

Evidencija prognozirane i ostvarene potrošnje električne energije

Uvod

Aplikacija se bazira na klijent-server arhitekturi, komunikacija klijent-server odvija se putem WCF-a (Windows Communication Foundation). Klijent je implementiran kao konzolna aplikacija, gde je potrebno uneti putanju do foldera koji sadrži 'csv.zip' fajl, folder može da se nalazi bilo gde na računaru. Nakon unete putanje sledi provera da li se 'csv.zip' fajl nalazi na toj putanji, ako ne traži se ponovno unošenje putanje, ako je fajl tu smešta se u MemoryStream koji se putem WCF-a šalje na server. Server prihvata MemoryStream sa zip fajlom koji se raspakuje u folder 'csv' koji će se pojaviti u folderu servera. Server se koristi za obradu podataka koji se preuzimaju iz csv fajlova. Server vrši sledeće operacije kako bi obradio podatke :

- Parsira fajlove iz foldera 'csv' koji sadrži podatke, parsiranje fajlova podrazumeva određivanje: id-a, procenjene vrednosti (ForecastValue), izmerene vrednosti (MeasuredValue), datuma i vremena i naziva fajlaova
- Nakon parsiranja ovi podaci se upisuju u bazu podataka (XML ili In-Memory) i u zavisnosti od opcije u App.config fajlu pokreću Event-a i Delegate-a
- Nakon što su podaci upisani sledi određivanje odstupanja (greške) u zavisnosti od opcije u App.config fajlu - apsolutno procentualno odstupanje ili kvadratno odstupanje
- Posle određenog odstupanja ažurira se baza podataka (XML ili In-Memory) pomoću Event-a i Delegate-a
- Nakon završene obrade podataka server putem WCF-a šalje podatke iz baze podataka (XML ili In-Memory) klijentu

Klijent dobija povratnu informaciju od servera o tome da li je obrada podataka uspešna ili nije i ispisuje obrađene podatke.

Opis arhitekture

Klijent-Server arhitektura. Komunikacija se odvija putem WCF-a (Windows Communication Foundation). Windows Communication Foundation (WCF) je tehnologija za razvoj distribuirane aplikacije u .NET-u. Sva podešavanje su u App.config fajlovima i na klijentu i na serveru. Koristi se interfejs 'IConnection' i adresa 'net.tcp://localhost:8000'. Klijent sadrži dodatno podešavanje vezano za maksimalnu veličinu poruke koju može da prihvati, a to podešavanje je - maxReceivedMessageSize="200000".

Opis komponenata aplikacije

- Client - realizovan kao konzolna aplikacija koja traži od korisnika da unese putanju do foldera sa 'csv.zip'. Nakon unosa proverava se da li na unetoj putanji stvarno postoji fajl, ako ne traži se ponovno unošenje putanje, ako je fajl tu on se smešta u MemoryStream. Nakon kreiranja MemoryStream-a sledi inicijalizacija za komunikaciju preko WCFa. Nakon toga poziva se funkcija kojoj se kao parametar daje MemoryStream, ova funkcija služi za obradu podataka na serveru. Ako funkcija uspešno obradi podatke vraća 'true' (tačno) i sledi poziv funkcije koja prima od servera obrađene podatke. Pošto se šalje velika količina podataka u App.config fajlu je izvršena dodatna logika koja je prethodno opisana u opisu arhitekture.
- Common - realizovan kao class library, koji sadrži klase i interfejs. Klase su sledeće : Audit - služi za opis fajlova koji ne prođu validaciju. ImportedFile - služi za opis fajlova koji su obrađeni. Load - služi za opis obrade podataka, u ovim objektima će biti glavni podaci. Pored ovih klasa sadrži i interfejs IConnection koji služi da bi povezali klijenta sa serverom preko WCF-a. IConnection sadrži 4 glavne funkcije cele aplikacije a to su : DataProcessing() - funkcija koja kao parametar prima MemoryStream i obradjuje sve podatke. PrintLoad() - funkcija za ispis Load objekata. PrintImportedFile() - funkcija za ispis ImportedFile objekta. PrintAudit() - funkcija za ispis Audit objekata.
- Data Base - realizovan kao class library, sadrži 2 klase za XML i In-Memory baze podataka koje se koriste za skladištenje podataka. Klasa MyDB je realizacija XML baze podataka, sadrži funkcije za upis i čitanje XML baze podataka i funkcije za rad sa Event-ma i Delegate-ma. Klasa InMemoryDB je realizacija In-Memory baze podataka, sadrži funkciju koja upisuje presleđeni objekat u odgovarajući rečnik (Dictionary) u zavisnosti od tipa prosleđenog objekta i funkcije za rad sa Event-ma i Delegate-ma.
- Server - realizovan kao konzolna aplikacija, ispisuje poruku da je server uspešno pokrenut i nakon toga čeka da mu klijent prosledi MemoryStream sa 'csv.zip' fajlom. Nakon što prihvati MemoryStream prvo raspakuje .zip u folder server/csv. Nakon ovoga sledi parsiranje fajlova tj. za procenjene podatke se poziva njena funkcija, a za izmerene podatke njena. Prvo se obrade csv fajlovi za procenjene podatke i oni su upišu u bazu podataka (XML ili In-Memory) u zavisnosti od opcije u App.config fajlu pomoću Event-a. Nakon toga sledi obrada izmerenih podataka. Nakon završetka obrade ažuriraju se podaci u bazi podataka pomoću Event-a. Kada su i procenjeni i izmereni podaci upisani u bazu sledi izračunavanje odstupanja tj. greške u zavisnosti od opcije u App.config fajlu. Nakon izračunavanja dolazi do ažuriranja baze pomoću Event-a. Kada se završi izračunavanje odstupanja funkcija je završila svoj deo i šalje klijentu indikator 'true' (tačno) da označi da je obrada uspešna. Pored ove obrade postoje i funkcije koje služe da se proslede obrađeni podaci iz baze klijentu.

Tok podataka

Kratak opis toka podataka u aplikaciji :

- 1.Korak : Na klijentu se unosi putanja do foldera sa 'csv.zip' fajlom i to se smešta u MemoryStream.
- 2.Korak : MemoryStream sa 'csv.zip' fajlom se šalje na server putem WCF-a.
- 3.Korak : Obrada podataka na serveru.
- 4.Korak : Povratna informacija da li je obrada uspešno prošla, klijent ispisuje poruku o uspešnoj obradi.
- 5.Korak : Klijent prima sve obradjene podatke od servera i prikazuje korisniku šta se dogodilo tokom obrade podataka.

Vodič za instalaciju i konfiguraciju

Potrebno je imati Visual Studio 2022, za preuzimanje i instalaciju posetiti :

- <https://visualstudio.microsoft.com/vs/>

Projekat se može preuzeti sa GitHuba, naziv repozitorijuma 'Virtuelizacija_procesa' :

- <https://github.com/MastilovicRadoslav>

Nema nikakve dodatne konfiguracije, projekat se može pokrenuti nakon raspakivanja. Potrebno je samo da CSV fajlovi budu zipovani sa nazivom 'csv.zip'. Za dodatno oko pokretanja možete pročitati u uvodu!

Reference

- <https://openai.com/blog/chatgpt>
- <https://www.eepsi.ftn.uns.ac.rs/group/virtuelizacija-procesa/discussion>