# Dokumentacija Projekta: Distribuirani Sistem za Upravljanje Fajlovima

## 1. Uvod

Ovaj dokument opisuje rad i strukturu distribuiranog sistema za upravljanje fajlovima, baziranog na Chord (DHT) arhitekturi. Sistem omogućava efikasno skladištenje, preuzimanje i upravljanje fajlovima u distribuiranoj mreži. Dokumentacija pruža detaljne informacije o implementaciji ključnih komponenti sistema.

## 2. Opis sistema

Sistem se sastoji od distribuirane mreže čvorova koji koriste Chord DHT arhitekturu za skladištenje i preuzimanje fajlova. Svaki čvor u mreži može skladištiti i preuzimati fajlove, čime se osigurava visok stepen dostupnosti i otpornosti na otkaze. Sistem je projektovan da balansira opterećenje među čvorovima i obezbedi sigurnu pohranu podataka.

## 3. Komande

Sistem podržava sledeće komande koje omogućavaju upravljanje fajlovima i čvorovima u mreži:  
  
- add\_file [path] [private/public]: Dodaje novu datoteku u mrežu sa opcijom privatnog ili javnog deljenja.  
- view\_files [address:port]: Pregled javnih i privatnih datoteka na određenom čvoru.  
- remove\_file [filename]: Uklanja datoteku iz mreže.  
- add\_friend [address:port]: Dodaje novi čvor u listu prijatelja.  
- stop: Zaustavlja rad čvora u mreži.

## 4. Konfiguracija sistema

Sistem koristi konfiguracionu datoteku koja definiše ključne parametre rada:  
  
- Radni koren: Putanja do direktorijuma na lokalnom sistemu gde se čuvaju datoteke.  
- Port: Port na kojem čvor sluša.  
- IP adresa i port bootstrap čvora: Informacije potrebne za povezivanje sa bootstrap čvorom.  
- Granice za detekciju otkaza: Vremenski intervali za detekciju soft i hard fail-a.

## 5. Ključne komponente

\*\*ChordState Klasa\*\*  
Implementira logiku Chord sistema, uključujući upravljanje čvorovima, successor i predecessor tabelama, kao i skladištenjem i preuzimanjem vrednosti.  
  
- Atributi:  
 - CHORD\_SIZE: Maksimalna vrednost ključa.  
 - chordLevel: Logaritamski nivo sistema.  
 - successorTable: Tabela shortcut-ova.  
 - valueMap: Mapa vrednosti koje se skladište na čvoru.  
  
- Metode:  
 - chordHash(int value): Računa Chord ID za dati ključ.  
 - putValue(int key, DistributedFile value): Skladišti vrednost lokalno ili prosleđuje dalje.  
 - getValue(int key): Preuzima vrednost lokalno ili šalje zahtev drugom čvoru.  
  
\*\*Buddy Klasa\*\*  
Periodično ping-uje prethodnika i sledbenika kako bi detektovala otkaze čvorova i osigurala reorganizaciju mreže.  
  
- Atributi:  
 - PING\_TIME: Interval pingovanja.  
 - SOFT\_FAIL: Interval za detekciju soft fail-a.  
 - HARD\_FAIL: Interval za detekciju hard fail-a.  
  
- Metode:  
 - run(): Periodično ping-ovanje čvorova i detekcija otkaza.  
 - buddyPonged(int buddyPort): Ažurira status kada se primi odgovor.  
  
\*\*Backuper Klasa\*\*  
Upravlja pravljenjem rezervnih kopija podataka kako bi se osigurala otpornost na otkaze.  
  
- Metode:  
 - run(): Periodično šalje rezervne kopije podataka successor-u.  
  
\*\*DistributedFile Klasa\*\*  
Predstavlja datoteku koja se distribuira unutar Chord mreže.  
  
- Atributi:  
 - isPublic: Oznaka da li je datoteka javna ili privatna.  
 - filePath: Putanja do datoteke.  
  
- Metode:  
 - toString(): Vraća string reprezentaciju datoteke.

## 6. Detekcija otkaza i reorganizacija

Sistem koristi Buddy mehanizam za detekciju otkaza. Svaki čvor periodično ping-uje svog prethodnika i sledbenika kako bi detektovao eventualne otkaze. Kada čvor ne odgovori na ping, šalje se poruka svim čvorovima u mreži da je čvor otkazao, što pokreće reorganizaciju mreže.

## 7. Backup sistema

Sistem pravi rezervne kopije podataka svakih pet sekundi, kako bi osigurao da podaci nisu izgubljeni u slučaju otkaza čvorova. Svaki čvor šalje spisak svojih fajlova successor čvoru, koji čuva ove fajlove kao rezervne kopije.

Mateja Civkaroski RN75/20