

Konzistentan sistem

Predmet: Softver nadzorno-upravljačkih sistema

Ana Šinik SV11/2022
Nataša Kašiković SV69/2022
Lazar Nagulov SV61/2022

Novi Sad, 2025

Uvod.....	3
Arhitektura.....	3
Klientska aplikacija.....	3
Serverska aplikacija.....	4
Aplikacija koja zahteva poravnanje.....	4
Implementacija.....	4
Struktura rešenja.....	4
WCF konfiguracija.....	5
SQLite.....	6
Kvorum i replika.....	6
CAP teorema.....	6
Primeri.....	7

Uvod

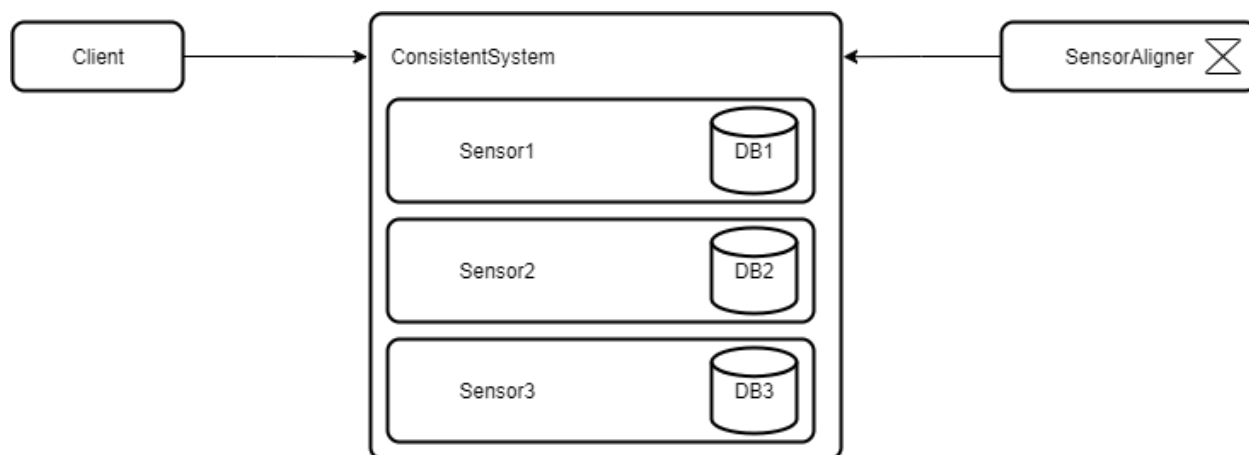
Ovaj dokument predstavlja dokumentaciju projekta za predmet softver nadzorno-upravljačkih sistema. Sistem se sastoji od 3 senzora koji mere temperaturu, klijentske aplikacije koja ima mogućnost čitanja poslednje izmerene vrednosti sa senzora i aplikacije koja vrši poravnanje senzora na svaki minut, nezavisno od klijentske aplikacije.

Klijent i server za komunikaciju koriste *WCF* (eng. *Windows Communication Foundation*), dok svaki senzor sadrži svoju *SQLite* bazu podataka za čuvanje izmerene temperature.

Arhitektura

Sistem se sastoji iz tri aplikacije:

- Klijentska aplikacija (Client)
- Serverska aplikacija (ConsistentSystem)
- Aplikacija koja zahteva poravnanje, nezavisno od klijentske aplikacije (SensorAligner)



Slika 1 - Arhitektura

Klijentska aplikacija

Klijent očitava poslednje temperature sa svih senzora i računa prosečnu vrednost. Sistem koristi kvorum replikaciju, tako da je merenje prihvaćeno kao tačno ako se najmanje dva od tri senzora nalaze u opsegu ± 5 od prosečne vrednosti. U suprotnom, smatra se da nije postignut kvorum i pokreće se poravnanje. Nakon

poravnanja, svi senzori za poslednju vrednost uzimaju srednju vrednost poslednjih merenja.

Serverska aplikacija

Pokreće tri senzora koji nezavisno popunjavaju svoju SQLite bazu podataka. Merenje se vrši na svakih 1-10 sekundi. U slučaju da dođe do poravnanje, klijent se obaveštava o početku poravnanja, sprečava čitanje vrednosti iz baze sve dok se poravnanje ne završi. Klijent dobija obaveštenje da je poravnanje završeno i u mogućnosti je da ponovo čita poslednje vrednosti sa senzora.

Aplikacija koja zahteva poravnanje

Na svaki minut, šalje zahtev za poravnanje serveru, nezavisno od klijenta. Slično kao kod poravnanja na zahtev klijenta, dokle god traje poravnanje, nije moguće čitati vrednosti sa senzora.

Implementacija

U nastavku se nalazi detaljniji opis samog rešenja. Čitav projekat je javan na sledećem github linku: <https://github.com/lazarnagulov/consistent-system>

Struktura rešenja

- Client
 - Program.cs
- ConsistentSystem
 - Common
 - Models
 - Measurement.cs
 - DataBaseHelper.cs
 - Contracts
 - ISensorService.cs
 - Sensor.Core
 - SensorRepository.cs
 - SensorWorker.cs
 - SensorService.cs
- SensorAligner
 - Program.cs

WCF konfiguracija

Komunikacija sa klijentom se obavlja putem sledećih interfejsa:

```
namespace ConsistentSystem.Contracts
{
    [ServiceContract(CallbackContract = typeof(ISensorCallback))]
    public interface ISensorService
    {
        [OperationContract]
        Measurement GetLastMeasurement(string sensorId);

        [OperationContract(IsOneWay = true)]
        void Align(double value);

        [OperationContract]
        string GetSensorName(string sensorId);
    }

    public interface ISensorCallback
    {
        [OperationContract(IsOneWay = true)]
        void OnAlignmentStarted();

        [OperationContract(IsOneWay = true)]
        void OnAlignmentCompleted(double alignedValue);
    }
}
```

Kod 1 - WCF Interfejsi

Konfiguracija WCF servisa se nalazi u *Web.config* datoteci.

```
<services>
  <service name="ConsistentSystem.Sensor.SensorService" behaviorConfiguration="DefaultBehavior">
    <endpoint address="" binding="wsDualHttpBinding" contract="ConsistentSystem.Contracts.ISensorService" />
    <host>
      <baseAddresses>
        <add baseAddress="http://localhost/SensorService" />
      </baseAddresses>
    </host>
  </service>
</services>
```

Kod 2 - WCF konfiguracija

SQLite

Svaki senzor u sebi sadrži *SQLite* bazu podataka sa sledećom tabelom:

```
CREATE TABLE IF NOT EXISTS Measurements (  
    Id INTEGER PRIMARY KEY AUTOINCREMENT,  
    Temperature REAL NOT NULL,  
    Timestamp DATETIME DEFAULT CURRENT_TIMESTAMP  
);
```

Kod 2 - Kreiranje SQL tabele

Dok se sama komunikacija vrši pomoću *System.Data.SQLite* biblioteke.

Kvorum i replika

U ovom sistemu doslednost podataka se postiže korišćenjem kvorum replikacije. To znači da se vrednost merenja smatra validnom ako se najmanje dve replike senzora slažu sa prosečnom vrednošću u definisanom tolerancionom opsegu (± 5). Ovaj pristup omogućava da sistem radi čak i u slučaju kada jedan senzor daje odstupajuće ili pogrešne vrednosti, što povećava pouzdanost očitavanja.

CAP teorema

CAP kaže da u distribuiranom sistemu sa replikama ne možeš istovremeno uvek imati:

- C – konzistentnost (sve replike uvek vraćaju istu vrednost),
- A – dostupnost (sistem uvek odgovara na upit),
- P – toleranciju particije (sistem i dalje radi i ako je deo mreže nedostupan).

U projektu se može uočiti primena CAP teoreme:

- Konzistentnost se postiže poravnavanjem senzora, tako da nakon poravnanja svi senzori imaju iste vrednosti.
- Dostupnost se žrtvuje, jer tokom poravnanja klijent ne može da čita vrednosti iz senzora dok se proces ne završi.
- Tolerancija particije je obezbeđena kroz kvorum - sistem funkcioniše i kada jedan senzor nije dostupan ili daje odstupajuće vrednosti, jer se vrednost prihvata na osnovu najmanje dva senzora koja su u skladu.

Primeri

U nastavku se nalaze primeri funkcionisanja aplikacije.

```
Sensor: 95, Temp: 23.90 °C at 10/1/2025 12:52:05 PM
Sensor: 92, Temp: 14.96 °C at 10/1/2025 12:52:07 PM
Sensor: 101, Temp: 24.15 °C at 10/1/2025 12:52:05 PM
[10/1/2025 2:52:11 PM] Valid measurements found (avg=21.00 °C):
```

Primer 1 - Priprihvaćena izračunata vrednost

```
Sensor: 27, Temp: 10.08 °C at 10/1/2025 12:46:20 PM
Sensor: 28, Temp: 20.32 °C at 10/1/2025 9:40:08 AM
Sensor: 30, Temp: 47.43 °C at 10/1/2025 12:46:19 PM
[10/1/2025 2:46:21 PM] Alignment started. Please wait...
[10/1/2025 2:46:21 PM] Alignment completed. All sensors now at 25.94 °C
```

Primer 2 - Klijent zahteva poravnanje

```
Sensor: 45, Temp: 21.76 °C at 10/1/2025 12:48:09 PM
Sensor: 49, Temp: 41.91 °C at 10/1/2025 12:48:12 PM
Sensor: 52, Temp: 10.22 °C at 10/1/2025 12:48:09 PM
[10/1/2025 2:48:14 PM] Alignment started. Please wait...
[10/1/2025 2:48:14 PM] Alignment completed. All sensors now at 24.63 °C
Sensor: 46, Temp: 24.63 °C at 10/1/2025 12:48:14 PM
Sensor: 50, Temp: 24.63 °C at 10/1/2025 12:48:14 PM
Sensor: 53, Temp: 24.63 °C at 10/1/2025 12:48:14 PM
[10/1/2025 2:48:16 PM] Valid measurements found (avg=24.63 °C):
```

Primer 3 - Stanje nakon poravnanja

```
[10/1/2025 2:56:18 PM] Alignment started...
[10/1/2025 2:56:18 PM] Alignment completed: 38.02 °C
[10/1/2025 2:57:18 PM] Alignment started...
[10/1/2025 2:57:18 PM] Alignment completed: 38.30 °C
```

Primer 4 - Automatsko poravnanje

```
[10/3/2025 12:37:23 PM] Alignment started. Please wait...
Sensors are aligning; no measurements available.
Sensors are aligning; no measurements available.
```

Primer 5 - Sprečavanje pristupa tokom poravnanja