



Наш број: 01.сл

Ваш број:

Датум: 2019-01-17

## ИЗВОД ИЗ ЗАПИСНИКА

Наставно-научно веће Факултета техничких наука у Новом Саду, на 10. редовној седници одржаној дана 16.01.2019. године, донело је следећу одлуку:

*-непотребно изостављено-*

### **ТАЧКА 12.1. Верификација нових техничких решења и именовање рецензента**

**Тачка 12.2.1: У циљу верификације новог техничког решења усвајају се рецензенти:**

1. Проф. др Томислав Шекара, Електротехнички факултет у Београду
2. Проф. др Жељко Ђуровић, Електротехнички факултет у Београду

Назив техничког решења:

### **“ПОТ РЕШЕЊЕ ЗА ДЕТЕКЦИЈУ КВАРОВА”**

Аутори техничког решења: Милан Рапаић, Жељко Кановић, Мирна Капетина, Зоран Јеличић.

*-непотребно изостављено-*

Записник водила:

Јасмина Димић, дипл. правник

Тачност података оверава:  
Секретар

Иван Нешковић, дипл. правник

Декан

Проф. др Раде Дорословачки



Трг Доситеја Обрадовића 6, 21000 Нови Сад, Република Србија  
Деканат: 021 6350-413; 021 450-810; Централа: 021 485 2000  
Рачуноводство: 021 458-220; Студентска служба: 021 6350-763  
Телефакс: 021 458-133; e-mail: ftndean@uns.ac.rs

ИНТЕГРИСАНИ  
СИСТЕМ  
МЕНАџМЕНТА  
СЕРТИФИКОВАН ОД:



Наш број: 01.сл

Ваш број:

Датум: 2019-01-31

## ИЗВОД ИЗ ЗАПИСНИКА

Наставно-научно веће Факултета техничких наука у Новом Саду, на 11. редовној седници одржаној дана 30.01.2019. године, донело је следећу одлуку:

*-непотребно изостављено-*

### **ТАЧКА 12.3. Верификација нових техничких решења и именовање рецензента**

Тачка 12.2.2.: На основу позитивног извештаја рецензента верификује се техничко решење (М84) под називом:

Назив техничког решења:

### **“ПОТ РЕШЕЊЕ ЗА ДЕТЕКЦИЈУ КВАРОВА”**

Аутори техничког решења: Милан Рапаић, Жељко Кановић, Мирна Капетина, Зоран Јеличић.

*-непотребно изостављено-*

Записник водила:

Јасмина Димић, дипл. правник

Тачност података оверава:  
Секретар

Иван Нешковић, дипл. правник

Декан

Проф. др Рале Дорословачки

## РЕЦЕНЗИЈА ТЕХНИЧКОГ РЕШЕЊА

### Подаци о техничком решењу:

Назив техничког решења	ИОТ решење за детекцију кварова
Аутори техничког решења	Милан Рапаић, Жељко Кановић, Мирна Капетина, Зоран Јеличић
Реализатори	ФТН
Пројекат на ком је развијено	ТР32018
Област на коју се односи	Детекција и изолација кварова у индустрији
Категорија техничког решења	М84 - битно побољшан постојећи производ или технологија

### Подаци о рецензенту:

Име, презиме и звање	проф. др Жељко Ђуровић
Ужа научна област	Аутоматика
Установа у којој је запослен	Електротехнички факултет, Београд

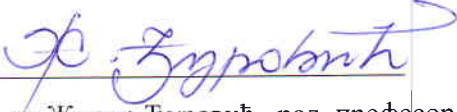
### Стручно мишљење рецензента:

„ ИОТ решење за детекцију кварова “ је техничко решење из области детекције кварова у индустрији базирано на најмодернијим технологијама Интернета ствари. Намењено је праћењу стања индустријских система, уређаја и опреме на стандардизован начин, чиме се омогућава једноставно повезивање и остваривање комуникације између елемената система. Решење предвиђа праћење параметара процеса са различитим радним карактеристикама чиме се покрива широк спектар елемената индустријских система. Концепт решења је веома флексибилан и омогућава адаптацију на различите типове индустријских система. Резултати тестирања потврђују да се ради о квалитетном производу који може значајно да утиче на повећање ефикасности и поузданости производног процеса, с обзиром на то да је стално доступан и да захтева минимално одржавање од стране корисника.

На основу претходно изложеног, предлажем Наставно-научном већу Факултета техничких наука у Новом Саду да пријављено техничко решење „ ИОТ решење за детекцију кварова “ прихвати као:

**Техничко решење- битно побољшан постојећи производ или технологија М84.**

У Београду, 22.01.2019.

  
проф. др Жељко Ђуровић, ред. професор  
Електротехнички факултет у Београду

## РЕЦЕНЗИЈА ТЕХНИЧКОГ РЕШЕЊА

### Подаци о техничком решењу:

Назив техничког решења	IIOT решење за детекцију кварова
Аутори техничког решења	Милан Рапаић, Жељко Кановић, Мирна Капетина, Зоран Јеличић
Реализатори	ФТН
Пројекат на ком је развијено	ТР32018
Област на коју се односи	Детекција и изолација кварова у индустрији
Категорија техничког решења	М84 - битно побољшан постојећи производ или технологија

### Подаци о рецензенту:

Име, презиме и звање	проф. др Томислав Шекара
Ужа научна област	Аутоматика
Установа у којој је запослен	Електротехнички факултет, Београд

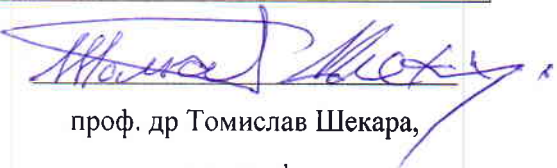
### Стручно мишљење рецензента:

Техничко решење под називом „ IIOT решење за детекцију кварова “ намењено је за рану детекцију и дијагностику кварова у индустријским погонима. Његовом применом унапређује се цео процес одржавања, а поузданост и ефикасност рада постројења подижу се на значајно виши ниво. Решење је базирано на IIOT (Industrial Internet Of Things) технологијама које су веома популарне у тренутним истраживањима и литератури. На овај начин, омогућено је једноставно и стандардизовано повезивање уређаја и опреме, размена података и приступање различитим софтверским сервисима. Примена приказаног техничког решења у индустријским постројењима свакако побољшава експлоатационе параметре и омогућава поузданији рад без нежељених прекида производног процеса, те је његова примена у индустријским постројењима у потпуности оправдана.

На основу претходно изложеног, предлажем Наставно-научном већу Факултета техничких наука у Новом Саду да пријављено техничко решење „ IIOT решење за детекцију кварова “ прихвати као:

**Техничко решење- битно побољшан постојећи производ или технологија М84.**

У Београду, \_\_\_\_\_

  
проф. др Томислав Шекара,  
редовни професор

## **Obrazac za prijavu tehničkog rešenja<sup>1</sup>**

<b>Naziv tehničkog rešenja</b>	<b>IIOT rešenje za detekciju kvarova</b>
<b>Autori tehničkog rešenja</b>	<b>Milan Rapaić, Željko Kanović, Mirna Kapetina, Zoran Jeličić</b>
<b>Kategorija tehničkog rešenja</b>	<b>M84 - bitno poboljšan postojeći proizvod ili tehnologija</b>

### **Za koga je rađeno tehničko rešenje i u okviru kog projekta MPNTR:**

Tehničko rešenje realizovano je u okviru projekta tehnološkog razvoja „Inteligentni nadzorno-upravljački sistem za rano otkrivanje i eliminaciju neželjenih stanja i promena na uređajima, opremi i procesima u procesnoj industriji“ (TR 32018) na Fakultetu tehničkih nauka u Novom Sadu. Rađeno je za potrebe implementacije u industrijskim postrojenjima, u saradnji sa preduzećem INDAS d.o.o. iz Novog Sada.

### **Ko koristi tehničko rešenje:**

Korisnik tehničkog rešenja je preduzeće INDAS d.o.o. – preduzeće za industrijski inženjering, automatiku, mernu i regulacionu tehniku iz Novog Sada.

### **Godina kada je tehničko rešenje urađeno:**

U okviru rada na projektu TR 32018 realizovano je više tehničkih rešenja iz oblasti rane detekcije i izolacije kvarova na opremi u industriji. Predstavljeno tehničko rešenje predstavlja poboljšanje prethodnih varijanti, posebno tehničkih rešenja baziranih na analizi signala vibracija i PBRC algoritmu iz 2012. i 2015. godine. Predstavljeno rešenje realizovano je tokom 2017. i 2018. godine.

### **Ko je prihvatio-primenjuje tehničko rešenje:**

Rešenje je realizovano u okviru projekta TR 32108 za potrebe implementacije u industrijskim sistemima, a primenjuje ga Fakultet tehničkih nauka u Novom Sadu u saradnji sa preduzećem INDAS d.o.o. sa kojim FTN zajednički nastupa kod trećih lica.

### **Kako su rezultati verifikovani (od strane kog tela):**

- Tehničko rešenje realizovano je i ispitano u laboratorijama Katedre za automatsko upravljanje Fakulteta tehničkih nauka u Novom Sadu. Testirano je u praktičnom radu u pogonima preduzeća VictiaOil u Šidu, kao i u TE-TO Zrenjanin, koja funkcioniše kao sastavni deo preduzeća Elektroprivreda

---

<sup>1</sup> U skladu sa odredbama *Pravilnika o postupku, načinu vrednovanja i kvantitavnom iskazivanju naučnoistraživačkih rezultata istraživača*, donetog od strane ministra prosvete, nauke i tehnološkog razvoja na osnovu člana 70. st. 7. i 8. i člana 86. stav 2. Zakona o naučnoistraživačkoj delatnosti („Službeni glasnik RS”, br. 24/2016 i 21/2017).

Srbije. Preduzeće INDAS d.o.o. koristi tehničko rešenje i vrši njegovu implementaciju u pogonima trećih lica.

- Algoritmi primenjeni u rešenju su dobrim delom originalni i verifikovani su objavljivanjem u mnogim naučnim radovima, od kojih navodimo sledeće:
  - M Petković, MR Rapačić, ZD Jeličić, A Pisano, "On-line adaptive clustering for process monitoring and fault detection", *Expert Systems with Applications* 39 (11), 10226-10235
  - Milena Petković, Željko Kanović, Milan Rapačić, „Self- Adaptive Expert System For Process Monitoring And Fault Detection“, In: Martin, D. (Ed.) „ Fault Detection: Methods, Applications and Technology“ (81-104), Nova publishers, 2016
  - Dragan Matić, Željko Kanović, " Vibration Based Broken Bar Detection in Induction Machine for Low Load Conditions", *Advances in electrical and computer engineering*, Vol. 17, No. 1 (2017), 49-54
  - Stefana Jocić, Milan Rapačić, Željko Kanović, Vukan Turkulov, „An example of fault detection system for induction motors based on Internet of Things“, *International Conference on Electrical, Electronic and Computing Engineering IcETRAN*, Palić; 2018.
- Dato je pisano mišljenje dva recenzenta – eksperta iz oblasti tehničkog rešenja:
  - dr Tomislav Šekara, redovni profesor, Elektrotehnički fakultet Beograd
  - dr Željko Durović, redovni profesor, Elektrotehnički fakultet Beograd
- Nastavno-naučno veće Departmana za računarstvo i automatiku Fakulteta tehničkih nauka, na osnovu mišljenja recenzenta i priloženih dokaza, izdalo je Uverenje o priznavanju tehničkog rešenja koje potvrđuje da ono ispunjava sve uslove da bude priznato kao bitno poboljšan postojeći proizvod ili tehnologija kategorije M84

### **Oblast na koju se tehničko rešenje odnosi**

Tehničko rešenje pripada oblasti detekcije i izolacije kvarova u industriji, odnosno softverskoj arhitekturi sistema za detekciju kvarova, koja spada u klasu IIOT (Industrial Internet of Things) rešenja. Realizovan je sistem za praćenje rada industrijskih postrojenja, koji je zamišljen kao integrativni okvir pomoću koga se postojeći industrijski sistemi, uređaji i oprema na jednostavan i standardizovan način povezuju, razmenjuju podatke i pristupaju različitim softverskim servisima, prvenstveno sa servisima za praćenje radnih stanja, servisima za detekciju otkaza i sličnim.

### **Problem koji se tehničkim rešenjem rešava**

Predložena je arhitekturu sistema, koja obezbeđuje prihvatanje i obradu tri kategorije procesnih veličina:

1. Sporo promenjive veličine, koje mogu biti deo sistema za nadzor i čije vreme odabiranja ne prelazi 100 ms.
2. Veličine, koje su deo PLC regulatora ili frekventnih regulatora, koje bi po svom karakteru spadale u veličine srednje brzine odabiranja koja je reda oko 10 ms.
3. Brze ulaze, koji su i najkarakterističniji za obradu kvarova i nosioci su obeležja koja želimo da izdvojimo. Tipični primeri su struje motora i vibracija merenja, koja zahtevaju 10000 uzoraka u sekundi, ili više, za pravilno klasifikovanje.

U okviru pilot IIOT sistema za detekciju kvarova, testirani su algoritmi za detekciju kvarova, vodeći računa o zaštiti podataka i bezbednosnim protokolima, koju su standardni za slične aplikacije (npr. komercijalni distribuirani SCADA sistemi).

### **Stanje rešenosti problema u svetu**

Interesantno je primetiti da je tema IIOT sa primenom u detekciji kvarova najčešće zastupljena u naučno-stručnoj literaturi, sa primerima kroz pilot postrojenja. Prema našim najboljim saznanjima, komercijalna rešenja ovog tipa, nisu široko rasprostranjena u industriju svetu. Na domaćem tržištu, nismo našli slične primere.

## Objašnjenje suštine tehničkog rešenja i detaljan opis sa karakteristikama

### Fault Detective

„Fault Detective“ je sistem za praćenje rada sistema zamišljen kao integrativni okvir pomoću koga se postojeći industrijski sistemi, uređaji i oprema na jednostavan i standardizovan način povezuju, razmenjuju podatke i pristupaju različitim softverskim servisima, prvenstveno sa servisima za praćenje radnih stanja, servisima za detekciju otkaza i sličnim.

Servise pružaju servis provajderi, kompanije koje svojim korisnicima omogućavaju povezivanje na određene servise i pružaju odgovarajuću tehničku i poslovnu podršku za određenu novčanu naknadu. Ponuđeni servisi su raznovrsni kao na primer skladištenje podataka, izrada web aplikacija, aplikacija za mobilne uređaje, IoT (eng. Internet of Things) rešenja, itd. Korisnik može biti svako, od samo jedne osobe do čitave kompanije. Koncept „softver kao servis“ (eng. software-as-a-service, SaaS) datira iz šezdesetih godina prošlog veka, a prvi put se pojavio sa ASP (eng. Applications Service Provider) softver modelom. ASP predstavlja kompaniju koja nudi pristup raznim aplikacijama preko Interneta. Aplikacije se mogu „iznajmiti“ od servis provajdera na određeni vremenski period. Takođe, ASP je veoma koristan za kompanije, jer imaju mogućnost iznajmljivanja potrebnih resursa kako bi zadovoljili svoje tehničke potrebe. Dobra strana ASP jeste da za određenu naknadu klijent ne brine o održavanju aplikacija, infrastrukturi, podacima i dostupnosti sistema. Iako je ASP model taj koji je prvi predstavio SaaS koncept, pretrpeo je nekoliko ograničenja kao što su nemogućnost da razvije visoko interaktivne i potpuno prilagodljive aplikacije. Ipak u poslednje vreme postao je dovoljno dobar da zadobije poverenje nekih velikih kompanija.

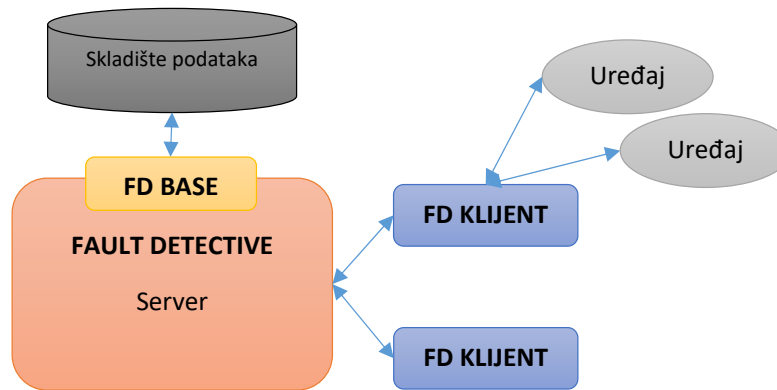
Neke beneficije koje uživa korisnik SaaS-a su:

- Neograničen izbor – internet daje mogućnost korisnicima da iznajmljuju veliki broj aplikacija koje su dostupne na mreži i da ih kombinuju po svojoj želji.
- Instalacija softvera nije potrebna – ASP softver je moguće koristiti odmah, bez dodatnih instalacija i na taj način zaobići vreme potrošeno na instalaciju i podešavanje softvera.
- Nema problema sa kompatibilnošću – korisnik ne mora da vodi računa da li je njegov sistem dovoljno jak da pokrene neki online softver ili da li postoji konflikt sa prethodno instaliranim softverom, jer je sve dostupno na mreži.
- Tehnička podrška nije potrebna - korisnik ne mora da zapošljava administratore da upravljaju kompleksnim softverom i potrebnom opremom. Servis provajderi se brinu o ovakvim problemima i obično su već uključeni u cenu pretplate.
- Servis je stalno dostupan - mnogi servis provajderi pružaju 24/7 dostupnost svojih servisa, jer korisnici zavise od njihovih usluga.

### Arhitektura rešenja

„Fault Detective“ (FD) nije zamišljen kao krut sistem, već kao okvir unutar koga se konkretna implementacija može jednostavno prilagoditi specifičnim potrebama konkretnog problema. Svaki „Fault Detective“ sistem se sastoji iz tri osnovne gradivne jedinice, kao što je prikazano Slika 1.





SLIKA 1. KONCEPCIJSKI PREGLED ARHITEKTURE "FAULT DETECTIVE" SISTEMA.

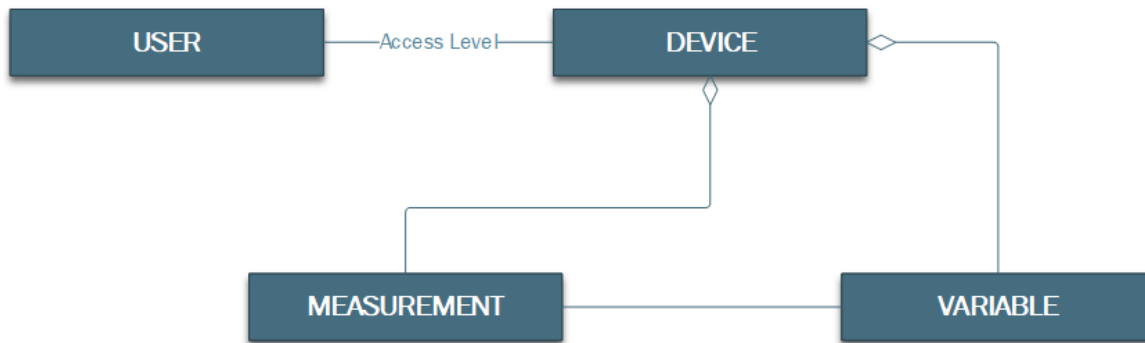
**FD Server** pruža jasno definisane i standardizovane usluge koje se tiču skladištenja podataka o uređajima, veličinama relevantnim za te uređaje i merenjima tih veličina. Funkcionalnost servera implementirana je „baznom komponentom“ – **FD Base**. Bazna komponenta se neposredno vezuje na odgovarajuće **Skladište podataka**, koje može biti relaciona ili nerelaciona baza podataka, datoteka lokalnog fajl-sistema, ali i proizvoljan drugi servis odgovarajuće prirode. Fleksibilnost po pitanju prirode servisa za skladištenje podataka neophodna je, naročito u okviru sistema „Interneta stvari“ (*Internet of Things* – IoT), s obzirom da pruža veliku fleksibilnost po pitanju hardverskog i softverskog okruženja u okviru koga će se implementirati sam server. Sem funkcionalnosti podržane baznom komponentom, svaki server takođe implementira odgovarajući protokol za prenos i serijalizaciju podataka. Na taj način, jednom implementirana i testirana bazna funkcionalnost može biti ponuđena klijentima putem HTTP REST protokola, ili putem Web Socketa protokola, ili na bilo koji drugi relevantan način.

**FD Klijent** je entiteti (fizički uređaj, softverski agent ili drugi servis) koji razmenjuju podatke sa FD Serverom. Klijenti ne poseduju informaciju o tome gde se i kako podaci skladište, niti poseduje druge informacije o detaljima implementacije servera. Na taj način, isti klijent se može koristiti uslugama većeg broja različitih servera, bez reprogramiranja ili rekonfiguracije. Primeri klijenata su pojedinačni uređaji u sistemu „Interneta stvari“, sistemi za agregaciju i vizualizaciju podataka koji jednovremeno mogu pristupati podacima sa velikog broja uređaja, Web-bazirani servisi (Cloud-servisi) za praćenje radnih stanja, detekciju otkaza, optimizaciju rada, i sl. Svaki klijent može biti neposredno povezan sa većim brojem fizičkih uređaja, ali to nije obavezno. Svaki klijent može, putem „Fault Detective“ sistema, pristupati informacijama sa proizvoljnog uređaja za koji ima odgovarajuća ovlašćenja.

Dijagram prikazan na **Error! Reference source not found.** pojednostavljen je u tom smislu što u konkretnim implementacijama može postojati veći broj servera. Serveri se mogu vezivati na isto skladište podataka, ali je dozvoljeno i da pojedini serveri koriste zasebna, lokalna skladišta (bilo zbog bezbednosti, bilo zbog brzine pristupa). Svaki server može podržavati jedan ili više tipova komunikacionih protokola, a svakom serveru jednovremeno može pristupati veliki broj klijenata.

#### Model podataka

„Fault Detective“ sistem poznaje sledeće entitete: **Korisnik** (*User*), **Uređaj** (*Device*), **Promenljiva** (*Variable*) i **Merjenje** (*Measurement*). UML dijagram koji ilustruje logičke veze ovih entiteta prikazan je na slici 2



SLIKA 2. OSNOVNI MODEL PODATAKA PODRŽAN TEKUĆOM IMPLEMENTACIJOM "FAULT DETECTIVE" SISETMA.

**Korisnik** je entitet (osoba, uređaj ili servis) koji se na jedinstven način identifikuje i kome su dozvoljena određena prava pristupa u sistemu. Svaki korisnik je na jedinstven način identifikovan svojim korisničkim imenom (*username*), a svoj identitet potvrđuje putem tajne šifre (*password*). U cilju sigurnosti, same šifre se ne čuvaju neposredno u skladištu podataka, niti se one ikada prenose putem neobezbeđene komunikacione mreže. U skladištu se čuvaju samo obogaćeni heš ključevi šifri, koji se mogu javno deliti bez bezbednosnih rizika (tzv. *salted cache*).

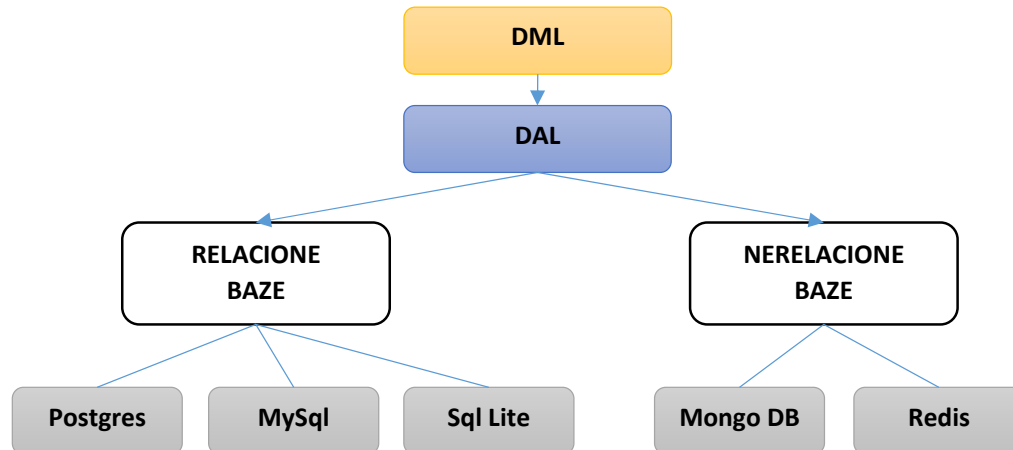
Svatom Korisniku dodeljuju se prava pristupa u odnosu na neke Uređaje. Uređaji su, takođe, na jedinstven način identifikovani imenom (*devicename*). U ovom trenutku, podržani su sledeći nivoi pristupa: **Vlasnik** (*Owner*), **Beležnik** (*Writer*), **Čitač** (*Reader*). Čitač ima pravo da pristupa informacijama o Uređaju koje već postoje u skladištu podataka, ali nema pravo da te informacije menja. Beležnik ima sva prava Čitača, ali on dodatno može unositi nova Merenja u sistem, te registrovati nove Promenljive. Vlasnik nekog Uređaja ima sva prava Čitača, ali on dodatno može menjati informacije o samom uređaju (može ažurirati meta-podatke). U odnosu na sistem u celini, nekim Korisnicima može biti omogućen administratorski nivo pristupa. Administratori imaju pravo dodavanja novih korisnika i menjanja prava pristupa postojećim. Administratori takođe mogu dodavati nove uređaje. Svaki Administrator ima automatski vlasnička prava nad svakim Uređajem registrovanim u sistemu.

**Uređaj** je model fizičkog uređaja, pogona ili dela pogona, senzora, regulatora i sl. U okviru „Fault Detective“ sistema, svakom uređaju pridružen je veći broj **Promenljivih**. Promenljiva je model veličine koju uređaj daje u procesu merenja. Svaka promenljiva je identifikovana imenom (*varname*), koje je jedinstveno na datom uređaju. **Merenja** su uvek pridružena nekom uređaju i vršena su u određenom trenutku (*timestamp*); u okviru jednog merenja prikupljaju se podaci vezani za jednu ili više Promenljivih sa datog uređaja. Za razliku od drugih entiteta, Merenje nije identifikovano imenom već celobrojnim jedinstvenim identifikatorom.

### Implementacija FD Servera

#### Osnovna komponenta: FD Base

Arhitektura FD Base komponente prikazana je slikom 3. Osnovni programski interfejs definisan je komponentom „**Domain Model Layer**“ (**DML**), koja definiše osnovne interfejse i funkcionalnosti komponente. Ova komponenta se sa svoje strane oslanja na „**Data Access Layer**“ (**DAL**) komponentu koja komunicira sa konkretnim skladištem podataka. DAL komponenta obezbeđuje jedinstven interfejs, nezavistan od konkretne implementacije skladišta podataka, te time obezbeđuje proširivost i lakše testiranje, a samim tim i pouzdanost sistema u celini.



SLIKA 3. OPŠTA ARHITEKTURA BAZNE KOMPONENTE FD BASE.

U nastavku, detaljnije specificiramo osnovne interfejse DML komponente, s obzirom da se ti interfejsi neposredno koriste pri implementaciji servera.

<b>IusersDml</b>
<code>add_user(username, password, access_level)</code>
<code>remove_user(username)</code>
<code>get_user(username)</code>
<code>update_password(username, password)</code>
<code>associate_to_device(username, devicename, association_type)</code>
<code>get_associated_devices(username)</code>
<code>get_associations(username, devicename)</code>

<b>IdevicesDml</b>
<code>add_device(devicename)</code>
<code>remove_device(devicename)</code>
<code>get_device(devicename)</code>
<code>get_variables(devicename)</code>
<code>add_measurement(data)</code>

<b>IvariablesDml</b>
<code>add_variable(devicename, varname)</code>
<code>remove_variable(devicename, varname)</code>
<code>get_variable(devicename, varname)</code>
<code>get_data(devicename, varname)</code>

#### *FD HTTP REST Server*

Osnovna varijanta FD Servera koju predlažemo (i koja je implementirana u okviru tekućeg projekta) podrazumeva REST arhitekturu putem HTTP protokola, pri čemu se podaci prenose u JSON formatu. Predloženo rešenje se dakle zasniva na standardnim, zrelim i široko primenjivanim tehnologijama. Posledično, moguće je klijentske aplikacije implementirati nezavisno od serverske, služeći se programskim okruženjima po izboru.

Sledi specifikacija podržanog HTTP interfejsa servera. Svaka podržana metoda opisana je posebnom tabelom u sledećem formatu:

<b>HTTP METHOD</b>	URL
INPUT DATA FORMAT	
OUTPUT DATA FORMAT	
Opis metode.	

Prilikom unosa URL koji odgovara nekoj metodi, elementi putanje uokvireni simbolima „<“ i „>“ odgovaraju promenljivim delovima čija se vrednost mogu menjati od poziva do poziva. Tako recimo <server\_url> označava adresu servera na kome se sistem hostuje.

Prilikom poziva bilo koje metode, korisnik se mora validirati, tako što će proslediti odgovarajuće autentifikacione podatke (username, password) prilikom HTTP poziva. Otuda je prilikom obrade svakog poziva korisnik poznat i verifikovan. Svi podaci, ulazni i povratni, se razmenjuju u JSON formatu. Prilikom specifikacije formata podataka koristićemo zapis

polje1: <tip>  
 polje2: <tip>

pri čemu ćemo vitičaste zagrade karakteristične za JSON format izostavljati. Ukoliko je potrebno specificirati da će metoda slati ili primiti niz promenljivih, koristićemo uglaste zagrade.

Tekuća verzija implementiranog sistema podržava sledeće metode:

<b>GET</b>	<server_url>/users/devices
-	
[ username: <str> devicename: <str> assoc_type: OWNER WRITER READER ]	
Vraća sve uređaje kojima verifikovani korisnik ima pravo pristupa, zajedno sa nivoom pristupa koji mu je dodeljen u odnosu na svaki uređaj.	
<b>PUT</b>	<server_url>/users/devices
username: <str> devicename: <str> assoc_type: OWNER WRITER READER	
-	
Dodeljuje pristup (ili modifikuje postojeće pravo pristupa) verifikovanom korisniku za dati uređaj.	
<b>DELETE</b>	<server_url>/users/devices
username: <str> devicename: <str>	
-	
Ukida svaki pristup datog korisnika datom uređaju.	

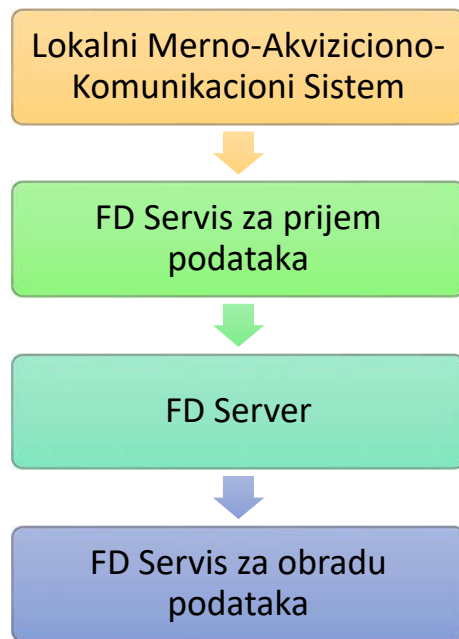
<b>GET</b>	<server_url>/devices/<devicename>
-	
devicename: <str> variables: <str>	
Vraća podatke o zahtevanom uređaju. Metod može pozvati samo korisnik koji ima definisan nivo pristupa ka zahtevanom uređaju (koji je registrovan barem kao Čitač) ili korisnik koji administratorske privilegije.	
<b>PUT</b>	<server_url>/devices/<devicename>
-	
-	

Dodaje novi uređaj u sistem. Tekući korisnik se automatski registruje kao vlasnik sistema.	
<b>DELETE</b>	<server_url>/devices/<devicename>
-	
devicename: <str> variables: <str>	
Briše dati uređaj iz spiska registrovanih uređaja. Metod može pozvati samo korisnik odgovarajućeg nivoa pristupa (Vlasnik uređaja ili korisnik sa administratorskim privilegijama).	

<b>GET</b>	<server_url>/devices/<devicename>/variables
-	
[ devicename: <str> varname: <str> ]	
Vraća podatke o promenljivama registrovanim na datom uređaju.	
<b>PUT</b>	<server_url>/devices/<devicename>/variables/<varname>
devicename: <str> varname: <str>	
-	
Registruje novu promenljivu za dati uređaj.	
<b>DELETE</b>	<server_url>/devices/<devicename>/variables/<varname>
-	
-	
Briše promenljivu iz spiska promenljivih registrovanih na datom uređaju. Metod može pozvati samo korisnik odgovarajućeg nivoa pristupa.	

#### Servisi za detekciju kvarova asinhronih motora kao primer FD klijenta

U toku realizacije projekta, kao primer klijentskih servisa koji bi koristili usluge FD servera, implementiran je servis za detekciju kvarova asinhronih motora. Napominjemo da se radi o test primeru, ne servisu koji treba da bude komercijalno rešenje. Opisaćemo najpre osnovni, motivacioni primer korišćenja (use case) ovakvog servisa, a potom ćemo detaljnije opisati i njegovu implementaciju, te prikazati neke rezultate rada. Tok podataka predviđen osnovnim primerom korišćenja koncizno je prikazan Slika 4.



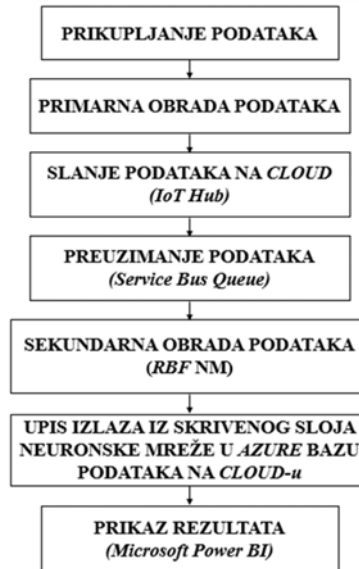
SLIKA 4. TOK PODATAKA U OSNOVNOM PRIMERU KORIŠĆENJA.

Osnovni akteri u sistemu su: same asinhronne mašine čiji se rad prati (odnosno elektronska oprema koja vrši merenje i akviziciju podataka lokalno, te slanje tih podataka ka FD serveru) i inženjeri u pogonu, odnosno osoblje zaduženo za praćenje rada i održavanje opreme u pogonu. Implementirano rešenje pruža punu podršku integrisanju asinhronih motora iz pogona (zajedno sa pripadajućom merno-akviziciono-komunikacionom opremom) u sistem „Interneta Stvari“ (IoT). Podaci se sa lokala prosleđuju najpre ka Cloud-baziranom „Message Queue“ servisu (u konkretnoj implementaciji korišćen je *Azure Service Bus Queue*) kako bi se obezbedila podrška jednovremenom prijemu velike količine podataka bez preopterećenja FD servera. Potom se podaci iz reda čekanja obrađuju prihvatnim servisom koji ih formatira i prosleđuje FD Serveru. Kada se nađu u FD serveru, podatke je moguće koristiti iz sekundarnih servisa za dodatnu obradu. U konkretnom slučaju implementiran je neuralni klasifikator koji procenjuje da li primljeni podaci ukazuju da se asinhrona mašina nalazi u stanju kvara, i u kojoj meri.

#### *Kratak opis softverske arhitekture*

Za izradu softverskog rešenja izabrana je Microsoft-ova platforma za Cloud, Microsoft Azure, programsko okruženje Visual Studio 2015 i Microsoft .NET tehnologija.

Cilj softverskog rešenja je omogućiti slanje sirovih i primarno obrađenih merenja (različitih veličina od interesa) na „Fault Detective“ servis radi čuvanja i eventualne kasnije upotrebe od strane drugih mrežnih servisa. Na taj način, praktično je realizovan decentralizovani sistema koji omogućava jednovremeno, paralelno praćenje radnih stanja asinhronih mašina koje su geografski udaljene, nalaze se u različitim tipovima pogona i rade pod različitim radnim uslovima. Takođe, ovakav servis omogućava da se postupak prikupljanja podataka i primarne obrade potpuno odvoji od finalnog postupka detekcije kvarova samih motora, koji može biti složen i zasnovan na naprednim algoritmima. Šta više, softverske i hardverske komponente, pomoću kojih se vrši akvizicija, mogu biti projektovane nezavisno od tih naknadnih i naprednijih postupaka koji se mogu dodavati i kasnije, ukoliko se sa njima pomaže potreba u eksploataciji rešenja. Shematski prikaz toka podataka klijentske aplikacije prikazan je slikom 5.



SLIKA 5. SHEMATSKI PRIKAZ TOKA PODATAKA KLIJENTSKE APLIKACIJE.

Kao deo Microsoft Azure-a korišćen je IoT Hub, servis koji omogućava dvosmernu komunikaciju između pametnih IoT uređaja i aplikacije. Na portalu Microsoft Azure-a moguće je kreirati IoT Hub, koji će služiti za primanje podataka sa uređaja. Među podešavanjima moguće je odabrati broj povezanih uređaja, broj poslatih poruka, itd.

Da bi uređaj mogao da šalje podatke na IoT Hub potrebno je da bude registrovan, odnosno da poseduje ključ preko koga će pristupiti IoT Hub-u. Kreiranjem jednostavne konzolne aplikacije dobija se ključ za određeni uređaj.

Podaci koji se šalju na Cloud predstavljaju primarno obrađene podatke, odnosno ulaze u neuronsku mrežu, koji će detaljnije biti objašnjeni u nastavku. Slanje podataka vrši se u okviru druge konzolne aplikacije, gde se podaci šalju u tačno određeni „endpoint“, koji ovde predstavlja *Service Bus Queue*. Kreiranje dodatnih „endpoint“-a moguće je preko Microsoft Azure portala.

U konkretnom rešenju, koristeći se terminologijom „Fault Detective“ sistema, Uređaji su pojedinačne asinhronne mašine, dok su promenljive veličine koje se na njima prate:

Opterećenje (*Load*)

Suma bočnih harmonika sa leve strane (*BkBarFeature\_Left*)

Suma bočnih harmonika sa desne strane (*BkBarFeature\_Right*)

Ugaona brzina vratila (*rotSpeed*)

Servis za pohranjivanje podataka registrovan je u FD Server-u kao vlasnik (Owner) svakog pojedinačnog uređaja čije podatke prikuplja i šalje. Sa druge strane, servisu za obradu podataka (i detekciju kvarova) dodeljena su prava čitanja (Reader) u FD Server sistemu.

**Rezultati**

*Ekperimentalni uzorak*

Korišteni su podaci iz TE - TO Zrenjanin u cilju analize rada viskonaponskih motora napojnih pumpi dva kotla, K1 i K2. Eksperimentalni uzorak na kome su vršena istraživanja se sastoji od dva identična visokonaponska elektromotora. Podaci o elektromotorima su sledeći:

Nazivna snaga: 3150 kW;

Nazivni napon: 6000 V;

Nazivna struja: 373 A;

Nazivna frekvencija: 50 Hz.

*Prikupljanje podataka*

Na oba motora vršeno je merenje vibracija u toku rada. Snimljeno je po 10 signala u stacionarnom režimu rada, sa promenljivim opterećenjem i brzinom obrtanja elektromotora. Na elektromotoru 2 napojne pumpe 2 kotla K1 snimani su signali sa opterećenjima od 80 i 100 t/h, dok su na elektromotoru 4 napojne pumpe 2 kotla K2 snimani signali sa opterećenjima od 60, 80 i 100 t/h (nominalno opterećenje 300 t/h). Rezultati obavljenih merenja i sprovedena analiza rezultata upućuju na to da elektromotor 4 ima otkaz tipa slomljene šipke na rotoru, dok kod elektromotora 2 nije detektovan kvar ovog tipa. U Tabela 1 prikazani su parametri opterećenja i brzine obrtanja rotora.

**TABELA 1. PODACI O ELEKTROMOTORIMA KORIŠTENIM PRILIKOM TESTIRANJA SISTEMA.**

Motor	Opterećenje [t/h]		
	60	80	100
	Brzina obrtanja rotora [Hz]		
Elektromotor 2 napojne pumpe 2 kotla K1	-	49,95	49,93
Elektromotor 4 napojne pumpe 2 kotla K2	49,95	49,94	49,92

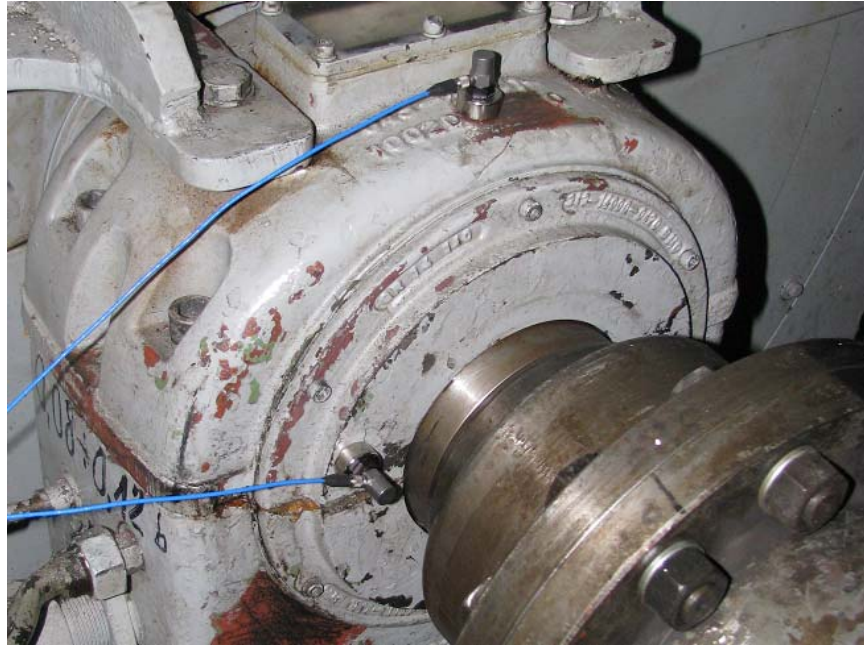
Za merenje vibracija korišćena su dva akcelometra, postavljena tako da prikupljaju podatke radijalnih vibracija u vertikalnoj ravni i askijalnih vibracija u horizontalnoj ravni. Pozicija senzora se može videti na slici 4.1. Signali su snimani sa frekvencijom odabiranja 10 kHz, a svaki snimak sadrži 100000 odbiraka. Merenje je izvršeno sledećom opremom:

Akvizionni sistem NI USB-9162 sa mernom karticom NI 9234 proizvođača *National Instruments*,

Akcelometri osetljivosti 100 mV/g proizvođača *PCB* (2 komada),

Notebook računar *Dell*.





SLIKA 6. NAČIN MONTAŽE SENZORA VIBRACIJA.

#### *Obrada podataka i priprema za klasifikaciju*

Obzirom na unapred poznatu vrstu kvara elektromotora, za računanje karakterističnih obeležja korišćen je izraz za izračunavanje karakteristične frekvencije oštećenja šipke rotora. Kako bi se izračunala karakteristična obeležja u frekvencijskom domenu potrebno je na signal iz vremenskog domena primeniti Furijeovu transformaciju (*FFT*), da bi se dobio spektar signala vibracija. Ovo obeležje je izračunato tako što je izvršeno sumiranje vrednosti amplituda svih odbiraka spektra signala koji se nalaze u opsegu  $f_c \pm \Delta F$ , gde je sa  $f_c$  označen levi i desni bočni opseg karakterističan za oštećenje šipke rotora ( $k = 1$ ), dok  $\Delta F$  predstavlja količnik frekvencije odabiranja i broja odbiraka u signalu.

#### *Klasifikacija stanja elektromotora*

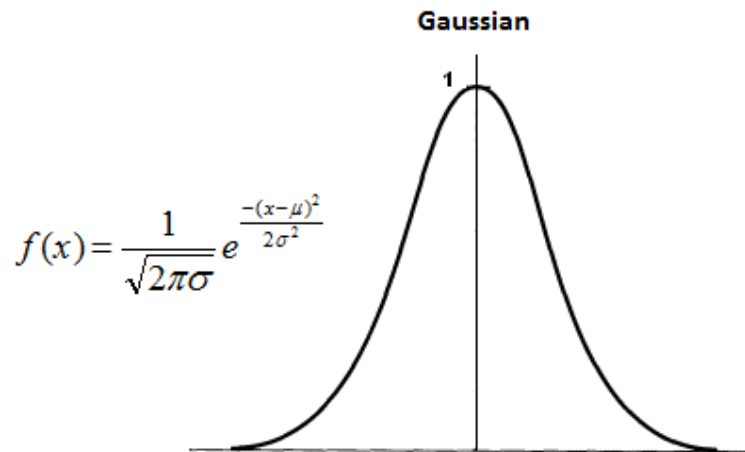
Za klasifikaciju stanja motora korišćena je RBF neuronska mreža (eng. *Radial Basis Function Neural Network*).

RBF neuronske mreže poseduju tri sloja:

Ulazni sloj – za svaku ulaznu promenljivu postoji po 1 neuron u ulaznom sloju. U slučaju promenljivih sa konačnim brojem mogućih vrednosti koristi se  $N-1$  neurona, gde  $N$  predstavlja broj kategorija.

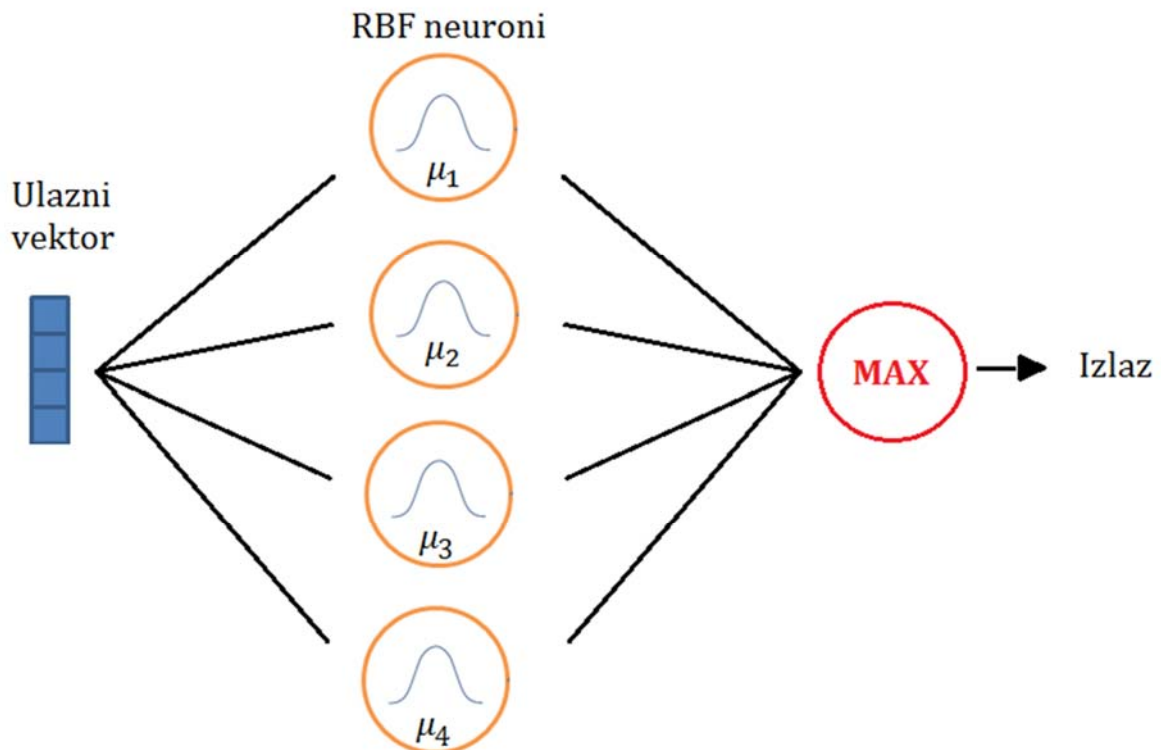
Skriveni sloj – u ovom sloju nalazi se promenljiv broj neurona. Svaki neuron sastoji se od radijalno koncentrisane funkcijesa onoliko dimenzija koliko ima predskazanih promenljivih. Radijus RBF funkcije može biti različit za svaku dimenziju. Centri i radijusi određuju se tokom treniranja neuronske mreže. Za funkciju se obično bira Gausova funkcija prikazana na slici 4.2. Za vektor  $x$  iz ulaznog sloja, neuron u skrivenom sloju računa Euklidovo rastojanje test slučaja od centra neurona, pa se na izračunato rastojanje primenjuje RBF funkcija. Izračunata vrednost prenosi se do izlaznog sloja.

Izlazni sloj – vrednost koja je izračunata u skrivenom sloju množi se sa težinom koja je povezana sa neuronom, pa se potom sve te vrednosti sabiraju i predstavljaju izlaz iz neuronske mreže. Za probleme klasifikacije postoji samo jedan izlaz iz mreže za svaku kategoriju. Vrednost izlaza kategorije predstavlja verovatnoću da je slučaj koji se procenjuje je iz te kategorije.



SLIKA 7. GAUSOVA FUNKCIJA

Struktura neuronske mreže upotrebijene za klasifikaciju stanja motora prikazana je na slici ispod.



SLIKA 8. STRUKTURA RBF MREŽE

Mreža se sastoji od tri sloja: ulaznog, skrivenog i izlaznog sloja. Ulaz u neuronsku mrežu je vektor od 4 elementa. Prvi element predstavlja opterećenje motora (može biti 60 ili 80 t/h, dok je nominalno opterećenje 300 t/h), skaliran od 0 do 1. Drugi i treći element predstavljaju količnik amplitude spektra signala vibracija na frekvenciji bočnih opsega karakterističnih za kvar tipa slomljene šipke rotora i amplitude spektra signala vibracija na frekvenciji obrtanja rotora. Poslednji element je frekvencija obrtanja rotora asinhronog motora.

U skrivenom sloju nalaze se 4 neurona redom za sva 4 karakteristična slučaja koja se javljaju:

Ispravan elektromotor sa opterećenjem od 100 t/h;

Neispravan elektromotor sa opterećenjem od 80 t/h;

Neispravan elektromotor sa opterećenjem od 100 t/h;

Ispravan elektromotor sa opterećenjem od 80 t/h.

Svaki od neurona sadrži centroid  $\mu_j$ , koji predstavlja vektor, čija je struktura ista kao i kod ulaznog vektora. Izlaz iz neurona definiše se izrazom:

$$\phi_j = e^{-\frac{\|x-\mu_j\|^2}{2\sigma_j^2}},$$

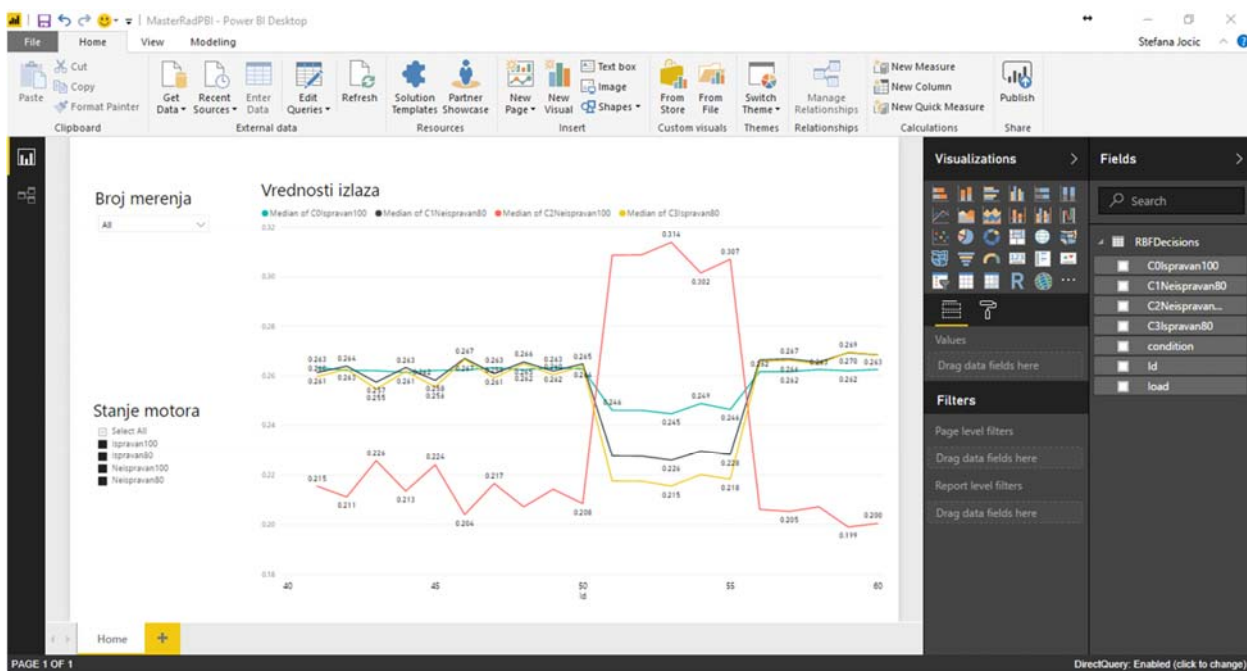
gde  $x$  predstavlja ulazni vektor, a  $\sigma$  promenljiv parametar koji u ovom slučaju uzima vrednost 1. Kada se izračunaju izlazi iz neurona, svi izlazi čija je vrednost manja od praga  $\varepsilon = 0.01$  postavljaju se na 0. Nakon toga potrebno je podeliti  $\phi_j$  sa sumom svih vrednosti za  $\phi_j$ , ukoliko je suma pozitivna. Na kraju, za svaki snimljeni signal od dobijene 4 vrednosti bira se maksimalna vrednost, koja predstavlja izlaz iz neuronske mreže. Na osnovu indeksa tog maksimalnog elementa motor se može klasifikovati kao ispravan ili neispravan i moguće je utvrditi opterećenje. U tabeli 2 prikazan je broj tačnih klasifikacija stanja elektromotora.

**TABELA 2. BROJ TAČNIH KLASIFIKACIJA STANJA ELEKTROMOTORA**

Stanje motora	Opterećenje motora [t/h]	Broj signala	Broj tačnih klasifikacija
Ispravan	80	5	2
Ispravan	100	5	5
Neispravan	80	5	5
Neispravan	100	5	5

Za svaki primer motora upotrebljeno je po 5 signala. Dobijeni rezultati su očekivani, jer prilikom manjeg opterećenja motora teško je detektovati kvar tipa slomljene šipke rotora. Zbog toga su od ukupno 5 signala snimljenih na ispravnom motoru opterećenim sa 80 t/h, samo su 2 klasifikovana kao takva, a ostali kao neispravni.

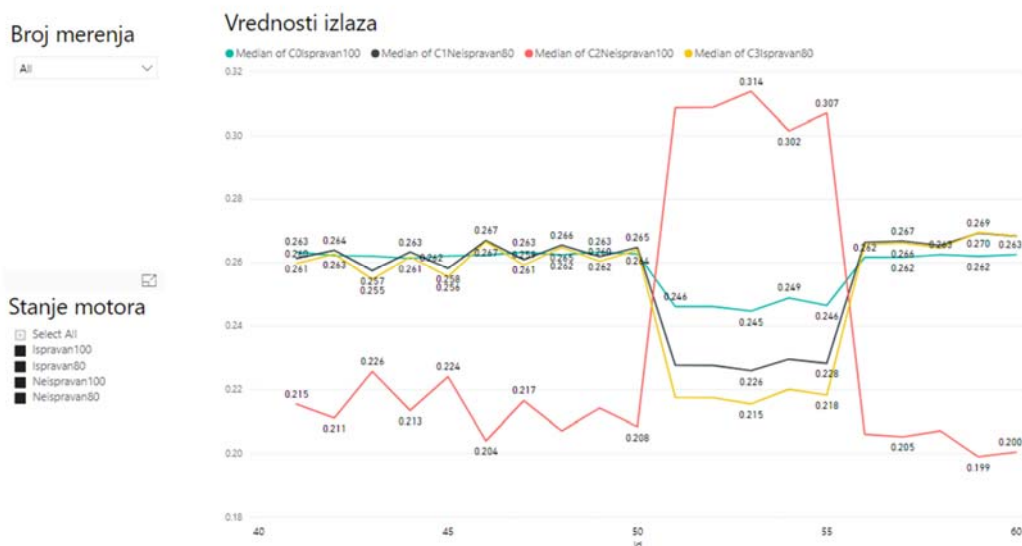
Za prezentaciju podataka iskorišćen je softverski paket za analizu podataka *Microsoft Power BI*. Ovaj alat omogućava uvoz podataka iz različitih izvora, kao što su baze podataka, *SQL* serveri, *Excel* tabele, itd. Odabirom različitih vizualizacija moguće transformisati sirove podatke i na taj način napraviti prezentacije, dokumentaciju i izveštaje. *Power BI* sastoji se od nekoliko komponenti, kao što su *Power BI Desktop*, *Power BI Service*, *Power BI Mobile Apps*, itd. *Power BI Service* predstavlja online servis gde je moguće kreiranje prezentacija i izveštaja, objaviti ih online i podesiti njihovo osvežavanje, tako da korisnici uvek imaju pristup najsvježijim podacima. *Power BI Desktop* predstavlja desktop aplikaciju za *Windows* računare. Na slici 9. može se videti izgled prozora *Power BI Desktop* aplikacije.



SLIKA 9. IZGLED ALATA MICROSOFT POWER BI

Na slici ispod može se videti grafik, koji predstavlja izlaze iz skrivenog sloja neuronske mreže korišćene za klasifikaciju motora. Moguće je izabrati pojedinačno merenje ili sva merenja odjednom i motore klasifikovane kao ispravne ili neispravne. Za svako merenje postoje 4 tačke, a ona koja ima maksimalnu vrednost pokazuje stanje motora i njegovo opterećenje, što se može videti na slici 9. Na slici 10. prikazani su izlazi iz skrivenog sloja neuronske mreže za motore klasifikovane kao ispravne, dok se grafik sa neispravnim motorima može videti na slici 11.

Na svakih sat vremena šalje se upit u Azure bazu podataka, odakle su preuzeti podaci za kreiranje prezentacije i ukoliko ima nekih promena grafik se osvežava, tako da korisnik može da prati stanja elektromotora.



SLIKA 10. PRIKAZ IZLAZA IZ SKRIVENOG SLOJA NEURONSKE MREŽE ZA SVA MERENJA

### Broj merenja

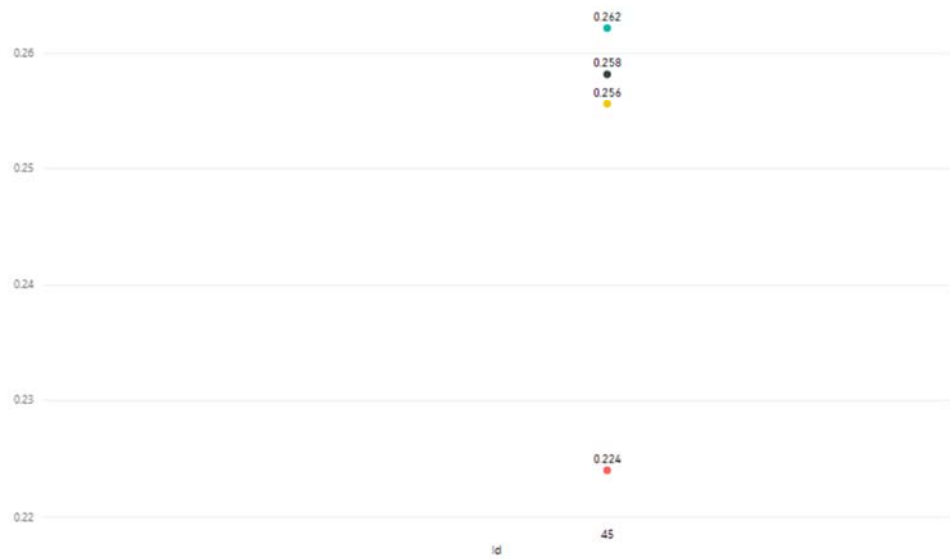
- 45
- Select All
  - 41
  - 42
  - 43
  - 44
  - 45
  - 46
  - 47
  - 48
  - ^

### Stanje motora

- Select All
- Ispravan100

### Vrednosti izlaza

● Median of C0Ispravan100 ● Median of C1Neispravan80 ● Median of C2Neispravan100 ● Median of C3Ispravan80



SLIKA 11. PRIKAZ JEDNOG MERENJA NA GRAFIKU

## Mogućnost primene

Kao što je rečeno „Fault Detective“ je integrativni okvir pomoću koga se postojeći industrijski sistemi, uređaji i oprema na jednostavan i standardizovan način povezuju, razmenjuju podatke i pristupaju različitim softverskim servisima, prvenstveno sa servisima za praćenje radnih stanja, servisima za detekciju otkaza i sličnim. Namera nam je da u punoj implementaciji, ovo rešenje bude integrativni deo sistema za nadzor ili zaseban servis za centralizovanu detekciju kvarova u više distribuiranih pogona.