

УНИВЕРЗИТЕТ У НОВОМ САДУ

**ФАКУЛТЕТ ТЕХНИЧКИХ НАУКА У**

**НОВОМ САДУ**

**SISTEM UPRAVLJANJA ISPADIMA U ELEKTRODISTRIBUTIVNOJ MREŽI**

**(OUTAGE MANAGEMENT SYSTEM)**

**Slobodan Brdar**

**Nina Grbić**

**Andrej Ivičiak**

**Aleksa Ignjatović**

**Teodora Kadić**

**Mladen Milošević**

**Dimitrije Mitić**

**Nemanja Simić**

Novi Sad, 2020.

# Sadržaj

[Sadržaj 1](#_Toc39673807)

[Spisak korišćenih skraćenica 3](#_Toc39673808)

[Uvod 4](#_Toc39673809)

[OMS 5](#_Toc39673810)

[SCADA - UVOD 5](#_Toc39673811)

[Mreža pod SCADA softverom 6](#_Toc39673812)

[Mreža bez nadzora 6](#_Toc39673813)

[Opis rešavanog problema 7](#_Toc39673814)

[Standardi i modeliranje elektroenergetskog sistema 7](#_Toc39673815)

[Simulacija elektroenergetskog sistema 7](#_Toc39673816)

[Održavanje i kontrola kvaliteta 8](#_Toc39673817)

[Napredni računarski sistemi 8](#_Toc39673818)

[Sigurnost i bezbednost 8](#_Toc39673819)

[Model 9](#_Toc39673820)

[Arhitektura 11](#_Toc39673821)

[Transaction Manager 12](#_Toc39673822)

[NMS 12](#_Toc39673823)

[Korisnički Interfejs (UI) 13](#_Toc39673824)

[Fizički sistem (Simulator) 14](#_Toc39673825)

# Spisak korišćenih skraćenica

|  |  |
| --- | --- |
| *Skraćenica* | *Značenje skraćenice* |
| *OMS* | *Outage Management System* |
| *XML* | *Extensible Markup Language* |
| *CIM* | *Common Information Model* |
| *NMS* | *Network Model Service* |
| *SCADA* | *Supervisory Control And Data Acquisition* |
| *UML* | *Unified Modeling Language* |
| *XMI* | *XML Metadata Interchange* |
| *RDF* | *Resource Description Framework* |
| *RDFS* | *Resource Description Framework Schema* |
| *DLL* | *Dynamic-link library* |
| *PLC* | *Programmable Logical Controller* |
| *MTU* | *Master Terminal Unit* |
| *RTU* | *Remote Terminal Unit* |
| *HMI* | *Human Machine Interface* |
| *TM* | *Transaction Manager* |
| *DMS* | *Distribution Management System* |

# Uvod

Elektrodistributivna mreža, kao jedna od najvažnijih infrastruktura jedne države, regiona i sveta, podrazumeva što veću pouzdanost i uvek postoji težnja da se ta pouzdanost što više poveća. Zbog nesavršenosti elektrodistributivne mreže dešavaju se ispadi (***outage***) tokom redovnog rada sistema. Ispadi podrazumevaju prestanak rada sistema ili dela sistema i time prekid protoka električne energije uzrokovanim kvarom nekih od elemenata mreže. Kvar može biti uzrokovan prirodnom nepogodom (najčešće) ili ljudskim faktorom. Postoji nekoliko tipova ispada:

* Prolazni kvar (***a transient fault***) je prekid protoka električne energije obično uzrokovan kvarom na električnim vodovima. Protok se ponovo uspostavlja nakon što se kvar ukloni.
* Pad napona (***a brownout***), namerni ili slučajni, može prouzrokovati blagi ispad u vidu loših performansi opreme ili njihovog nepravilnog funkcionisanja.
* Potpuni gubitak energije (***a blackout***) u određenoj oblasti je najkobniji od svih tipova ispada. Kada dođe do ispada ovog tipa oporavak je teško brzo izvesti. Ispad može trajati od par minuta do par nedelja u zavisnosti od njegove prirode i konfiguracije elektrodistributivne mreže.

Rotirajuće preusmeravanje potrošnje (***Rolling blackout***) je planirano isključenje napajanja gde se protok energije prekida na određeni period vremena za svaki deo distributivnog regiona naizmenično. Rolling blackout se dešava kada potražnja za električnom energijom prevaziđe proizvodnju, i na taj način dozvoljava nekim potrošačima da budu napajani energijom na račun ostalih potrošača koji neće biti napajani. Često se dešava u zemljama u razvoju i može biti planiran unapred, ali može se i desiti bez upozorenja.

Kod elektrodistributivnih mreža, proizvodnja energije i njena potrošnja moraju biti približno jednake svakog trenutka kako bi se izbeglo preopterećenje na komponentama mreže, koje može da ih trajno ošteti. Zaštitni releji (***protective relays***) i osigurači (***fuses***) se koriste za automatsko detektovanje preopterećenja i za prekidanje strujnog kola kada se pojavi opasnost od štete.

# OMS

Sistem upravljanja ispadima u elektrodistributivnoj mreži (***Outage Management System***) je softver koji će razrešavati ispade i kvarove koji se pojave u elektrodistributivnoj mreži. Projekat razmatra dva različita slučaja mreže (koja u realnosti postoje i zajedno čine celu mrežu):

1. Mreža koja je pod nadzorom SCADA softvera
2. Mreža koja nije pod nadzorom

## SCADA - UVOD

Kako bi objasnili prvi slučaj mreže, mora se prvo objasniti SCADA. Softver za nadzor, kontrolu i pribavljanje podataka (***Supervisory Control And Data Aqusition***) se koristi kod sistema koji imaju automatizovane procese. SCADA mreža se sastoji od jednog ili više MTU (***Master Terminal Unit***) koji su zapravo računarske stanice opremljene odgovarajućim [softverom](https://sr.wikipedia.org/wiki/%D0%A1%D0%BE%D1%84%D1%82%D0%B2%D0%B5%D1%80) i [operativnim sistemom](https://sr.wikipedia.org/wiki/%D0%9E%D0%BF%D0%B5%D1%80%D0%B0%D1%82%D0%B8%D0%B2%D0%BD%D0%B8_%D1%81%D0%B8%D1%81%D1%82%D0%B5%D0%BC). Ove stanice operatori koriste za praćenje i kontrolu jedne ili više RTU (***Remote Terminal Unit***). RTU je takođe računarski uređaj koji je obično namenjen za rad u industrijskim uslovima ili čak u ekstremnim kao što je npr. okruženje u vasioni. Njegov zadatak je da prikuplja [informacije](https://sr.wikipedia.org/wiki/%D0%98%D0%BD%D1%84%D0%BE%D1%80%D0%BC%D0%B0%D1%86%D0%B8%D1%98%D0%B0) sa raznih digitalnih i analognih [senzora](https://sr.wikipedia.org/wiki/Senzor) i da prosleđuje komande uređajima koji na neki način menjaju stanje upravljanog sistema. Često se za njihovu realizaciju koriste razni tipovi [PLC](https://sr.wikipedia.org/wiki/Programabilni_logi%C4%8Dki_kontroler) (***Programmable Logic Controller***).

Za prenos podataka između MTU i RTU se koristi neki od standardnih ili specijalizovanih protokola kao što su Ethernet, Modbus... Optički kablovi su idealni za realizaciju SCADA mreže jer nude potpunu zaštitu od emisije raznih zračenja koja bi mogla da dovede do prenosa pogrešnih informacija. U idealnom SCADA sistemu svi elementi su redundantni što dostupnost i pouzdanost sistema podiže na visok nivo.

Uloga komponenete HMI (***Human Machine Interface***) je da prezentuje informacije korisnicima SCADA pa je to najčešće računar sa softverom koji na jasan način prikazuje sve informacije o sistemu i realizuje njegovu upravljivost.

Ovi sistemi postoje u raznim oblicima od 60-tih godina, a od 90-tih godina 20. veka doživljavaju veliku ekspanziju pojavom sve bržih i efikasnijih računarskih i mikrokontrolerskih uređaja. Mogu se upotrebiti od npr. jednostavnog praćenja temperature, vlažnosti vazduha, pritiska, do npr. veoma kompleksnog praćenja i kontrole proizvodnih procesa fabrike ili saobraćaja na železnici.

## Mreža pod SCADA softverom

Sada, kada je SCADA softver definisan, biće ukratko objašnjen deo mreže koju pokriva SCADA. Objašnjenje sledi kroz primer rešavanja jednog ispada:

1. Dolazi do ispada u nekom delu mreže.
2. Pokreće se algoritam koji pronalazi gde se tačno kvar desio i automatski ga izoluje.
3. Šalje se ekipa koja treba da popravi izolovani kvar.
4. Ekipa popravlja kvar i deo mreže koji je bio u ispadu se vraća u rad.

## Mreža bez nadzora

Mreža koja nema nikakav nadzor nema mogućnost automatskog rešavanja ispada. Jedan od scenarija rešavanja ispada u ovakvoj mreži je sledeći:

1. Dolazi do ispada u nekom delu mreže.
2. Ljudi koji primete nedostatak električne energije (na krajnjoj potrošnji) šalju elektronske poruke (***e-mail***) elektrodistributivnoj službi.
3. Kada se dostigne određeni broj poziva (poruka) pokreće se algoritam za pronalaženje kvara (***tracing*** ***algorithm***), odnosno automatskog prekidača (***breaker***) koji je povezan na električni vod (***AC line***) na kom se desio ispad.
4. Nakon što se ustanovi koji je automatski prekidač u pitanju, šalje se ekipa koja treba da locira konkretno mesto kvara u tom odeljku mreže.
5. Ekipa pronalazi kvar i otklanja ga, i zatim taj deo mreže koji je bio u ispadu vraća se u rad.

# Opis rešavanog problema

Nakon objašnjenih osnovnih koncepata i pojmova rešavanog problema, ovaj deo teksta će se baviti opisom samog rešavanog problema. Razmatrani problem je podeljen u pet celina:

1. Standardi i modeliranje elektroenergetskog sistema
2. Simulacija elektroenergetskog sistema (SCADA)
3. Održavanje i kontrola kvaliteta (Cloud)
4. Napredni računarski sistemi (Upravljanje i optimizacija)
5. Sigurnost i bezbednost (DMS)

Svaki od nabrojanih delova je, zapravo, logička celina u sklopu projekta i biće predstavljena kratka analiza svakog od njih u nastavku. Osnovna ideja projekta je napraviti sistem za detekciju, analizu i upravljanje svim planskim isključenjima, kvarovima i događajima koji rezultuju prekidima napajanja u elektroenergetskom sistemu.

## Standardi i modeliranje elektroenergetskog sistema

Prva celina se odnosi na rad sa modelom. Ovaj deo je zahtevao kreiranje CIM profila koji zadovoljava potrebe sistema. Takođe, zahtevao je i unos podataka na osnovu profila i implementaciju distribuirane transakcije. Ova celina uključuje i postojanje višekorisničkog klijentskog rada. Na kraju, bilo je potrebno obezbediti ***shallow copy*** izmenu modela u toku transakcije.

## Simulacija elektroenergetskog sistema

Druga celina predstavlja SCADA softver. Podrazumeva se realizacija analognih i digitalnih ulazanih i izlaznih tačaka korišćenjem Modbus protokola. Potrebno je bilo obezbediti online informacije o trenutnom stanju sistema sa udaljenih (***remote***) uređaja. Udaljeni uređaji su simulirani preko stranog (***third party***) simulatora, gde je, takođe, omogućeno izdavanje komandi za izlazne tačke. Definisanje radnog i abnormalnog stanja i implementacija funkconalnosti alarma (***alarming***).

## Održavanje i kontrola kvaliteta

Treća celina obuhvata cloud tehnologiju i prebacivanje monolitičke aplikacije u mikroservisnu oslanjajući se na mikroservisnu arhitekturu, gde će biti pokazana razlika (prednosti) i mane između korišćenja mikroservisne arhitekture i monolitičke arhitekture. Baza korišćena u monolitičkoj aplikaciji (SQL baza) će biti zamenjena cloud baziranom bazom (Azure). Celokupna aplikacija će počivati na Azure tehnologiji gde će se koristiti svi potrebni dodaci i pogodnosti koja ta tehnologija pruža.

## Napredni računarski sistemi

Četvrta celina se bavi upravljanjem i optimizacijom elektrodistributivnog sistema. Primer upravljanja mrežom je da prilikom ispada nekog segmenta fidera (***feeder***) reaguje ***recloser*** (kada primeti da je sa jedne strane prisutan napon, a sa druge ne) koji ponovo spaja strujno kolo. Prisutnost recloser-a je i u mreži koja je pod nadzorom SCADA softvera i u mreži koja nije.

## Sigurnost i bezbednost

Peta i poslednja celina je u suštini srž aplikacije – DMS. Ovaj deo je realizacija algoritama o razrešavanju ispada i kvarova, u zavisnosti od tipa mreže (SCADA ili bez nje). Postojaće tri nivoa na kojima se kvar može lokalizovati:

1. Osigurač na niskonaponskim izvodima
2. Prekidač na niskonaponskoj strani transformatora
3. Prekidač na srednjenaponskoj strani transformatora

Nakon lokalizacije kvara izlazi ekipa koja treba da otkloni kvar i osposobi taj deo mreže koji je bio u ispadu (kao što je bilo objašnjeno u ranijem tekstu).

# Model

0..1

Core::IdentifiedObject

Core::PowerSystemResource

Core::Equipment

Core::ConductingEquipment

Wires::Conductor

Wires::ACLineSegment

Wires::Switch

Core::BasicIntervalSchedule

Core::IrregularIntervalSchedule

Outage::OutageSchedule

Outage::SwitchingOperation

0..\*

0..\*

0..\*

0..1

Wires::EnergySource

Wires::EnergyConsumer

Wires::Fuse

Wires::Disconnector

Wires::ProtectedSwitch

Wires::Breaker

Wires::LoadBreakSwitch

Protection::RecloseSequence

0..\*

1

Protection::ProtectionEquipment

1

0..\*

Protection::SynchrocheckRelay

Protection::CurrentRelay

0..1

0..\*

Core::Terminal

1

0..\*

Core::ConnectivityNodeContainer

1

0..1

Core::EquipmentContainer

Wires::Line

0..1

0..1

Topology::TopologicalNode

Topology::ConnectivityNode

0..1

0..\*

0..1

0..\*

0..1

0..\*

0..\*

0..1

0..1

0..\*

Meas::Measurement

0..\*

0..1

Meas::Control

Meas::Discrete

Meas::Analog

Meas::DiscreteValue

Meas::AnalogValue

0..\*

0..\*

1

0..1

Meas::MeasurementValue

Meas::Command

Meas::SetPoint

0..1

0..1

0..1

Dijagram 1 - Klasni model

U ovom poglavlju je prikazan izgled modela sistema. Treba napomenuti da informacioni sistemi i program rade sa određenom vrstom podataka i informacija. Takođe, različiti sistemi funkcionišui po principu komunikacije sa drugim sistemima u preko koje razmenjuju različite informacije. Da bi komunikacija i rad sa informacija bio moguć potrebno je definisati model podataka. Konkretno, za naš sistem je korišćen CIM (***Common Information Model***). Ovaj model predstavlja standard za modelovanje elektroenergetskih sistema. Pomoću komponenti CIM-a je definisan potreban profil i na osnovu njega je napravljen željeni model koji se koristi kao dalja osnova za implementaciju funkcionalnosti u aplikaciji. Na dijagramu 2 je prikazan klasni dijagram modela sa svim klasama koje se koriste i koje su iz CIM-a.

# Arhitektura

Arhitektura celokupnog sistema predstavljena je na dijagramu 2. Sa dijagrama se mogu videti sve komponente i komunikacija između njih. U daljem tekstu će biti priloženi kratki opisi svake komponente, njenih funkcionalnosti i komunikacije sa drugim komponentama.

Dijagram 2 - Arhitektura sistema

**SCADA**

**FIZIČKI SISTEM**

**SIM**

**NMS**

**DB**

**CE**

**UI**

**TM**

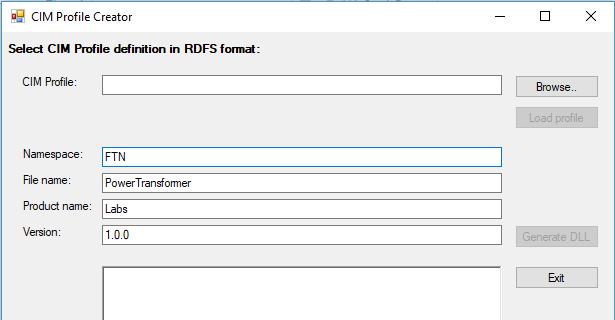
Napomena: Znak Ꚛ kod TM (***Transaction Manager***) znači da je ta komponenta u sprezi sa svim ostalim komponentama.

## Transaction Manager

Pre detaljnog objašnjenja svih drugih komponenti potrebno je objasniti ovu komponentu, jer je ona povezana na svaku drugu komponentu. Transaction Manager služi za koordinaciju u distribuiranoj transakciji prilikom dodavanja novih elemenata u sistem. Učesnici distribuirane transakcije su NMS, SCADA, CE, Simulator i UI. Kada korisnik pomoću aplikacije za importovanje doda novi element u NMS, on komunicira sa TM-om i svi ostali učesnici prolaze kroz faze ***enlist*** i ***prepare*** gde se prikupljaju informacije o novim elementima. Kada se prikupljanje novih informacija sa NMS-a završi, sledi operacija ***commit*** ili ***rollback*** u zavisnosti da li su svi učesnici uspešno učitali promene ili ne. Postupak distribuirane transakcije će se ponavljati tokom objašanjavanja ostalih komponenti i na taj način opisati kompletan postupak detaljno za svaku od komponenti.

## NMS

Prva komponenta arhitekture koja će biti predstavlena je NMS (***Network Model Service***). Kao što je već rečeno, u ovoj komponenti se definiše odgovarajući model za OMS sistem. Prilikom definisanja modela treba odrediti elemente iz CIM-a koji su potrebni za sistem. Određivanje potrebnih elemenata u našem slučaju se odnosi na definisanje odgovarajućeg CIM profila. Prilikom modelovanja korišćeni su različiti alati poput CIM tool-a i Enterprise Architect-a. Enterprise Architect alat se koristi za rad sa UML modelom u procesu modeliranja proširenja CIM standarda. Nakon kreiranja UML modela izgenerisana je XMI datoteka. XMI predstavlja standard za razmenu meta podataka. Korišćenjem CIM tool-a, nakon generisanja XMI datoteke, se dizajnira CIM profil na bazi UML-a koji je serijalizovan u okviru XMI formata. Kao rezultat rada sa CIM tool-om dobija se CIM profil u RDFS formatu. Dobijeni profil u RDFS formatu se dalje koristi za generisanje DLL-a. Program za generisanje DLL-a na osnovu ulaznog RDFS fajla prikazan je na slici 1. To je ***third party*** program čija je svrha da konvertuje definisan profil u DLL koji sadrži sve potrebne klase modela za naš OMS sistem.



Slika 1 - Third party aplikacija za konverziju RDFS-a u DLL

## Korisnički Interfejs (UI)

Korisnički interfejs, kao sprega sa kranjim korisnicima sistema (operaterima) je implementiran preko web tehnologije (Angular) za frontend, odnosno u .NET tehnologiji za bekend. Početna stranica je glavna stranica i na njoj se nalazi graf koji prikazuje stanje celokupnog sistema (mreže). Sve akcije (komandovanje, iščitavanje, itd.) se vrše preko grafa. Sa leve strane se nalazi meni (***browser***) koji sadrži tabele sa arhiviranim i aktuelnim ispadima. U tim tabelama se nalaze svi neophodni podaci vezani za ispade. Na slici 2 se nalazi izgled početne stranice.

**MESTO ZA SLIKU (Kada završimo projekat, biće finalna slika)**

## Fizički sistem (Simulator)

Kao simulacija fizičkog sistema je korišćena ***third-party*** aplikacija EasyModbus Simulator. Aplikacija simulira vrednosti određenih digitalnih i analognih veličina koje se pojavljuju u sistemu. Preko ove aplikacije se mogu očitavati vrednosti koje se prosleđuju SCADA-i radi praćenja nepravilnosti. Takođe, može se direktno komandovati menjanje određenih vrednosti radi testiranja. Aplikacija koristi modbus protokol za komunikaciju sa SCADA-om.

Za simulaciju ispada korišćena je aplikacija koja je projektovana posebno za naš sistem. Izabere se deo mreže na kom želimo da se desi ispad (ovo se radi u toku testiranja projekta, kao simulacija potencijalnog ispada u mreži) i nakon toga na korisničkom interfejsu posmatramo dalji razvoj automatike i eventualno delujemo (kao operater) u skladu sa protokolom i tipom ispada.

## Calculation Engine (CE)