacije koji odgovara specifičnim potrebama i preferencijama sistema.

Preko stranice Pages dodat je widget na Overview stranicu koji pokazuje trenutno stanje senzora. Overview je prva stranica koju korisnici vide pri posjeti openHAB-u. Ova stranica dostupna je svim korisnicima bez obzira da li imaju administratorski pristup ili ne. Dakle, na ovaj način omogućeno je praćenje stanja RF senzora. Oznaka OPEN ekvivalentna je stanju **ON**, dok je CLOSED ekvivalentan stanju **OFF**. Ukoliko bi se izostavila implementacija prikazanog pravila, stanje senzora nakon prve detekcije ostalo bi nepromijenjeno, odnosno senzor bi na openHAB platformi uvijek signalizirao da je detektiran pokret, čak i kada to nije tačno.



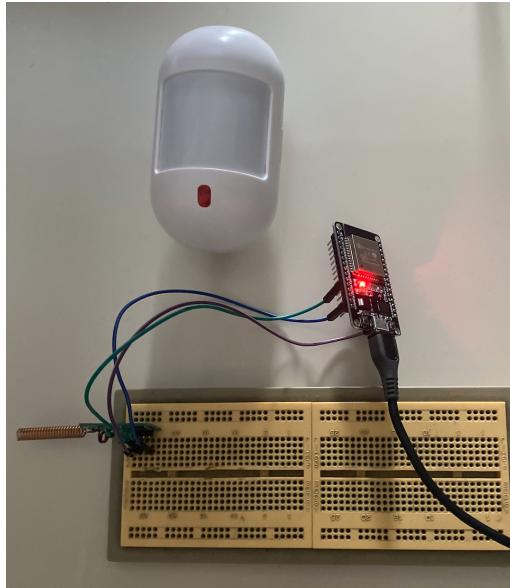
Slika 16: Overview pri detekciji prisustva



Slika 17: Overview 2 sekunde nakon detekcije prisustva

4 Korištenje i upravljanje

Integracija RF senzora unutar OpenHAB sistema omogućava krajnjem korisniku jednostavno i pouzdano praćenje prisustva u svom domu ili konkretnoj prostoriji. PIR senzor se postavlja na željeno mjesto u dozvoljenoj udaljenosti od receiver-a, ne većoj od 100m. U zavisnosti od toga da li je prisustvo detektovano ili ne, izgled senzora je prikazan na slici ispod.



(a) Bez detekcije prisustva



(b) Detektovano prisustvo

Unutar overview-a na openHAB sistemu moguće je pratiti stanje senzora. Dakle, ukoliko se detektuje prisustvo korisnik će moći vidjeti da se stanje senzora OPEN kao što je na slici 19a, a nakon 2 sekunde stanje će se promijeniti u CLOSED. Ovakva implementacija omogućava korisniku da prati prisustvo u prostorijama, a moguće je nadograditi da služi kao jednostavni alarm sistem, tako što bi se detekcijom prisustva emitovao neki zvuk.

FRIDAY, FEBRUARY 9
Overview

esp_rf
CLOSED

(a)

FRIDAY, FEBRUARY 9
Overview

esp_rf
OPEN

(b)

5 Zaključak

RF PIR senzor (*Radio Frequency Passive Infrared sensor*), kao tehnologija koja kombinuje radio talase i infracrveno zračenje, vrlo pouzdano detektuje prisustvo u prostoriji. Kao takav predstavlja nezamjenljiv dodatak za različite aplikacije, od kućne automatizacije pa do sigurnosnih sistema.

U pitanju je bežična komunikacija, što implicira da ne zahtijeva fizičko povezivanje sa drugim uređajima. Ovakav vid komunikacije od velikog je značaja ukoliko je potrebno senzor postaviti na mjesto koje je teško dostupno (gdje kablovi nisu praktični/estetski prihvativi). Stoga, omogućena je fleksibilnost u postavljanju senzora.

Integrисани RF receiver modul prima signale/informacije o detekciji prisustva, poslane od strane RF senzora. Ovaj modul povezan je na ESP32 mikrokontroler, koji obrađuje primljeni signal i šalje informacije o detekciji prisustva na računar putem WiFi mreže.

U kontekstu openHAB sistema, integracija RF senzora omogućava korisnicima da prate stanje senzora u prostoriji. Ukoliko je stanje OPEN detektovano je prisustvo u prostoriji, dok ukoliko je stanje CLOSED prostorija je prazna. Na ovaj način omogućeno je informisanje korisnika o događajima u realnom vremenu.

Postavljeni zahtjevi prema postavci projektnog zadatka su uspješno realizirani. Međutim, ostaje dosta prostora za poboljšanje i ažuriranje sistema sa sigurnosnog aspekta. Dodavanje dodatnih funkcionalnosti poput zvučnog alarma ili slanja obavještenja na mobilni telefon može znatno unaprijediti korisničko iskustvo i korisnost sistema detekcije prisustva.

Ovakva funkcionalnost postiže se adekvatnim programiranjem ESP32. Navedeni pristup problemu omogućit će korisniku da odmah reaguje na detekciju prisustva, čak i ukoliko se nužno ne nalazi u analiziranoj prostoriji. Ovako unapređen sistem postaje krucijalan dio sigurnosne infrastrukture, pružajući korisnicima mogućnost brze reakcije na potencijalne prijetnje.

6 Literatura

- [1] Predavanja na predmetu: Internet stvari, Elektrotehnički fakultet, Univerzitet u Sarajevu, ak.2023/2024
- [2] Laboratorijske vježbe na predmetu: Internet stvari, Elektrotehnički fakultet, Univerzitet u Sarajevu, ak.2023/2024
- [3] OpenMQTTGateway, <https://docs.openmqttgateway.com>
- [4] OpenHAB, <https://www.openhab.org/docs/>
- [5] 433MHz receiver modul datasheet,
<https://components101.com/modules/433-mhz-rf-receiver-module>