



UNIVERZITET U NIŠU
ELEKTRONSKI FAKULTET



Platforma za Internet stvari bazirana na arhitekturi mikroservisa

Master rad
Studijski program: Računarstvo i informatika
Modul: Inženjerstvo podataka

Student:

Danilo Veljović, br. ind. 1120

Mentor:

Doc. dr. Aleksandar Stanimirović

Niš, septembar 2021. god.

Univerzitet u Nišu
Elektronski fakultet

Platforma za Internet stvari bazirana na arhitekturi mikroservisa

Internet of Things platform based on microservice architecture

Master rad
Studijski program: Računarstvo i informatika
Modul: Inženjerstvo podataka

Student: Danilo Veljović

Mentor: Doc. dr. Aleksandar Stanimirović

Zadatak: *Proučiti i prikazati moguće arhitekture platformi za Internet stvari. Detaljno izučiti platformu zasnovanu na arhitekturi mikroservisa. Implementirati primer platforme zasnovan na arhitekturi mikroservisa.*

Datum prijave rada: 15.07.2021

Datum predaje rada:

Datum odbrane rada:

Komisija za ocenu i odbranu:

1. Doc dr Aleksandar Stanimirović, Predsednik Komisije

2. <Prof. / Doc dr Ime Prezime>, Član

3. <Prof. / Doc dr Ime Prezime>, Član

PLATFORMA ZA INTERNET STVARI BAZIRANA NA ARHITEKTURI MIKROSERVISA

Sažetak

Internet stvari je pojam koji poslednjih godina sve više dobija na važnosti. Usled razvoja "*pametnih*" senzora moguće je pribaviti sve detaljnija merenja i slati ih na obradu do nekog "*edge*" uređaja. *Edge* uređaji imaju sve veću moć obrade i mogu da u realnom vremenu skladište i analiziraju primljene vrednosti, kao i da vrše akciju. U ovom radu, predloženo je rešenje koje objedinjuje sve ove funkcionalnosti, u vidu *cloud platforme* za internet stvari. Cloud platforme su sve popularnije, jer omogućavaju pristup podacima sa bilo kog uređaja koji može da se konektuje na internet. Proučavane su različite arhitekture cloud platformi. Neke od arhitektura koje su proučene su monolitna arhitektura i mikroservisna arhitektura. Sagledane su prednosti i mane svake arhitekture, kao i neki primeri korišćenja. Kao fokus ovog rada, posebno je proučavana arhitektura mikroservisa. Arhitektura mikroservisa je sve popularnija, naročito za web aplikacije. Omogućava jednostavno skaliranje, postavljanje na server i razdvajanje servisa. Ovim se sistem strukturiše u celine koje mogu međusobno nezavisno da funkcionišu. U okviru arhitekture mikroservisa, proučavano je kako se mogu izdvajati nezavisne funkcionalnosti i kako se mogu dodeljivati posebnim servisima. Poseban akcenat je dat na razmatranje, koji servisi bi trebalo da se izvršavaju na *edge* delu sistema, a koji na *cloud* delu sistema.

U praktičnom delu rada implementiran je prototip platforme za internet stvari baziran na arhitekturi mikroservisa. Aplikacija je kreirana za svrhe agrikulture, odnosno navodnjavanja zemljišta i predikcije kvaliteta zemlje. Razvijeni prototip sistema se sastoji od servisa koji se bave skladištenjem podataka, analizom podataka, izvršavanjem upita i akcijom. Na kraju rada se diskutuje predloženo za poboljšanje sistema, kao i još neki mogući načini primene.

Ključne reči: arhitektura mikroservisa, Internet stvari, platforma, cloud, edge.

INTERNET OF THINGS PLATFORM BASED ON MICROSERVICE ARCHITECTURE

ABSTRACT

Internet of things is becoming an increasingly relevant topic in the recent years. Because of the development of smart sensors, it is possible to obtain very frequent and detailed measurements and to send them to an edge node to be processed. The processing power of edge devices is increasing, and they can store and analyze the received data in real-time, as well as do an actuation. In this thesis, a solution is suggested to unite all of these functionalities in the form of a cloud platform. Cloud platforms are all the more popular solution these days, because they allow access to the data from any device that can be connected to the internet. Different kinds of cloud architectures were explored. Some of those are monolithic architecture and microservice architecture. Benefits and downsides to each architecture were studied, as well as some examples of use. The focus of this work is the microservice architecture. Microservice architecture is ever more popular architecture solution especially for the web applications. They allow simple scaling of services, independent deployment and loosely coupled services. This way the system is structured as a set of independent parts that can work together, without being coupled. The way in which the functionalities of the system can be discerned and how can they be assigned to services is also explored. A special accent is placed on the investigation of which services should be deployed to an edge device, and which should be deployed on the cloud.

The second part of the thesis contains an implementation of a prototype of the cloud platform based on the microservice architecture. Application is created for the purposes of agriculture, irrigation of the land and prediction of quality of the soil. The developed prototype consists of services that store the data, analyze it, query it and send actuation requests. Finally, in the final section of the thesis, suggestions are given on how to improve the platform and few new ways on where it could also be used.

Keywords: microservice architecture, internet of things, platform, cloud, edge.

SADRŽAJ

1. UVOD 6

2. IOT SISTEMI 8

2.1. IoT sistemi.....	8
2.2. Karakteristike IoT sistema.....	9
2.3. Arhitektura IoT sistema.....	10
2.4. Topološka arhitektura IoT sistema.....	12

3. ARHITEKTURE IOT PLATFORMI 14

3.1. Životni ciklus IoT sistema.....	14
3.2. IoT platforma.....	16
3.3. Monolitna arhitektura.....	17
3.4. Mikroservisna arhitektura.....	17
3.5. Komparativna analiza arhitektura.....	17

4. IMPLEMENTACIJA IOT PLATFORME ZASNOVANE NA ARHITEKTURI MIKROSERVISA18

5. PREDLOZI ZA POBOLJŠANJE RAZVIJENOG PROTOTIPA 19

6. ZAKLJUČAK 20

LITERATURA 21

1. UVOD

Internet stvari je pojam koji poslednjih godina sve više dobija na važnosti. Zbog velike količine podataka koju je danas moguće preneti internetom, moguće je u realnom vremenu pratiti stanje bolnica, nuklearnih elektrana, poljoprivrednih imanja itd. Usled razvoja "pametnih" senzora, moguće je prikupiti sve detaljnija merenja u sve kraćim vremenskim intervalima. Danas senzori mogu da vrše merenja u intervalima od par milisekundi, što može da rezultira u po više stotina merenja u sekundi. Ovako veliku količinu podataka, potrebno je dostaviti do edge ili cloud uređaja. Na edge ili cloud sloju se prikupljeni podaci perzistiraju, analiziraju, i ako su merenja takva da se neki uslov ispunio, vrši se i akcija određenih uređaja.

Funkcionalnosti poput skladištenja primljenih podataka, zatim njihova analiza, i akcija uređaja predstavljaju srž svakog sistema za internet stvari. Sistemi za internet stvari moraju da objedinjuju sve ove funkcionalnosti i zbog toga se vrlo često i nazivaju platformom za internet stvari. Platforme su popularno rešenje jer omogućavaju pristup podacima sa bilo kog uređaja koji se može povezati na internet. Ovim je omogućena cross-platform podrška, koju svi korisnici očekuju. Potrebno je omogućiti da platforma funkcioniše, čak i ako je broj podataka koji treba da se obradi reda veličine nekoliko desetina gigabajta ili par desetina terabajta. Za sisteme koji su projektovani u monolitnoj arhitekturi, ovolika količina podataka može predstavljati problem.

Monolitni sistemi teže skaliraju. Ako odjednom u sistem krene da pristiže neuobičajeno velika količina podataka, potrebno je da se cela aplikacija skalira. Nije moguće samo imati redundantne instance jednog dela sistema. U monolitnim sistemima se teže dodaju nove funkcionalnosti. Vrlo često, usled kvara na jednom delu sistema, cela aplikacija prestaje da funkcioniše. Ovo može da izazove velike probleme, ili čak nedetektovane događaje na koje je bilo potrebno reagovati.

Kao predloženo rešenje problema monolitnih sistema data je arhitektura mikroservisa. Arhitektura mikroservisa je standard u projektovanju web aplikacija današnjice. Omogućava podelu sistema na nezavisne servise, omogućava nezavisno postavljanje na server tih servisa i njihovo jednostavno skaliranje. Arhitektura mikroservisa uvodi i određene probleme poput intenzivnije komunikacije između servisa, težeg održavanja konzistentnosti podataka u sistemu itd.

Arhitektura mikroservisa predstavlja rešenje mnogih problema kod aplikacija internet stvari. Vrlo često je potrebno skladištiti ili analizirati veliku količinu podataka. Ako bi postojao nezavistan servis koji skladišti podatke, on bi se mogao skalirati u toj meri, da sistem može da nastavi da normalno funkcioniše. Ako iz nekog razloga, servis za analitiku ne funkcioniše, to neće dovesti do pada celog sistema. Sistem razvijen na ovaj način je pouzdan, otporan na padove, ima nizak down time i skalabilan je.

U narednom poglavlju biće proučene različite arhitekture IoT platformi. Proučiće se prednosti i mane svake arhitekture. Poseban akcenat biće dat na arhitekturi mikroservisa, budući da je ona fokus ovog rada. U trećem poglavlju je opisana implementacija IoT

platforme bazirane na arhitekturi mikroservisa. Četvrto poglavlje je diskusija o prednostima i manama ovog sistema, kao i o mogućnostima za poboljšanje implementiranog prototipa.

2. IOT SISTEMI

2.1. IoT sistemi

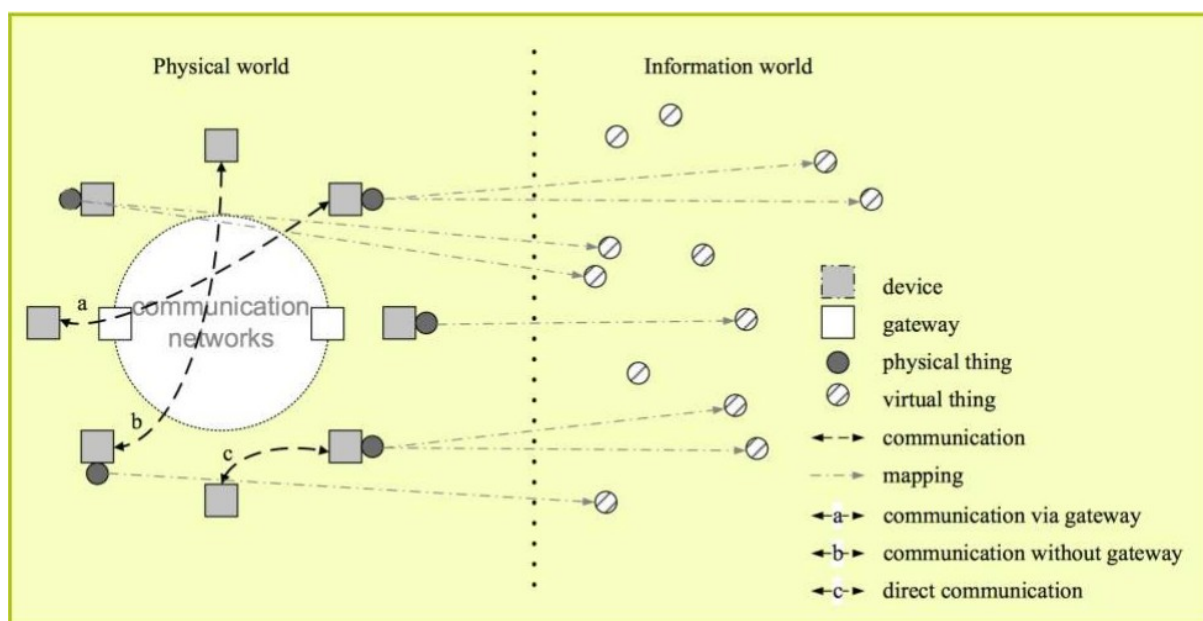
Internet stvari (engl. *Internet of things*) predstavlja mrežu povezanih *stvari* (engl. *things*). Stvari predstavljaju fizičke uređaje ili servise koji se izvršavaju na nekoj mašini. Stvari imaju ugrađene u sebe senzore kojima mogu prikupljati informacije iz svoje spoljašnjosti. Zatim tako prikupljene informacije mogu obrađivati ili slati drugim uređajima.

[1]

Stvari se dele na:

- *Fizičke stvari*
- *Virtuelne stvari*

Fizičke stvari su hardverski uređaji koji imaju senzore, pomoću kojih detektuju stvari iz svoje okoline. Imaju sposobnost *aktuacije* tj. reagovanja na događaje i uticaja na svoju okolinu. Fizičke stvari mogu da detektovane podatke šalju drugim uređajima. Primeri fizičkih uređaja su industrijski roboti, senzori, Raspberry Pi itd. Virtuelne stvari postoje u vidu softvera. Virtuelnim stvarima se može pristupiti, mogu se čuvati, i obrađivati. Primeri virtuelnih stvari su multimedijalni sadržaj i aplikativni softver.^[2] Na slici 1 dat je prikaz jednog IoT sistema i interakcija između fizičkih i virtuelnih stvari. Komunikacija između stvari se dešava preko interneta. Internet stvari obuhvata i vidove komunikacije poput machine-to-machine (srp. Mašina sa mašinom), bežične senzorske mreže, senzorske mreže, 2G/3G/4G/5G, GSM, GPRS, RFID, WI-FI, GPS itd.^[3]



Slika 1: Ilustracija virtuelnih i fizičkih stvari ^[2]

Tehnologije koje učestvuju u internetu stvari se mogu podeliti u tri grupe:

1. Tehnologije koje omogućavaju stvarima da prikupe informacije iz okoline – senzori
2. Tehnologije koje omogućavaju stvarima da obrade informacije iz okoline
3. Tehnologije koje poboljšavaju privatnost i sigurnost

Prve dve kategorije predstavljaju funkcionalne blokove bilo kog sistema za internet stvari. ^[3] Tu spadaju tehnologije koje omogućavaju uređajima da budu opaženi, i kontrolisani putem postojeće mrežne infrastrukture. Težnja je ostvariti što efikasniju, tačniju i nezavisniju komunikaciju između velikog broja različitih uređaja. Poslednjih godina, razvojem veštačke inteligencije, se teži da se čovek eliminiše iz ove komunikacije. Sistemi se projektuju tako da funkcionišu bez ljudskog uplitanja. Treća kategorija tehnologija dobija na važnosti otkako se sve više uređaja integriše u IoT sisteme. Potrebno je osigurati da komunikacija između IoT uređaja bude sigurna i bezbedna. Ne sme biti sigurnosnih propusta koje bi potencijalni napadači mogli da iskoriste. ^{[3][1]}

2.2. Karakteristike IoT sistema

Neke od osnovnih karakteristika IoT sistema su:

- **Povezanost:** Bilo koja stvar se može povezati u sistem bez obzira na njene hardverske mogućnosti ili softversku specifikaciju.
- **Heterogenost:** Uređaji u IoT sistemima su heterogeni. Uređaji imaju različite softverske i hardverske karakteristike. Koriste različite protokole za komunikaciju. Takve uređaje je potrebno objediniti u jedinstven sistem.
- **Dinamične promene:** Stanja uređaja se dinamično menjaju. Uređaji spavaju (engl. sleep) i bude se (engl. wake up), povezuju se na mreže. Menja im se lokacija, brzina (engl. velocity). Broj uređaja se takođe dinamički menja.
- **Ogromna razmera sistema:** Broj uređaja u IoT sistemima je od nekih 1000 uređaja u manjim sistemima, do 100.000 uređaja u velikim industrijskim sistemima. Potrebno je omogućiti da sistem funkcioniše pouzdano. Upravljanje podacima je posebno problematično u ovako velikim sistemima. Potrebno podatke strukturirati na pravi način tako da svaki od uređaja može da ih šalje, prima i obrađuje.
- **Bezbednost:** Sistem mora da se dizajnira tako da bude bezbedan. Mora se obezbediti bezbednost ličnih podataka i bezbednost sistema. Potrebno je primeniti sigurne protokole za komunikaciju, obezbediti svaki endpoint u sistemu i integritet postojećih podataka.
- **Servisi koji su okrenuti stvarima:** IoT sistemi mogu da obezbede servise koji su okrenuti stvarima. Ti servisi obezbeđuju potrebnu privatnost i semantičku konzistentnost između fizičkih i virtuelnih uređaja. Tehnologije koje podržavaju servise okrenute stvarima će morati da se menjaju kako bi obezbedili potrebni stepen privatnosti i pouzdanosti i kako bi se odgovorilo na ograničenja koje stvari imaju. ^[3]

Na slici 2 dat je prikaz svih tehnologija koje učestvuju u IoT sistemima. Karakteristike



Slika 2:

Tehnologije koje učestvuju u internetu stvari

2.3.Arhitektura IoT sistema

Na slici 3 se nalazi prikaz tipične IoT arhitekture, tj elemenata koji su sastavni deo svakog IoT sistema.

Slojevi u strukturi IoT sistema su:

A) Sloj pametnih senzora/uređaja (engl. *smart device/sensor layer*) :

Najniži sloj se sastoji od “pametnih” stvari sa ugrađenim senzorima. Senzori omogućavaju povezivanje “pametnog” uređaja i okoline. Senzor detektuje fizičku pojavu i pretvaraju je u električni signal koji uređaj razume. Senzorom se detektuju i mere određene pojave od interesa u realnom vremenu. Postoje različite vrste senzora u zavisnosti od događaja koji se detektuje, kao i sa različitim stepenima osetljivosti i preciznosti u merenju. Postoje senzori za merenje temperature, vazdušnog pritiska, brzine, geografske širine i dužine itd. Senzori mogu imati određenu količinu memorije kojom mogu čuvati određenih broj najnovijih merenja. Informacije koje su dobijene sa senzora se obično transportuju putem mrežnog sloja do sloja za upravljanje servisima.^{[3][1]}

B) Mrežni sloj (engl. *network layer*)

Senzori proizvode velike količine podataka. Prenos te količine podataka zahteva robustan i pouzdan žičani/bežični mehanizam koji može da podnese po nekoliko terabajta podataka u sekundi. Takođe je potrebno da se obezbedi komunikacija između senzora koji se nalaze na različitim mrežama. Kako bi se ovo ostvarilo mrežni sloj mora da bude u stanju da podrži veliki broj uređaja, da razume protokole koje ti uređaji koriste, i da rutira poruke na pravi način. ^[3]

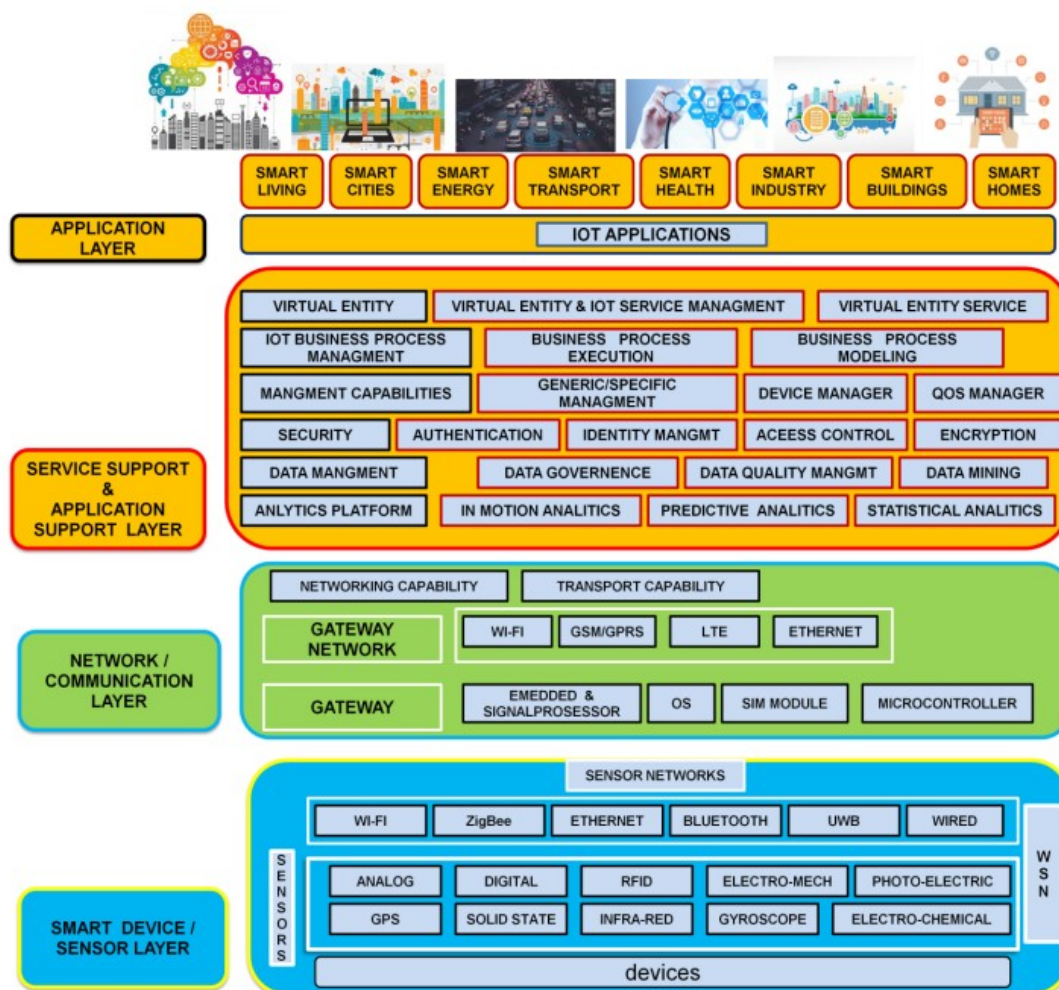
C) Sloj za upravljanje servisima (engl. *Management Service Layer*)

Ovaj sloj je zadužen za obradu informacija, analitiku, sigurnosne kontrole, modelovanje procesa i upravljanje uređajima. Jedan od najvažnijih uloga ovog sloja je pomoć pri donošenju odluka. Ovaj sloj prikuplja informacije, obrađuje ih u skladu sa određenim pravilima i skladišti ih nakon obrade. Takođe analizira pristigle podatke i detektuje trendove u podacima ili podatke/događaje koji su netipični. Na osnovu ovoga sistema ili korisniku se pruža mogućnost da reaguje na određeni način. ^{[3][1]}

Ovaj sloj se takođe bavi i upravljanjem toka podataka. Pod upravljanjem podataka est podrazumeva pristup podacima, integracija i kontrola toka. Sloj za upravljanje servisima može da filtrira podatke koji idu ka aplikativnom sloju i time da smanje broj podataka koji će aplikativni sloj morati da obradi. Takođe ovaj sloj može da vrši prepakivanje podataka i njihovu agregaciju kako bi se samo najvažnije informacije prosleđivale. ^[3]

D) Aplikativni sloj

Na ovom sloju se nalaze aplikacije koje koriste primljene podatke. Neke od poznatijih primena IoT aplikacija su u domenu transporta, agrikulture, pametnih gradova, medicine, prodaje itd. ^{[3][1]}



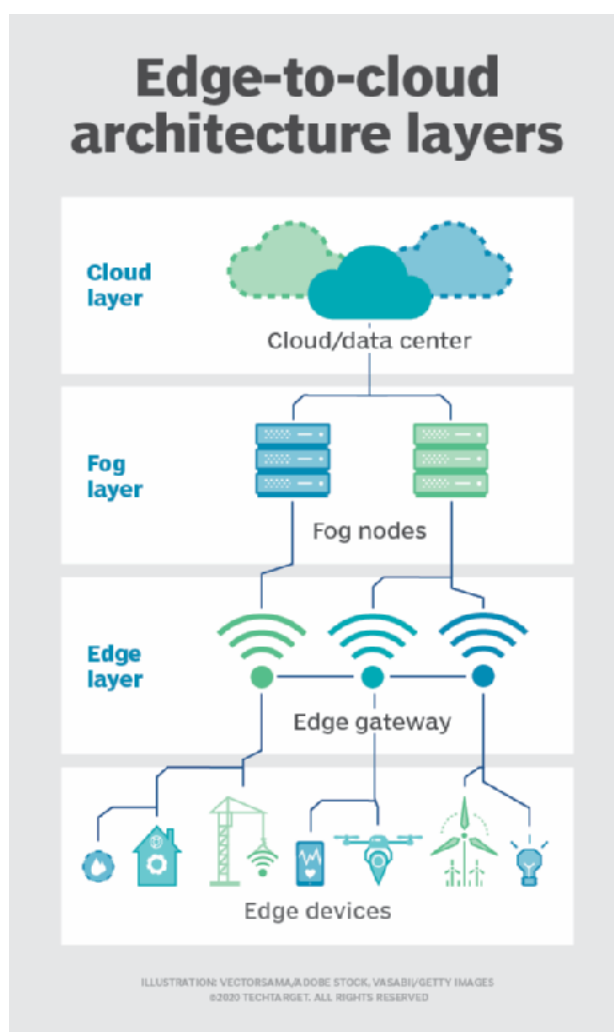
Slika 3: Arhitektura IoT sistema^[3]

2.4. Topološka arhitektura IoT sistema

U današnjem svetu koncept IoT arhitekture je adaptiran tako da odgovara potrebama i mogućnostima savremenih računarskih sistema. Po blizini obrade podataka u odnosu na izvor i po procesnoj moći, savremeni IoT sistemi imaju tri sloja:

- Edge layer (srp. *Ivični sloj*)
- Fog sloj (srp. *Sloj magle*)
- Cloud sloj (srp. *Sloj oblaka*)

Na slici 4 je dat prikaz savremene IoT arhitekture i sve slojeve koje obuhvataju.



Slika 4: Savremena IoT arhitektura^[4]

Kod ovih slojeva je slično to da prihvataju distribuiranu prirodu IoT sistema. Razlika između njih je u tome gdje se nalaze u odnosu na izvor podataka i koliku procesnu moć imaju.

Edge sloj

Kod Edge sloja se računarski resursi koji služe za obradu i skladištenje informacija nalaze na izvoru ili neposredno blizu izvora podataka. U ovom slučaju se latencija u prenosu izmerenih podataka potpuno eliminiše. Ovo znači da se podaci mogu čitati i analizirati u realnom vremenu, tako da se sve promene mogu detektovati odmah i na njih se može adekvatno reagovati. Ono što je problem kod Edge sloja jeste ograničena procesna moć uređaja koji ga čine. Za razliku od uređaja na Fog i Cloud sloju, ovde se obično stavljaju uređaji ograničene procesne moći, što može da oteža obradu podataka. ^{[4][1]}

Jedan od slučajeva korišćenja edge sloja je kod vetroturbina. Blizu vetroturbina se može dodati par servera pomoću kojih bi se skupljali i analizirali podaci koje vetroturbina prikuplja. Još jedan od primera je železnička stanica. Na njoj se mogu postaviti serveri skromnijih performansi, kojima bi se mogli prikupljati i analizirati razne informacije koje dolaze sa raznih senzora postavljenih na stanici. ^[4]

Fog sloj

Problem edge sloja je što su resursi ograničenih performansi ili previše razučeni. Problem cloud sloja je u tome što je vrlo često predaleko od izvora informacija. Kako bi se premostio jaz koji postoji između cloud i edge sloja, osmišljena je ideja Fog sloja. Fog sloj približava moćne računarske resurse, slične kakve postoje na cloud sloju, izvoru podataka.

Tipičan primer korišćenja Fog sloja je u sistemima gde se generiše previše informacija za edge sloj, pa je potrebno obezbediti veću moć obrade podataka. Takvi su slučajevi "pametnih" zgrada, "pametnih" gradova itd. Ako se uzme primer pametnih gradova, tu se pomoću podataka sa senzora može pratiti sistem javnog prevoza, gradske usluge, komunalne usluge, pratiti trendovi u urbanom razvoju grada itd. Jedan edge sloj neće biti dovoljan da podnese toliko opterećenje. Zato se dodaje dovoljan broj Fog čvorova (engl. *node*) koji će pomoći u obradi svih pristiglih podataka. ^[4]

Cloud sloj

Cloud sloj se takođe naziva i računarstvo u oblaku. Cloud sloj je najveći od svih slojeva, ima najveću moć obrade i skalabilan je. Obično je Cloud sloj krajnji sloj jedne IoT arhitekture. Cloud provajderi obično pružaju i određene servise za IoT operacije, i time olakšavaju deployment, monitoring i skaliranje aplikacija i servisa koji se tamo izvršavaju. Cloud je na taj način postao omiljena platforma za IoT servise, jer je potrebna minimalna konfiguracija kako bi se neki servis podigao na server. Takođe ogromna procesna moć koju ima Cloud sloj garantuje da se može opslužiti bilo koja količina podataka. ^[4]

Mana Cloud sloja je ta što je potrebno određeno vreme dok se podaci transportuju od izvora do cloud-a. Ovo može da ima negativnog uticaja, ako je potrebno u realnom vremenu pratiti performanse sistema. ^[4]

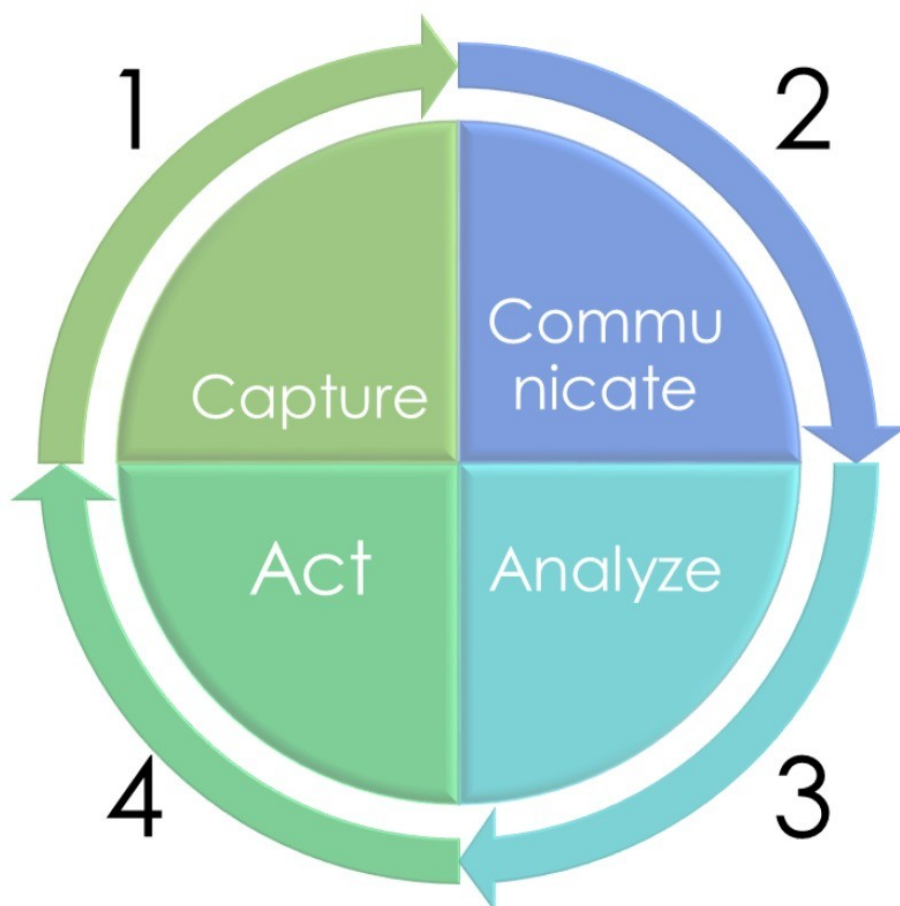
3. ARHITEKTURE IOT PLATFORMI

3.1. Životni ciklus IoT sistema

Ako gledamo strukturu IoT sistema, možemo da primetimo četiri stanja u kojima sistem može da se nađe. Ta četiri stanja su

1. *Capture* (srp. detektuj)
2. *Communicate* (srp. komuniciraj)
3. *Analyze* (srp. analiziraj)
4. *Act* (srp. odradi akciju) ^[5]

Ova četiri stanja predstavljaju ujedno i životni ciklus svakog IoT sistema. Na slici 5 je dat prikaz životnog ciklusa IoT sistema.



Slika 5: Životni ciklus IoT sistema

Capture

Prvo stanje ciklusa je *Capture* stanje. U ovom stanju se nalaze oni uređaji koji imaju senzore u sebi. Oni mogu da detektuju spoljašnje pojave poput temperature, ambijentalnog osvetljenja, vlažnosti vazduha, količinu čestica u prostoru. Ove pojave senzori konvertuju u digitalne signale, koji se zatim prosleđuju dalje kroz sistem. ^[5]

Communicate

Sledeće stanje u ciklusu je *Communicate*. Za ovu fazu postoji preduslov, da postoji veza između uređaja putem neke mrežne tehnologije. U ovoj fazi se ono što je izmereno u sensorima komunicira ka drugim uređajima. Ovo može biti ili ka nekom centralizovanom serveru ili ka nekom konkretnom uređaju. Komunikacija između uređaja u IoT sistemu se vrši preko već postojećih protokola za komunikaciju npr. HTTP, MQTT itd. ^[5]

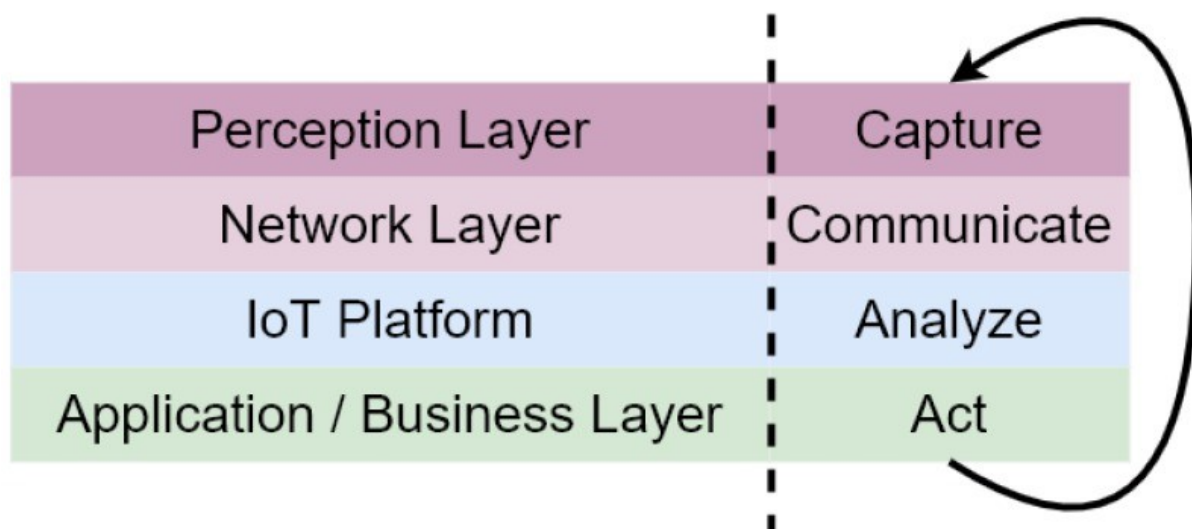
Analysis

Treća faza ciklusa, *Analysis*, je odgovorna za analiziranje podataka koji dolaze sa IoT uređaja. Analize se mogu vršiti na dva načina, u *batch*-u ili u *realnom vremenu*. Batch analiza akumulira pristigle podatke neko vreme, i nakon isteka tog vremena, analizira podatke. Analiza podataka u realnom vremenu podrazumeva da se podaci analiziraju kako pristižu. ^[5]

Act

Act je finalna faza u životnom ciklusu IoT sistema. U ovoj se fazi, na osnovu rezultata dobijenih u prethodnoj fazi, vrši određena akcija. Akcije koje se izvršavaju se dele na dva tipa. Prvi tip akcije je *vizualizacija podataka* za krajnjeg korisnika. Drugi tip akcije se često naziva i *aktucija*. Aktucija predstavlja delovanje nekog uređaja na okolinu u kojoj se nalazi. Npr u sistemu za navodnjavanje, može se instalirati uređaj koji može da pokreće ventil za navodnjavanje. ^[5]

Na slici 6 je dat prikaz kako se faze u životnom ciklusu IoT sistema podudaraju sa slojevitom arhitekturom IoT sistema, datoj u poglavlju 2.3. Sa leve strane grafika su slojevi u tradicionalnoj slojevitoj arhitekturi IoT sistema, a s desne strane su faze u životnom ciklusu IoT sistema.



Perception (srp. Percepcija, očitavanje) sloj je sinonim za Sensor layer u slojevitoj arhitekturi. Njegov zadatak je u *Capture* fazi, tj da očitava podatke sa svakog IoT uređaja (njegovo stanje, temperaturu, vlažnost vazduha, itd). ^[5]

Network (srp. mrežna) sloj šalje podatke sa IoT uređaja ka višim slojevima. Može koristiti 2G, 3G, 4G, ili odskora i 5G tehnologiju, kao i Wifi ili Bluetooth za prenos podataka. Odgovara *Communication* fazi životnog ciklusa. ^[5]

Application/Business sloj odgovara posljednjem, Aplikativnom sloju u slojevitoj arhitekturi. Na ovom sloju se upravlja obrađenim podacima koji dolaze iz prethodnog sloja. Ovde se primenjuju određena biznis pravila na neki scenario, npr „smart“ home, „smart“ health itd. Na ovom sloju se vizualizuju podaci, ili se vrši određena akcija. Zato ovaj sloj odgovara *Act* fazi u životnom ciklusu. ^[5]

IoT platform sloj (srp. Sloj Iot platforme) analizira i perzistira podatke koji dolaze sa IoT uređaja i šalje ih da budu vizualizovani. Sloj IoT platforme odgovara *Analyze* fazi u životnom ciklusu. ^[5] O ovom sloju će biti posebno reči u narednom delu, naročito o njegovoj arhitekturi, funkcionalnostima i karakteristikama.

3.2.IoT platforma

3.3.Monolitna arhitektura

3.4.Mikroservisna arhitektura

3.5.Komparativna analiza arhitektura

4. IMPLEMENTACIJA IOT PLATFORME ZASNOVANE NA ARHITEKTURI MIKROSERVISA

5. PREDLOZI ZA POBOLJŠANJE RAZVIJENOG PROTOTIPA

6. ZAKLJUČAK

Закључак се даје у облику кратких и јасних реченица које представљају осврт на главне доприносе (резултате) у мастер раду. У закључку треба истаћи потенцијалну примену добијених резултата. На крају закључка обично се дају смернице за будући рад у вези теме рада, а имајући у виду намеру да будући читаоци рада добију добру основу за проширивање и унапређење истраживања.

Треба водити рачуна да закључак треба да буде концизан и јасан, без сувишних детаља и понављања претходних реченица. Уобичајено се закључак пише у обиму до 1 странице А4 формата.

LITERATURA

- [1] Vikipedijini korisnici, "Internet stvari," Vikipedija, , [//sr.wikipedia.org/w/index.php?title=%D0%98%D0%BD%D1%82%D0%B5%D1%80%D0%BD%D0%B5%D1%82_%D1%81%D1%82%D0%B2%D0%B0%D1%80%D0%B8&oldid=23043878](https://sr.wikipedia.org/w/index.php?title=%D0%98%D0%BD%D1%82%D0%B5%D1%80%D0%BD%D0%B5%D1%82_%D1%81%D1%82%D0%B2%D0%B0%D1%80%D0%B8&oldid=23043878) (pristupljeno jul 30, 2021).
- [2] Zennaro M., "Introduction to the Internet of Things", ITU, https://www.itu.int/en/ITU-D/Regional-Presence/AsiaPacific/SiteAssets/Pages/Events/2017/Nov_IOT/NBTC%E2%80%93ITU-IoT/Session%201%20IntroIoTMZ-new%20template.pdf (pristupljeno jul 30, 2021)
- [3] Salazar C. et al, "Internet of Things-IOT: Definition, Characteristics, Architecture, Enabling Technologies, Application & Future Challenges" Research Gate, https://www.researchgate.net/publication/330425585_Internet_of_Things-IOT_Definition_Characteristics_Architecture_Enabling_Technologies_Application_Future_Challenges (pristupljeno jul 30, 2021)
- [4] Bigelow S, "What is edge computing? Everything you need to know," <https://searchdatacenter.techtarget.com/definition/edge-computing> (pristupljeno 1.08.2021)
- [5] Trilles, Sergio et al. "An IoT Platform Based on Microservices and Serverless Paradigms for Smart Farming Purposes." Sensors (Basel, Switzerland) vol. 20,8 2418. 24 Apr. 2020, doi:10.3390/s20082418
- [6] A. Trzynadlowski, Control of Induction Motors, Academic Press, New York, 2001.
- [7] Web-site <http://www.tecategroup.com> [Accessed July 2011].
- [8] Web-site <http://www.plagiarism.org/> (pristupljeno 1. oktobra 2014.).
- [9] Uputstvo za učenje: Pisanje eseja, Izbegavanje plagijata, Navodjenje referenci korišćenjem Harvard sistema (sistem autor-datum), Ekonomski fakultet, Beograd, 2009.