

RAZVOJ APLIKACIJE ZA NADZOR I UPRAVLJANJE SISTEMOM KUĆNE AUTOMATIZACIJE U IGNITION RAZVOJNOM OKRUŽENJU KORIŠĆENJEM MQTT PROTOKOLA**DEVELOPMENT OF AN APPLICATION FOR MONITORING AND CONTROL OF A HOME AUTOMATION SYSTEM IN IGNITION PLATFORM USING MQTT PROTOCOL**Milica Ujić, *Fakultet tehničkih nauka, Novi Sad***Oblast – MEHATRONIKA**

Kratak sadržaj – U ovom radu opisan je razvoj veb aplikacije za nadzor i upravljanje sistemom kućne automatizacije, radi olakšavanja života korisnika. U radu je data teorijska pozadina koncepta Interneta stvari i kućne automatizacije, a zatim je prikazana razvijena aplikacija. Za razvoj aplikacije korišćeno je Ignition razvojno okruženje. Komunikacija između uređaja i servera implementirana je pomoću MQTT protokola po Sparkplug specifikaciji, koji je ujedno i jedan od sve češće korišćenih protokola u IoT tehnologijama.

Ključne reči: Internet stvari, kućna automatizacija, MQTT, Sparkplug, Ignition

Abstract – This paper presents the development of a web application for monitoring and control of a home automation system. The paper gives a short overview of the Internet of Things and Home automation technologies, followed by the description of the developed application, which was built in Ignition platform, using MQTT and Sparkplug specification as one of the most popular communication protocols used in IoT technologies.

Keywords: Internet of Things, Home Automation, MQTT, Sparkplug, Ignition

1. UVOD

Poslednjih decenija svedoci smo velikih promena, kako u društvu tako i na celoj planeti. Posledice globalnog zagrevanja, brzinsko nestajanje šuma i prašuma, znatno povećan broj ljudi na planeti, sve veća urbanizacija kao i brojni drugi faktori doprineli su u mnogome ne samo promeni klime i drugih uslova na planeti, nego su u skladu sa tim počeli da se razvijaju i drugačiji, novi životni trendovi. Svest o sve težem stanju planete je počela masovno da se širi, pa se u skladu sa tim traže rešenja kako sprečiti dalje zagađivanje i propadanje za sada jedine sredine u kojoj je moguć život.

Istorija svedoči o velikom broju tehnoloških otkrića i promena koje su se dešavale, a koje su dosta doprinele povećanju kvaliteta života ljudi na planeti, počevši od Prve industrijske revolucije i otkrića parne mašine, preko Druge industrijske revolucije kojom je uveden niz tehnoloških promena pomoću otkrića sijalice, naizmenične struje i načina njenog prenosa, elektromotora, kao i motora sa unutrašnjim sagorevanjem.

NAPOMENA:

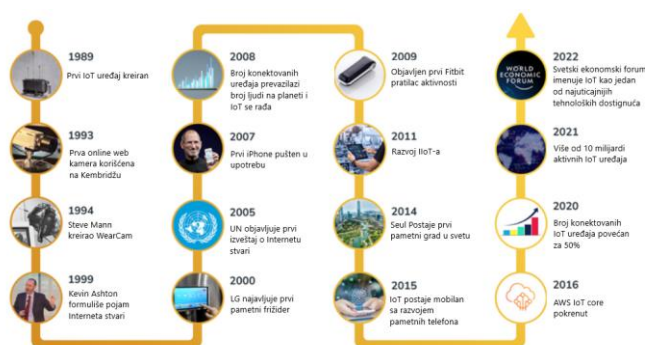
Ovaj rad proistekao je iz master rada čiji mentor je bila dr Ivana Šenk, vanr. prof.

Trend se nastavio i tokom Treće industrijske revolucije prelaskom sa mehaničkih i analognih tehnologija na digitalne.

Iz prethodno navedenog, uočava se da su se u relativno kratkom vremenskom periodu desile mnogobrojne promene koje su se u velikoj meri negativno odrazile na planetu i čije posledice su već danas uočljive. Sve ovo je doprinelo i razvoju potpuno nove, Četvrte industrijske revolucije čija je glavna karakteristika uklanjanje barijera između čoveka i mašina, a nosioci ovih promena su veštačka inteligencija (*engl. Artificial Intelligence, AI*), nanotehnologije, robotika, komunikacione tehnologije i Internet stvari koji će biti ugrađen u gotovo svaki kućni uređaj i koji će između ostalog, dovesti i do uštede energije.

2. INTERNET STVARI

Pojam Interneta stvari (*engl. Internet of Things, IoT*) se odnosi na milijarde fizičkih uređaja širom sveta opremljenih sensorima i komunikacionim mogućnostima za povezivanje na internet kako bi međusobno prikupljali i delili podatke. Na ovaj način, uređajima se omogućava tzv. „Digitalna inteligencija”, odnosno daje im se mogućnost da bez čovekovog učešća koriste potrebne podatke u odgovarajuće vreme i na taj način sami upravljaju svojim radom **Error! Reference source not found.** Istorijski razvoj IoT-a dat je na 0.



Slika 1. Prikaz razvoja Interneta stvari kroz vreme [2]

Među brojnim benefitima koje uključuje korišćenje IoT-a, izdvajaju se [3]:

- Efikasno korišćenje resursa,
- Minimizacija ljudskog napora,
- Ušteda vremena,
- Razvoj AI kroz IoT,
- Poboljšanje bezbednosti sistema.

2.1. Kućna automatizacija

Kućna automatizacija predstavlja poddomen Interneta stvari, a ostvaruje se povezivanjem kućnih uređaja od interesa na internet radi međusobne komunikacije, deljenja i obrade podataka (slika 2).



Slika 2. Skica koncepta upravljanja i praćenja kućnih uređaja putem aplikacije [4]

Tipični zadaci u sistemu kućne automatizacije su upravljanje osvetljenjem, uređajima (često preko upravljanja utičnicama), roletnama, električnim bravama, grejanjem, kao i nadzor temperaturnih senzora, senzora prisustva, detektora dima ili kontakata prozora. Ključna reč „pametni“ u sistemu kućne automatizacije ukazuje na to da u pozadini sistema postoji inteligentno upravljanje kojim se mogu izvršavati zadate radnje, npr. upaliti svetlo kada neko uđe u sobu, proveriti da li su zaključana vrata kada vlasnik izađe iz kuće i sl.

Dugoročni benefiti uvođenja kućne automatizacije mogli bi biti opšte unapređenje kvaliteta života, kroz sisteme koji omogućavaju jednostavnije i brže obavljanje svakodnevnih radnji, udobnost i bezbrižnost usled prilagođavanja ovakvih sistema navikama korisnika, bezbednost zahvaljujući različitim sistemima zaštite i nadzora, kao i smanjenje potrošnje energije optimizacijom rada uređaja, npr. automatskim isključivanjem svetala i električnih uređaja kada nisu potrebni.

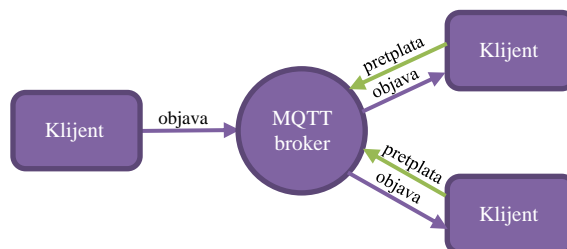
3. KOMUNIKACIONI PROTOKOLI U INTERNETU STVARI

Da bi se uspostavila komunikacija između pojedinačnih pametnih uređaja i servera potreban je odgovarajući komunikacioni protokol koji predstavlja sistem pravila koja omogućavaju da dva ili više entiteta komunikacionog sistema razmenjuju informacije. Protokol definiše pravila, sintaksu, semantiku i sinhronizaciju komunikacije, kao i mogućnost oporavka od grešaka. Protokoli mogu biti hardverski, softverski i kombinacija oba. Protokoli za komunikaciju s pametnim uređajima su jednostavni, ali je njihova složenost uslovljena hardverskim i softverskim mogućnostima samih *IoT* uređaja. Zbog toga se u *IoT* sistemima koriste prilagođeni namenski protokoli. U ovom radu, biće opisan *MQTT* protokol i *Sparkplug* specifikacija koji su korišćeni za komunikaciju između klijenata i servera u razvijenoj aplikaciji.

3.1. MQTT protokol

MQTT (engl. *Message Queue Telemetry Transport*) protokol, kao jedan od najčešće korišćenih protokola u *IoT*

aplikacijama, predstavlja *M2M* (engl. *machine to machine*) protokol aplikativnog nivoa koji koristi *TCP/IP* komunikacioni stek. Veoma je brz protokol koji koristi malo resursa. Ovo je mali protokol čije poruke u paketima ne zauzimaju mnogo komunikacionog prostora, pa se može istovremeno slati veći broj poruka. Baziran je na komunikacionom obrascu pretplate i objave (engl. *publish/subscribe*) poruka na određenim temama (kanalima), kao što je prikazano na slici 3.



Slika 3. *MQTT* sistem, funkcioniše po obrascu pretplate i objave (*publish/subscribe*)

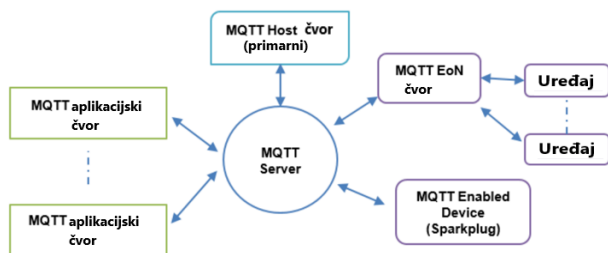
Svaki *MQTT* sistem sastoji se od *MQTT* brokera ili servera, koji predstavlja centralnu tačku sistema i *MQTT* klijenata koji se povezuju sa brokerom i preko njega komuniciraju. Kako bi se izvršilo usmeravanje poruka od predajnih klijenata ka prijemnim klijentima, definišu se odgovarajuće teme za komunikaciju. Uređaj koji šalje poruke, objavljuje poruku na određenu temu, dok se uređaj koji prima poruke pretplaćuje na određenu temu. *MQTT* broker, kao centralna tačka sistema, prati sve dostupne teme u sistemu, kao i uređaje koji su pretplaćeni na odgovarajuće teme, te raspoređuje sve pristigle poruke od klijenata koji objavljuju poruke svim uređajima koji su pretplaćeni na datu temu.

MQTT komunikaciona sesija je podeljena u četiri faze: konekcija, autentifikacija, komunikacija i prekid. Klijent započinje komunikaciju kreiranjem *TCP/IP* konekcije ka brokeru. Povezivanje i prekid veze sa brokerom obavlja se na sledeći način: *MQTT* klijent se povezuje sa brokerom slanjem *CONNECT* poruke, na koju broker odgovara *CONNACK* porukom. Svaka poruka se sastoji od zaglavlja od dva bajta koje je opciono, same poruke (podataka) i *QoS* (engl. *Quality of Service*) nivoa. *QoS* se u *MQTT* protokolu odnosi na garanciju isporuke poruke, a sadrži i informaciju o tome da li se zahteva od brokera da poruku sačuva i isporuči naknadno konektovanim klijentima (*retain* polje). *QoS* nivoi višeg reda su pouzdaniji, međutim, sa druge strane, imaju veće kašnjenje i traže bolji protok i višestruku razmenu potvrda između klijenta i brokera, pa klijenti koji se pretplaćuju mogu spram svojih potreba da biraju *QoS* nivo sa kojim žele da primaju poruke.

3.1.2. Sparkplug specifikacija

Za razliku od „običnog“ *MQTT* protokola gde teme i poruke možemo proizvoljno da određujemo spram primene, *Sparkplug* specifikacija definiše tačan format poruka i način njihovog slanja. Ideja koja je dovela do nastanka *Sparkplug* specifikacije je da se u odnosu na prvobitni *MQTT* protokol, dodatno obezbede i mehanizmi koji će ovakvim aplikacijama omogućiti i da se prati stanje bilo kog *MQTT device* čvora u infrastrukturi [5]. Po ovoj specifikaciji, ne postoji mogućnost specijalnog definisanja imena tema (engl. *Namespaces*) i formata

poruka (*engl. Payload*) koje se šalju ili na koje se pretplaćuje (*publish/subscribe*), već su oni unapred definisani. Komponente koje čine infrastrukturu jednog ovakvog sistema su prikazane na slici 4..



Slika 4. MQTT Sparkplug B infrastruktura [5]

Svi MQTT klijenti koji koriste Sparkplug specifikaciju treba da isprate sledeću strukturu pri definisanju tema: [name-space/group_id/message_type/edge_node_id/\[device_id\]](#).

4. RAZVOJ APLIKACIJE ZA UPRAVLJANJE UREĐAJIMA U SISTEMU KUĆNE AUTOMATIZACIJE

U okviru ovog rada razvijena je aplikacija za kućnu automatizaciju zasnovana na IoT (*engl. Internet of Things, Internet stvari*) tehnologijama, s ciljem lakšeg i jednostavnijeg upravljanja uređajima u kućnom okruženju i njihovog nadzora, pomoću bilo kog prenosnog uređaja (mobilnog telefona, tableta, računara i slično). Aplikacija je razvijena u Ignition razvojnom okruženju [6] uz primenu MQTT komunikacionog protokola po Sparkplug specifikaciji, i povezana je sa simuliranim sistemom kućne automatizacije sa umreženim pametnim uređajima. Za razvoj aplikacije korišćeno je Ignition Maker Edition izdanje razvojnog okruženja, koje je namenjeno za manje nekomercijalne projekte.

U ovom poglavlju dat je detaljan pregled svih opcija koje aplikacija poseduje, a koje su namenjene što lakšem i jednostavnijem upravljanju kućnim uređajima od strane korisnika. Cilj je bio napraviti aplikaciju koja je vrlo jednostavna za upotrebu, a da pri tom bude fleksibilnog dizajna i obezbedi mogućnost pristupa sa različitih uređaja koji imaju drugačije veličine i orijentacije ekrana. Pored vizualizacije prostorija i uređaja u kući postoji i dodatna opcija prikazivanja vremenskih uslova u gradu, a koji direktno utiču na ponašanje određenih uređaja u kući (npr. u zavisnosti od položaja Sunca definiše se pozicija roletni i slično).

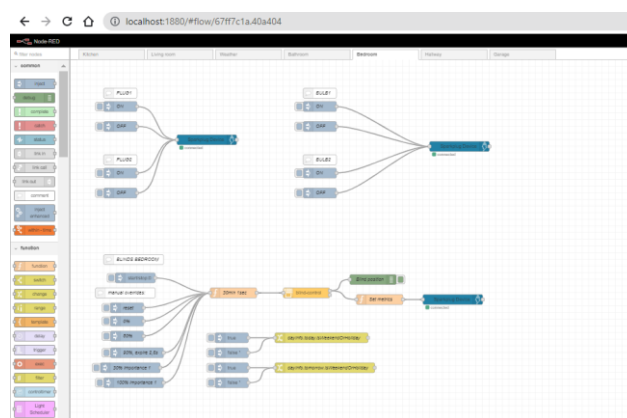
4.1. Ignition razvojno okruženje

Ignition softver predstavlja razvojno okruženje za izradu SCADA i IoT aplikacija, a proizvod je kompanije Inductive Automation. Ignition se može posmatrati kao centralna platforma koja nudi mogućnost povezivanja sa različitim uređajima poput: ekrana na dodir, pametnih telefona i tableta, neograničenog broja baza podataka, senzora, vaga, barkod skenera, itd. a pored toga ima ugrađene i drajvere za komunikaciju sa programabilnim logičkim kontrolerima (PLC-ovima) preko OPC standarda (*engl. Open Platform Communications*), povezivanje sa ERP (*engl. Enterprise resource planning*) sistemima, itd. Velika prednost ovog softvera je ta što pruža sve veb

usluge i može da se pokreće bilo gde, sa različitih uređaja. Za izradu interfejsa aplikacije korišćen je Perspective modul zbog prednosti u pogledu prilagodljivog dizajna i kompatibilnosti koja se ogleda u korišćenju uređaja sa različitim operativnim sistemima i veličinama ekrana.

4.2. Simulacija rada uređaja. Node-RED program.

Akcent ovog rada je na razvoju softverske aplikacije za nadzor i upravljanje uređajima u kućnoj automatizaciji, ali kako bi sistem kućne automatizacije bio potpun odabran je pristup gde se rad različitih kućnih uređaja simulira putem Node – RED programa (slika 5). Ovaj softver predstavlja moćan alat za programiranje koji omogućava razvoj upravljačkih tokova za aplikacije zasnovane na događajima, a sastoji se iz grafičkog editora u kom se povezuju funkcijski čvorovi koji mogu da postignu različite mogućnosti [7]. Implementacija čvorova se zasniva na Java Script programskom jeziku.



Slika 5. Uređaji u spavaćoj sobi. Struktura tokova.

U okviru prikaza sobe može se uočiti da su za sve uređaje korišćeni Sparkplug uređaji kod kojih se teme ne mogu proizvoljno definisati, već je format poruka i način slanja definisan samom specifikacijom, za razliku od običnog MQTT protokola gde teme i poruke proizvoljno mogu da se odrede. Kod Sparkplug uređaja (*engl. Sparkplug Devices*), moraju da se definišu metrike kako bi se poruka objavila na određenoj temi. Metrika mora da se definiše u samom uređaju ali i na suprotnoj strani sa koje se signal šalje kako bi se uspostavila međusobna komunikacija i pravilan prenos poruke.

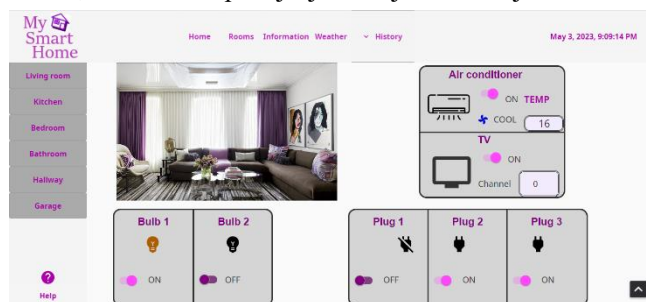
Poseban tok (*Flow Weather*) u okviru Node-RED simulacije posvećen je vremenskim parametrima koji su od posebnog značaja korisnicima aplikacije. Da bi se u svakom trenutku imala realna slika o vremenskim parametrima u gradu, iskorišćen je servis vremenske prognoze sa sajta <https://openweathermap.org/>. Podaci su generisani u JSON formatu, a potom su u simulaciji parsirani samo podaci od interesa za aplikaciju: temperatura, vlažnost i pritisak vazduha, naziv grada, oblaci i brzina vetra.

4.2. Vizualizacija sistema

Pri izradi aplikacije My Smart Home težilo se tome da korisnički interfejs bude što intuitivniji i jednostavniji za upotrebu, a da pri tom pruža sve neophodne informacije korisniku. Cilj je bio razviti mobilnu aplikaciju sa jednostavnim pristupom putem linka na veb pretraživaču, nezavisno od korisnikovog softvera ili veličine ekrana

uređaja pošto je aplikacija u tom pogledu vrlo fleksibilna. Na ovaj način se korisnicima pruža mogućnost nadzora i upravljanja kućnim uređajima na vrlo efikasan i koristan način čak i kada nisu kod kuće.

Da bi se omogućila prilagodljivost različitim vrstama ekrana korišćena su dva tipa kontejnera za smeštanje komponenti: kontejneri sa prekidom (*engl. Breakpoint kontejneri*) i fleksibilni kontejneri. Kreirano je više prikaza, po jedan za svaku prostoriju, vremenske informacije, arhivirane podatke, izveštaje, rukovanje alarmima, i opšte informacije o projektu. Na slici 6 je prikazan jedan od ekrana, za nadzor i upravljanje uređajima u dnevnoj sobi.



Slika 6. Prikaz ekrana za nadzor i upravljanje uređajima u dnevnoj sobi

Promena svake promenljive odnosno taga se direktno oslikava u prozorima za vizualizaciju, pomoću podešavanja za *Tag Binding*-e u sklopu komponenti.

Pored prikaza svih prostorija i njihovih uređaja ponaosob, u okviru padajućeg menija *History* postoji opcija praćenja potrošnje uređaja u željenom vremenskom periodu, opcija generisanja izveštaja o aktivnosti uređaja ili opcija praćenja alarmnih stanja u sistemu koja je izuzetno korisna jer u slučaju nedozvoljenih stanja šalje korisniku obaveštenje kako bi moglo pravovremeno da se odreaguje.

Da bi se omogućilo skladištenje vrednosti promenljivih u *PostgreSQL* bazu podataka i njihovo dalje korišćenje u aplikaciji neophodno je bilo konfigurisati bazu podataka u *Ignition Gateway*-u. Pored ovih podešavanja, bilo je još potrebno posebno za svaki tag za koji želimo da mu se vrednost arhivira omogućiti *History Enabled* opciju i potom navesti ime baze podataka u kojoj želimo da se vrednosti smeštaju. S obzirom da je korišćena *Sparkplug* specifikacija, vrednosti tagova su smeštene u folder *Edge Nodes/My Devices/Node-Red/Sparkplug Device/house./* u okviru *Tag Browser*-a u *Ignition*-u.

5. ZAKLJUČAK

U ovom radu je predstavljena ideja i osnovni koncept pojma *Interneta stvari* (*engl. Internet of Things – IoT*). Dat je kratak osvrt na istorijski razvoj i predstavljeni su benefiti njegove primene. Akcenat rada je stavljen na kućnu automatizaciju koja je poslednjih godina u sve većoj ekspanziji u domaćinstvima širom sveta. Zatim je opisan razvoj konkretne aplikacije za nadzor i upravljanje uređajima u okviru sistema kućne automatizacije, uz opis neophodnih komponenti za razvoj ovakvog sistema.

Za izradu je izabrano *Ignition* razvojno okruženje, te je iz tog razloga dat i osvrt na ovo okruženje i njegove prednosti pri izradi *IoT* aplikacija. Akcenat rada i jeste upravo na softverskom rešenju problema vezanih za upravljanje uređajima, dizajnu, fleksibilnosti i efikasnosti. Razvijena aplikacija omogućava uvid u aktivnost i potrošnju svakog uređaja pojedinačno, što daje mogućnost korisnicima da detaljno analiziraju i optimizuju korišćenje svojih uređaja, te da pametno troše energiju čime se i širi svest o problemima sa kojima se suočava čovečanstvo.

Komunikacija sa uređajima unutar aplikacije se obavlja po *MQTT* protokolu koji je jedan od najčešće korišćenih protokola u *IoT* aplikacijama. Dat je pregled ovog protokola, a posebno potpoglavlje je posvećeno i *Sparkplug* specifikaciji koja je iskorišćena za što efikasniju komunikaciju.

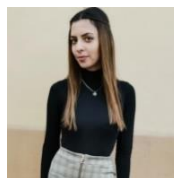
Za testiranje rada sistema, svi uređaji kućne automatizacije su simulirani u *Node-RED* okruženju, čime je omogućena simulacija rada, uključivanje i isključivanje uređaja, kao i programiranje rada po određenom obrascu.

U realnosti, pri korišćenju ovog sistema bi se umesto *Node-RED* simuliranih uređaja koristili stvarni uređaji koji bi u mnogome olakšali život korisnicima.

4. LITERATURA

- [1] P. Sethi and S. R. Sarangi, "Internet of Things: Architectures, Protocols, and Applications," *Journal of Electrical and Computer Engineering*, Vol. 2017
- [2] A brief history of Internet of Things, pristup 30. 5. 2023. preko <https://bytebeam.io/blog/a-brief-history-of-internet-of-things/>
- [3] B. L. Risteska Stojkoska and K. V. Trivodaliev, "A review of Internet of Things for smart home: Challenges and solutions," *Journal of Cleaner Production*, vol. 140, pp. 1454-1464, 2017.
- [4] Why is Smart home more of a necessity than a want, pristup 30. 5. 2023. preko: <https://www.homengine.ae/Blog/why-is-smart-home-more-of-a-necessity-than-a-want/9>
- [5] MQTT Sparkplug B Specification, pristup 30. 5. 2023. preko: <https://sparkplug.eclipse.org/specification/version/-2.2/documents/sparkplug-specification-2.2.pdf>.
- [6] Ignition documentation: pristup 30. 5. 2023. preko: <https://docs.inductiveautomation.com/>
- [7] Node-RED, pristup 30. 5. 2023. preko <https://nodered.org/>

Kratka biografija:



Milica Ujić rođena je u Sokocu 02.02.1998. god. Diplomirala je na Fakultetu tehničkih nauka u Novom Sadu 2021. godine sa prosečnom ocenom 9.65 na osnovnim akademskim studijama Mehatronike, kada je upisala i master studije na istom studijskom programu. kontakt: ujicmilica98@gmail.com