Dokumentacija SSDP protokola

Opis:

Ova dokumentacija opisuje Java kod za klijente i server koji koriste multicast socket komunikaciju radi razmene podataka. Klijenti šalju obaveštenja (notify) serveru prilikom povezivanja na mrežu o uređajima koji su dostupni na mreži, dok server prima ta obaveštenja i održava listu uređaja. Takođe kada je potrebno server šalje MSearch poruku klijentima radi otkrivanja servisa na mreži. Oba dela koda koriste multicastSocket za slanje i primanje poruka.

Klijent (Client.java):

- Klijent je odgovoran za slanje obaveštenja serveru o dostupnim uređajima.

- Koristi se kroz komandnu liniju sa tri argumenta: IP adresa uredjaja, kategorija uređaja i tip uređaja.

- Kategorije uređaja su: Senzor, Aktuator, Kontroler i Aplikacija.

- Tipovi uređaja su specifični za svaku kategoriju (npr. za Senzor: Temperatura, Vlažnost, Svetlo; za Aktuator: Ventilator, Pumpa, Svetlo).

- Klijent šalje obaveštenje u multicast grupu na određenom portu koristeći sendData() metodu iz SocketFunctions klase.

- Zatim klijent periodično šalje obaveštenje na svakih 7 sekundi.

Server (Server.java):

- Server osluškuje multicast grupu i prima obaveštenja od klijenata o dostupnim uređajima.

- Kada primi obaveštenje (notify), server kreira objekat klase Device koji sadrži informacije o uređaju (IP adresa, kategorija, tip, ID) i dodaje ga u listu uređaja.

- Server takođe periodično prikazuje sve uređaje na konzoli na svakih 2 sekunde, koristeći printAllDevices() metodu klase Device.

Klase:

1. Client: Glavna klasa koja sadrži main metodu i iz koje se šalju obaveštenja serveru.

2. ReadThread: Runnable klasa koja se koristi za asinhrono čitanje poruka sa multicast soketa. Kada primi "msearch" poruku od servera, šalje obaveštenje serveru koristeći sendData() metodu iz SocketFunctions klase.

3. SocketFunctions: Klasa koja sadrži metode za slanje i primanje poruka preko multicast soketa.

4. Server: Glavna klasa koja sadrži main metodu i odgovorna je za prijem obaveštenja od klijenta i održavanje liste uređaja.

5. ReadThreadServer: Runnable klasa koja se koristi za asinhrono čitanje poruka sa multicast soketa. Kada primi "notify" poruku od klijenta, kreira objekat klase Device i dodaje ga u listu uređaja.

6. SocketFunctionsServer: Klasa koja sadrži metode za slanje i primanje poruka preko multicast soketa specifične za server.

7. Device: Klasa koja predstavlja uređaj. Sadrži informacije o IP adresi, kategoriji, tipu, ID-u uređaja i vremenu poslednjeg ažuriranja.

- Konstruktor klase Device prima poruku obaveštenja od klijenta i iz nje izvlači relevantne informacije o uređaju.

- Metoda printDevice() ispisuje informacije o uređaju na konzolu.

- Metode getCategory(), getType(), getIp(), getId() i getTime() vraćaju odgovarajuće vrednosti atributa uređaja.

- Metoda setTime() postavlja vreme poslednjeg ažuriranja uređaja.

8. SocketFunctions: Klasa koja sadrži statične metode za slanje i primanje poruka preko multicast soketa.

- Metoda sendData() prima poruku, multicast grupu, port i multicast soket te šalje poruku preko soketa.

- Metoda recvData() prima multicast grupu, port i multicast soket te čeka na prijem poruke, koju zatim vraća kao rezultat.

9. SocketFunctionsServer: Klasa koja sadrži statične metode za slanje i primanje poruka preko multicast soketa, specifične za server.

- Metoda sendData() i recvData() su ekvivalentne metodama iz klase SocketFunctions, ali prilagođene za server.

Napomena:

Kod je napisan na osnovu Java programskog jezika i koristi se za multicast komunikaciju između klijenata i servera. Klijenti šalju obaveštenja serveru o dostupnim uređajima, dok server prima ta obaveštenja i održava listu uređaja. Pokretanje više klijenata se radi pokretanjem script.sh skripte.

1 2

2