**Projekat (DRUS)**

Zadatak projekta je da se kreira entity baza koja skladišti informacije o vrednostima koje se generišu na senzorima (u našem slučaju 10 senzora). Potrebno je kreirati 10 tabela u bazi, gde će svaka tabela sadržati informacije o svakom senzoru zasebno. Nakon toga potrebno je kreirati wcf service i klijent aplikaciju koje će međusobno komunicirati i skladištiti odgovarajuće podatke u bazu. Struktura projekta je prikazana sa slici:

A screenshot of a computer

Description automatically generated

**Entity baza**

Za kreiranje baze koristimo Entity Framework i zatim definišemo klase sa atributima koji će se nalaziti u tabeli baze. U našem slučaju definišemo atribute Id (int), DateTime (Timestamp) i Value (value). Nakon toga inicijalizujemo konstruktor ove klase i definišemo klasu koja nasleđuje klasu DBContext. U svakoj bazi treba da se nalazi onoliko vrednosti koliko puta je izvršena metoda za poravnanje vrednosti koja se izvršava na svaki minut od pokretanja programa.

A screen shot of a computer program

Description automatically generatedA screen shot of a computer program

Description automatically generated

**WCF Service aplikacija**

Najpre potrebno je kreirati Iservice1 Interface sa metodama koje će se koristiti u daljem kodu i Callback Interface koji sadrži metode koje obaveštavaju klijenta o vrednostima generisanim sa senzora.

A screen shot of a computer

Description automatically generated

Nakon toga potrebno je kreirati klasu Service1 koja nasleđuje Interface Iservice1 i definiše sve funkcije koje se nalaze u interfejsu.

**Subscribe** – metoda koja omogućava klijentu da bude pretplaćen i da dobija informacije na njegovoj konzoli o trenutnim vrednostima generisanim na senzorima.

A computer screen with a black background

Description automatically generated

**ClearAllSensorData** – omogućava brisanje sadržaja iz svih tabela i poziva se prilikom pokretanja programa.

A screen shot of a computer

Description automatically generated

**StartSensors** – metoda koja inicijalizuje tajmere za pokretanje funkcije poravnanja i nakon toga inicijalizuje tajmer za simuliranje vrednosti na senzorima za svaki senzor posebno u periodu od 1-10 sekundi nasumično.

A screen shot of a computer code

Description automatically generated

**StopSensors** – ovom metodom se oslobađaju resursi svih tajmera i zaustavlja se sam rad aplikacije.

A screen shot of a computer program

Description automatically generated

**AlignReplicasCallback** – callback metode koja vrši poravnanje vrednosti u bazi. Na strani klijenta obaveštava kada je došlo do poravnanja i u slučaju ako je došlo do nekog problema.

A computer screen with text

Description automatically generated

**AlignReplicas** – predstavlja poravnanje vrednosti u bazi. Najpre uzimaju se sve vrednosti iz rečnika u kojem su skladištene informacije prvobitno (<int, List<double>>) i računa se srednja vrednost za svaki senzor posebno. Zatim se računa granica koja predstavlja uslove koje vrednost mora da zadovoljava da bi bila upisana u bazu, u suprotnom uzima se vremenski poslednja vrednost i skladišti se u bazu.

A screen shot of a computer program

Description automatically generated

**AddReadingToDatabase** – funkcija koja skladišti odgovarajuću vrednost iz rečnika u bazu i poziva se kada se provere svi uslovi za skladištenje.

**A screen shot of a computer program

Description automatically generated**

**SimulateSensorReading** – metoda koja simulira vrednosti sa svih senzora i zaokružuje vrednosti na 2 decimale. Nakon toga ih skladišti u rečnik i obaveštava klijente o vrednosti sa senzora.

A screen shot of a computer program

Description automatically generated

**Klijent aplikacija**

Sadrži klasu Callback u kojoj se nalazi sadržaj koji će biti prikazan na konzoli klijenta tokom rada programa kada dođe do promene na service strani. Nakon toga nalazi se klasičan Main program u kojem se pokreću senzori, pretplaćuje klijent i pritiskom tastera ENTER zaustavlja program.

A screen shot of a computer program

Description automatically generated

Prikaz rada programa:

A screenshot of a computer

Description automatically generated A screenshot of a computer

Description automatically generated

**Petar Popov E2 52/2024**