## **Project Screenshots**

- Java Development Kit (JDK) and JRE (Java Runtime Environment) is installed on the System
- Ant Tool is used to run the Build Script
- CentralIndxServer folder contains -

The Source file of the CentralIndxServer – "CentralIndxServer.java" Compiled files - "CentralIndxServer.class", "PortListener.class" and "begin.class" "build.xml" which is an "ant"-BuildScript to compile and build the CentalIndxServer.

Peer1, Peer2 and Peer3 folders contain –

The Source file "PeerServer.java"

Compiled files – "PeerServer.class" and "PortListenerSend.class"

"indxip.txt" – contains the IP Address of the CentralIndxServer

Testfiles- Simple text files to test the system

"build.xml" to compile and build the peers

### Steps taken-

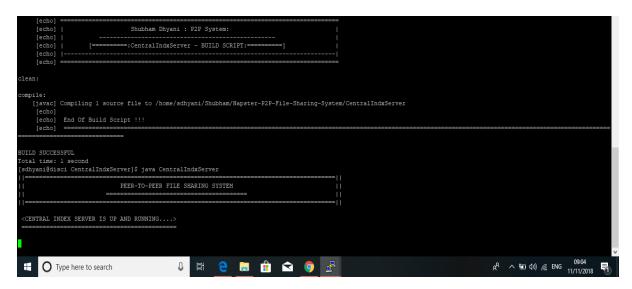
- 1. Open Multiple-tabs and navigate into CentralIndxServer folder on the First tab
- 2. On the Command prompt Run the build-script using the ant command. \$ and
- 3. The Build-script is executed and it cleans and compiles the files of the CentralIndxServer present in the CentralIndxServer directory. After successful build, a message is displayed on the terminal as shown in the screenshot:

Build Successful Total Time: 1 second

4. Run the CentralIndxServer by the command, A message is displayed saying the "<Central Index Server is Up and Running>" This runs the CentralIndxServer and it starts listening to other Peers on the Port:

\$ java CentralIndxServer

The following message is displayed-

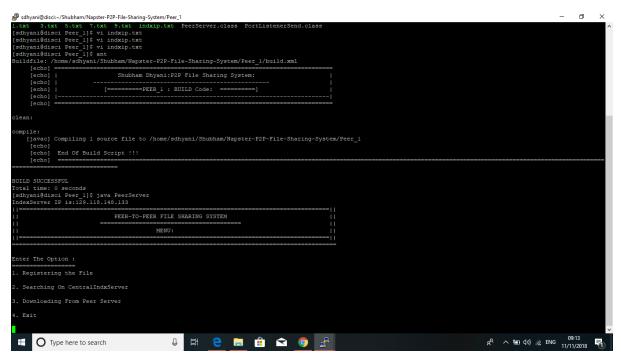


- 5. Below three screenshots shows the peer1, peer2 and peer3
- 6. Run the build-script using the ant command on each peer
- 7. The Build-script is executed and it cleans and compiles the files of the peers present in the Peer\_1, Peer\_2 and Peer\_3 directory.\$ ant
- Run Peer1, Peer2 and Peer3 using the command\$ java PeerServer

```
[sdhyani@disci Peer_1]% 1s

10.txt 2.txt 4.txt 6.txt 8.txt build.xml sy.txt PeerServer.java testfile.txt

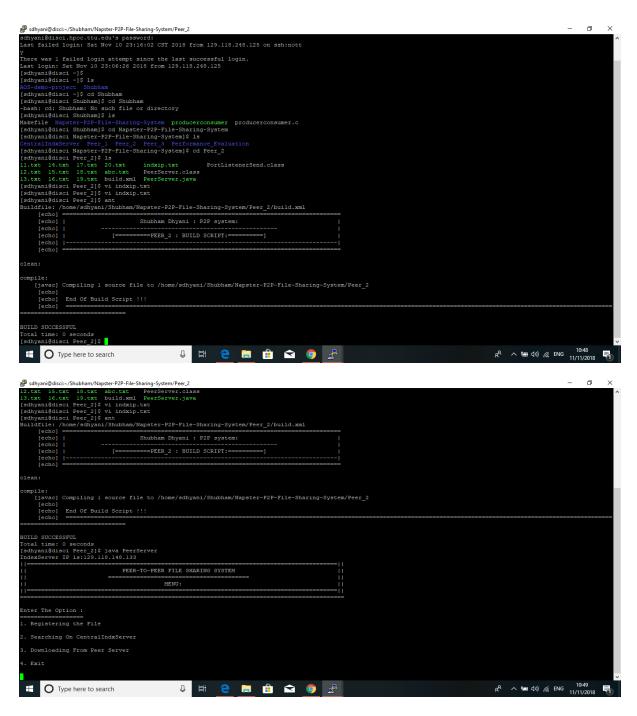
1.txt 3.txt 5.txt 7.txt 9.txt indxip.txt PeerServer.class FortListenerSend.class
[sdhyani@disci Peer_1]% vi indxip.txt
[sdhyani@disci Peer_1]% vi indxip.txt
[sdhyani@disci Peer_1]% vi indxip.txt
[sdhyani@disci Peer_1]% vi indxip.txt
[sdhyani@disci Peer_1]% nowledge in
```



9. The Peer is run, and a MENU is displayed on the screen with the following options for the Peer to select:

#### Enter the Option:

- 1. Registering the File
- 2. Searching on CentralIndxServer
- 3. Downloading from Peer Server
- 4. Exit
- 10. Following shows the Peer\_2 startup



11. Following shows the Peer 3 startup

```
|QGISC| Feer_3|$ vi PeerServer.java
|QGisc| Peer_3|$ ant
|e: /home/sdhyani/Shubham/Napster-F2P-File-Sharing-System/Feer_3/build.xml
                            Shubham Dhyani : P2P System:
          Compiling 1 source file to /home/sdhyani/Shubham/Napster-P2P-File-Sharing-System/Peer 3
otal time: 0 seconds
sdhyani@disci Peer_3]$
                                           Q # C II C II C II
                                                                                                                                   g<sup>Q</sup> Λ 🖅 Φ)) /ς ENG 13:51 🖥
Type here to search
     nloading From Peer Server
```

#### 12. Registration Screenshots -

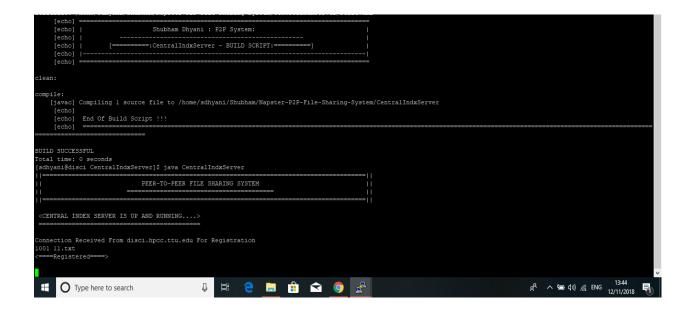
If for example the Peer\_2 wants to Register the file, it can choose option 1.

Enter the String in Format: 4Digit id and File Names separated by Space

Then the Peer 1 is Connected to Register on CentralIndxServer on port 2001

Below screenshot shows the text file 11.txt is being registered to the Central Indexing Server.

13. Below screenshot shows the message displayed of successful registration of 11.txt on the Central Indexing Server.



- 14. The Peer can also register multiple files
- 15. Next is the screenshot for the search performed by Peer\_3 for the text file 11.txt

  The result tells us about the PeerID at which the file is present

The Central Indexing Server also displays the message of the search operation performed by Peer\_3 as shown below -

16. **Downloading a file** – Screenshot below shows one peer downloading a file from another. Here Peer\_2 requests the CIS for downloading the file 25.txt. File 25.txt was already registered to the CIS by Peer\_3. The CIS establishes the connection directly between Peer\_2 and Peer\_3. Screenshots below shows the process and the results.

```
Enter The Option:

Enter The Option:

Enter The Option:

Enter The Option :

Enter The Option for CentralIndxServer

3. Registering the File

2. Searching On CentralIndxServer

3. Downloading From Peer Server

4. Exit

3

Enter Peer id:
1001

Enter peer IP Address to download file:
129-118-104-148

Enter the File Name to be Downloaded:
15-txt

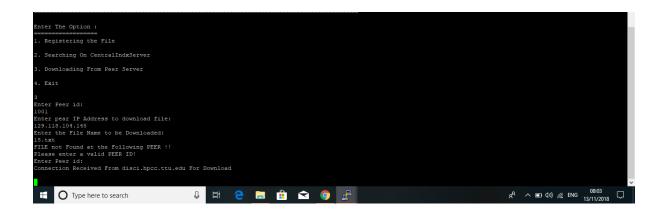
FILE not Found at the Following PEER !!
Please enter a valid PEER ID!
Enter Peer id:
3001

Enter peer IP Address to download file:
129-118-104-148

Enter the File Name to be Downloaded:
25-txt

Connected to peerid: 3001

25.txt: Downloaded
```



# THANK YOU SHUBHAM DHYANI