

Προγραμματιστική Άσκηση Υλοποίησης Κατανεμημένου Συστήματος Ομότιμων (P2P) για την ανταλλαγή αρχείων – P-tO-P-A (Φάση 2)

Ομάδα 1034

Κουλουρίδης Σωτήριος-Παναγιώτης 3200082

Πέτρος Καλύβας 3190070

Η υλοποίηση μας καλύπτει την λογική των ερωτημάτων που ζητήθηκαν στο πλαίσιο της εργασίας και αποτελείται από τα ακόλουθα αρχεία :

- **Peer** : Η κλάση του Peer , όπου διαχειρίζεται συνδέσεις με trackers και άλλους peers , καθώς ανοίγει sockets , τα απαραίτητα threads , περιέχει λίστες και πολύ σημαντικές μεταβλητές . Επίσης περιέχει πολλές μεθόδους για επεξεργασία και διαμοιρασμο δεδομένων , που χρησιμοποιούνται απο τα threads .
- **PeerMessageReceiver** : Thread του peer που δέχεται μηνύματα από τον Tracker .
- **PeerMessageSender** : Thread του peer που στέλνει μηνύματα στον Tracker .
- **FileCollaborativeSender** : Thread του peer που διαχειρίζεται την επικοινωνία με άλλους Peers .
- **CollaborativeDownloadHandler** : Thread του FileSender που διαχειρίζεται την λειτουργία CollaborativeDownload .
- **Partition** : Thread του PeerMessageSender που μετα το login μοιράζει το αρχείο σε 10 νέα αρχεία της μορφής file-1.txt , file-2.txt ... file-10.txt .
- **fileAndPorts.java** : Χρησιμοποιείται ώστε να αποθηκεύονται μαζί και να είναι συνδεδεμένα τα ονόματα αρχείων με τα ids των user .
- **Tracker** : Η κλάση του Tracker , όπου διαχειρίζεται συνδέσεις με τους peers , καθώς ανοίγει sockets , τα threads του PeerHandler , περιέχει λίστες και πολύ σημαντικές μεταβλητές . Επίσης περιέχει πολλές μεθόδους για επεξεργασία και διαμοιρασμο δεδομένων , που χρησιμοποιούνται απο τα threads .
- **PeerHandler** : Thread του Tracker που διαχειρίζεται τα μηνύματα που στέλνει καθε Peer και αποστέλλει την ανάλογη απάντηση . Μέσω των μεθόδων που προσφέρει ο Tracker συμβάλλει καθοριστικά στην επεξεργασία των λιστών και μεταβλητών του Tracker .
- **User** : Χρησιμοποιείται για να έχουμε τα δεδομένα κάθε user συγκεντρωμένα . Ένα variable τύπου User περιέχει όλα τα δεδομένα που μπορεί να έχει ένας user . Για αυτό ανάλογα με την φάση που βρίσκεται το πρόγραμμα έχουμε φτιάξει πολλές διαφορετικές εκδοχές του constructor του .
- **PortCounter** : Χρησιμοποιείται για να έχουμε τα ports και τις φορές που εξυπηρέτησαν τον Peer μας συγκεντρωμένα . Χρησιμοποιήθηκε για να βρούμε ποιός Peer έχει εξυπηρετήσει τον Peer μας περισσότερες φορές .

Τεκμηρίωση εφαρμογής :

Το προγράμμα μας αναπτύχθηκε στο vs code , και η εκτέλεση του γινόταν μέσω του cmd . Στο προγράμμα μας περιέχεται η λογική όλων των ερωτημάτων , ομώς κατά την εκτέλεση υπάρχει απόκλιση από τις προδιαγραφές .

Αυτές είναι οι ακόλουθες εντολές που έχει κάθε peer διαθέσιμες :

```
//-----Available actions-----//  
//register username password  
//login username password  
//list  
//details id file  
//collaborativeDownload file  
//simpleDownload file  
//logout id  
//-----//
```

Γενικότερη αξιολόγηση :

Η λογική της δεύτερης φάσης του project μας δυσκόλεψε , καθώς συνεχώς αντιμετωπίζαμε προβλήματα κατά την σύνδεση των sockets , τα μηνύματα δεν στέλνονταν σωστά κλπ . Πολύ συχνό, να δημιουργούσαμε επικοινωνία που έκανε το προγράμμα μας να κολλάει σε διάφορα σημεία . Για αυτό το λόγο σε σχόλια υπάρχουν διάφορες εναλλακτικές που μελετούσαμε κατά την ανάπτυξη της εφαρμογής . Η εναλλακτική που μας φαινόταν λογική αλλά δεν λειτούργησε ήταν στο collaborativeDownloadHandler αντι να δημιουργούμε sockets συνέχεια , να τα κρατάμε σε μια λίστα . Η java όμως τα έκλεινε αυτόματα και δεν βρήκαμε τρόπο να τα ξαναανοίγουμε .

SCREENSHOTS:

```
//register username password  
//login username password
```

Peer:

```
C:\Users\pkaly\Docs\diktia\Test_Diktya_3\Test_Diktya>java Peer  
Peer started on port 8970...  
Connected to tracker...  
register p 1  
Received from tracker: Registration successful.  
login p 1  
Received from tracker: id: 1
```

Tracker:

```
C:\Users\pkaly\Docs\diktia\Test_Diktya_3\Test_Diktya>java Tracker  
Tracker started on port 8080...  
Peer connected: Socket[addr=/127.0.0.1,port=2950,localport=8080]  
Peer connected: Socket[addr=/127.0.0.1,port=2951,localport=8080]  
Received from peer: register p 1  
Registered user: p  
Received from peer: login p 1  
id is : 1  
Received from peer: inform 1 192.168.56.1 8970 file1.txt  
Peer is a seeder for files :  
File-to-Peers Mapping:  
File: file1.txt, Peers: [1]
```

## /list

Peer:

list

Received from tracker: Available files in p2p: file1.txt file2.txt file3.txt file4.txt

Tracker:

Received from peer: list

## details id file

Peer:

Received from tracker: Seeder-port: 8973 File: file3.txt

Seeder-port:

8973

File:

file3.txt

Seeder-port: 8973 File: file3.txt

Received from tracker: File file3.txt BestPeerSocket: 8973 Min time is: 31.0

Tracker:

Received from peer: details 1 file3.txt

[3]

[3]

Received from peer: Sending Check Active response from peer with id 3

Peer with ID 3 is active.

## collaborativeDownload file

(Το κάθε τμήμα αρχείου εμφανίζεται καθώς κατεβαίνει)

Peer (download):

collaborativeDownload file3.txt

Received segment: file3-1.txt from port: 8973

Received segment: file3-8.txt from port: 8973

Received segment: file3-9.txt from port: 8973

Received segment: file3-3.txt from port: 8973

Received segment: file3-2.txt from port: 8973

Received segment: file3-5.txt from port: 8973

Received segment: file3-7.txt from port: 8973

Received segment: file3-10.txt from port: 8973

Received segment: file3-6.txt from port: 8973

Received segment: file3-4.txt from port: 8973

All segments received for file: file3.txt

Combined file saved: C:\Users\pkaly\Docs\diktia\Test\_Diktya\_3\Test\_Diktya\shared\_folder1\file3.txt

Peer (send):

Peer connected: Socket[addr=/127.0.0.1,port=55676,localport=8973]

Peer connected: Socket[addr=/127.0.0.1,port=55677,localport=8973]

Sending the File to the Server . Also got filename : file3-2.txt

Peer connected: Socket[addr=/127.0.0.1,port=55678,localport=8973]

Peer connected: Socket[addr=/127.0.0.1,port=55679,localport=8973]

Sending the File to the Server . Also got filename : file3-5.txt

Peer connected: Socket[addr=/127.0.0.1,port=55680,localport=8973]

Peer connected: Socket[addr=/127.0.0.1,port=55681,localport=8973]

Peer connected: Socket[addr=/127.0.0.1,port=55682,localport=8973]

Sending the File to the Server . Also got filename : file3-7.txt

Αποτυχημένη αναζήτηση αρχείου:

Peer:

Received from tracker: Available files in p2p: file1.txt file2.txt file3.txt file4.txt

details 1 file1.txt

Received from tracker: File not tracked by any peer

Tracker:

Received from peer: details 1 file1.txt

## simpleDownload file

Peer (download):

simpleDownload file3.txt

Peer (send):

Sending the File to the Server . Also got filename : file3.txt

logout id

Peer:

logout 3

Received from tracker: Logout successful

Tracker:

Received from peer: logout 3