

Fakultet tehničkih nauka

Univerzitet u Novom Sadu

**Čitanje i keširanje podataka o potrošnji**

Virtuelizacija procesa

Članovi tima:

Dušica Crevar PR5/2020

Stefan Malinović PR62/2020

Srđan Mlađenović PR72/2020

Nikola Inđić PR150/2020

Sadržaj

**1.Uvod I opis projektnog zadatka, User Manual3**

1.1 Uvod i user manual3

1.2 Način rada aplikacije4

**2.Arhitektura projekta sa tokom podataka5**

**3.Opis interfejsa sa osnovnim funkcionalnostima5**

**4. Opis tehnologija koje su korištene6**

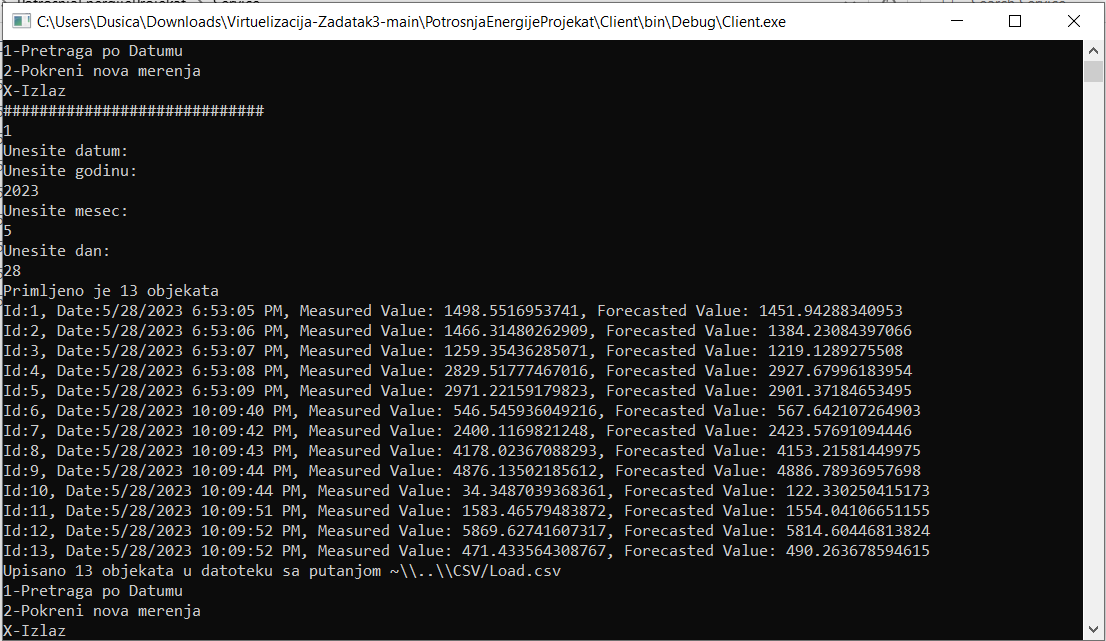
**5. Zaključak sa mogućim pravcima budućeg istraživanja I proširenjem zadatka** 6

5.1. Ideja projekta6

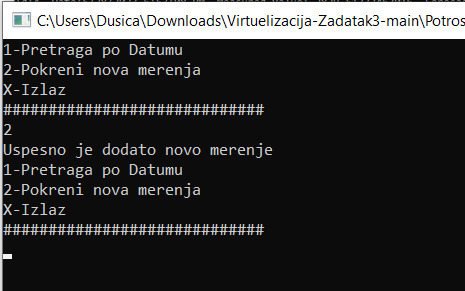
5.2 Pravci dodatnog proširenja i razvoja6

1.Uvod i opis projektnog zadatka, User Manual

**1.1 Uvod i user manual.** Projekat simulira komunikaciju između dva projekta, klijentskog I serverskog u cilju dobijanja podataka o izmerenoj električnoj energiji. Komunikacija između klijenta I servera pomoću WCF tehnologije. Klijent ima dve opcije (treća opcija je slanje znaka da je završio sa radom), a te dve opcije su mogućnost pretrage merenja po izabranom datumu ili pokretanje novih merenja koja će se izvršavati na serverskoj strani.

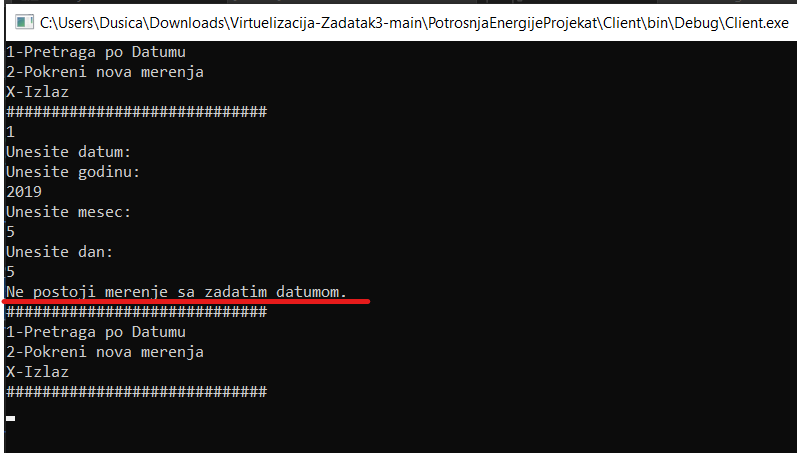


*Nakon izbora opcije 1 (Pretraga po datumu) korisnik dobija opciju unosa datuma tako što prvo unosi godinu, zatim mesec i na kraju dan. Time se dodatno pojednostavljuje korištenje jer klijent nema potrebe da stalno razmišlja o korektnom formatu. Nakon unosa dobija spisak merenja za uneti datum.*



*Nakon izbora opcije 2 (Pokreni nova merenja), klijent serveru daje znak da može da pokrene novo merenje što on i čini.*

**2. Način rada aplikacije.** Server komunicira sa dve baze podataka u cilju prikupljanja merenja za zadati datum koji je primio od strane klijenta. Dve baze podataka su XML baza podataka i InMemory baza podataka koja predstavlja Dictionary. Kada primi poruku od klijenta sa zadatim datumom, server se prvo obraća InMemory bazi podataka u cilju pretrage za željenim merenjima. Ukoliko pronađe podatke, on ih pakuje i šalje nazad klijentu kao odgovor. Odgovor se šalje kao Lista Load objekata koji se sastoje od njegovog ID-a, procenjene i izmerene vrednosti i datuma kada je merenje nastalo (Timestamp). Ukoliko server ne pronađe podatke u InMemory bazi podataka, on dalje nastavlja pretragu u XML bazi podataka. Pronađeni podaci se smeštaju u InMemory bazu podataka i šalju se klijentu kao odgovor. Međutim, ukoliko ni tu ne postoje odgovarajući podaci, kreira se novi objekat Audit koji nam javlja poruku o grešci i prosleđuje se klijentu kao odgovor na zahtev. Klasa Audit se sastoji od sopstvenog ID-a, datuma nastanka, tipa poruke (Info, Warning ili Error) i na kraju konačno i poruke.

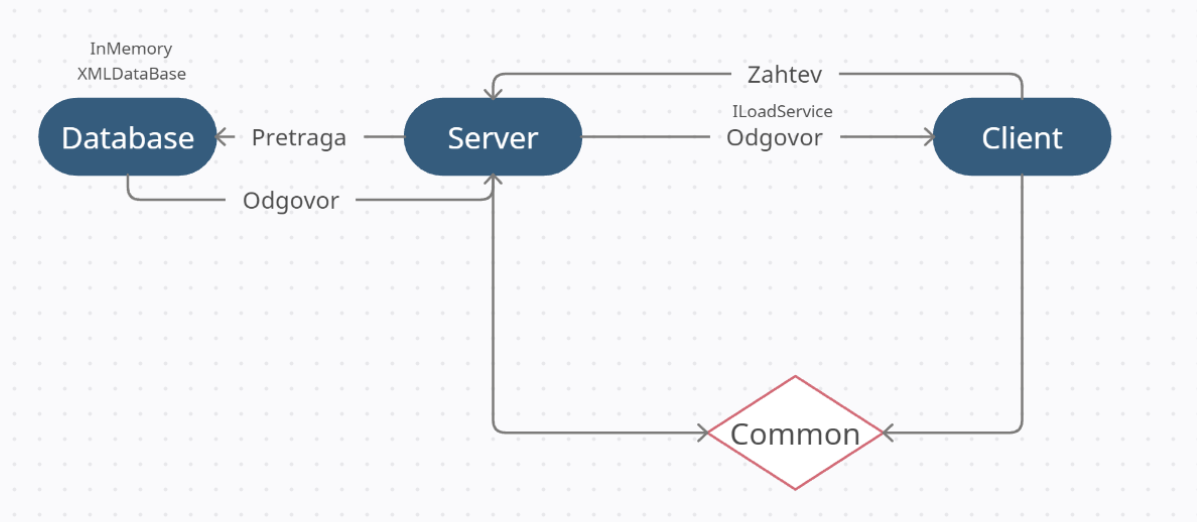


*Merenja nisu pronadjena, kao odgovor klijentu je prosleđena odgovarajuća poruka*

Server prati stanje zahteva i nova merenja zapisuje u XML bazu podataka, Load.xml, čuva istoriju pronalaska ili nepronalaska merenja za zadati datum (Audit objekte) u Audit.xml bazi podataka. Istorija pronađenih merenja se čuva u Load.csv fajlu koji se nalazi u CSV dadoteci čija je putanja određena u konfiguracionoj dadoteci App.config klijentske aplikacije. Važno je napomenuti da se zbog čuvanja resursa i memorije merenja iz InMemory baze podataka čuvaju određeno vreme (DataTimeout) koje je definisano u konfiguracionoj datoteci servisne aplikacije App.config i njeno podrazumevano vreme je 15 minuta nakon čega se brišu.

2.Arhitektura projekta sa tokom podataka

Arhitekutra projekta se zasniva na WCF tehnologiji koja omogućava komunikaciju između klijenta i servera u cilju razmene podataka. Obe aplikacije se referenciraju na projekat Common koji je ClassLibrary i služi kao njihov posrednik jer se u njemu nalaze zajednički podaci. Komunikacija je definisana u okviru contracta ILoadService koji predstavlja interfejs i on se upravo nalazi u zajedničkom projektu Common. Server komunicira sa bazom podataka (InMemory ili XmlDatabase) u cilju dobijanja skladištenih podataka.



*Arhitektura aplikacije – Common je predstavljen rombom iz razloga naglasenosti svojstva zajedničkog elementa koji je referenca i na server i na klijenta*

3. Opis interfejsa sa osnovnim funkcionalnostima

Kao što je već navedeno u prvoj tački (Uvod I osnovni pojmovi) predstavljen je korisnički interfejs koji se nalazi na klijentskoj strani. Korisnik ima dve osnovne mogućnosti, a to su pretraga merenja za zadati datum, kao i davanja signala serveru da može da izvrši novo dodatno merenje. Treća opcija je dodatna opcija koja omogućava bezbedno završavanje rada sa aplikacijom I njeno gašenje (opcija ‘X’). Detaljnije objašnjenje kao I primeri su navedeni u uvodnom obraćanju koje ujedno predstavlja I User Manual. Takođe je jasno prikazano šta će se desiti u slučaju unosa datuma za koje ne postoje merenja. (Poslaće se AUDIT poruka kao obavjest da ne postoje merenja za zadati datum).

4. Opis tehnologija koje su korištene

Aplikacija je izrađena u programskom okruženju C#, Visual Studio 2022, .Net Framework. Za komunikaciju između klijenta i servera je korišten WCF. Aplikacije u projektu su konzolne ili u slučaju Common-a, Class Library.

5. Zaključak sa mogućim pravcima budućeg istraživanja i proširenjem zadataka

**5.1. Ideja projekta.** Projekat je pored toga što je baziran na predstavljanju podataka koji mogu da se razmenjuju između klijenta I servera, veliku pažnju posvećuje i Memory Managementu, odnosno veliku pažnju posvećuje načinu na koji server komunicira sa bazama podataka, InMemory I XML. Dakle ukoliko podaci postoje u InMemory bazi podataka, server uopšte neće pristupati XML bazi podataka, što dodatno ubrzava program, ali i smanjuje opterećenje radne memorije jer se svi pristigli podaci ne moraju ponovo upisivati u InMemory bazu podataka. Takođe, nakon zadatog DataTimeota koji se konfiguriše u konfiguracionom fajlu serverske aplikacije, automatski se brišu nepotrebni podaci koji se nalaze u InMemory bazi. Time se dodatno čuvaju memorijski resursi jer se brišu nepotrebni podaci za koje se smatra da neće ponovo biti u upotrebi (prošlo je već 15 minuta, nema više potrebe za njima).

**5.2. Pravci dodatnog proširenja i razvoja.** Dodatno proširenje bi moglo biti omogućavanje novih mogućnosti klijentu, takođe pretrage na osnovu MeasuredValue ili ForecastValue gde bi klijent mogao da unese raspon vrednosti za merenja radi uvida u njih, slično tome I pretraga na osnovu maksimalne ili minimalne razlike odstpupanja između MeasuredValue I ForecastValue, takođe, dodatno usavršavanje interfejsa, kao određeni koraci koji bi omogućavali približavanje ili čak korištenje grafičkog interfejsa.