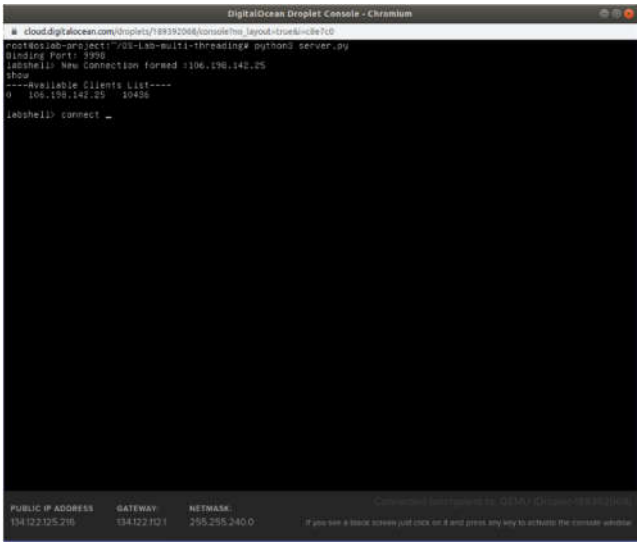
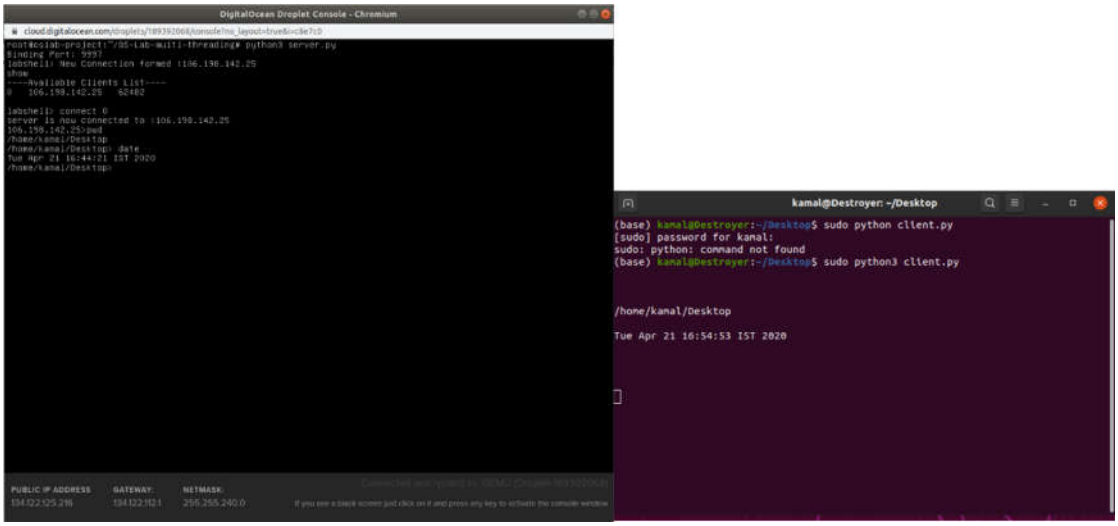
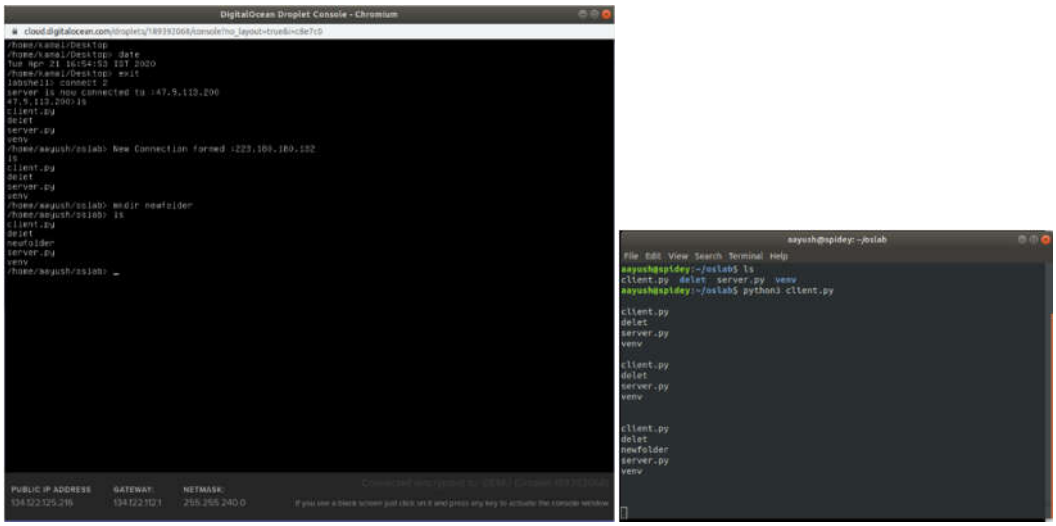
Output screenshots



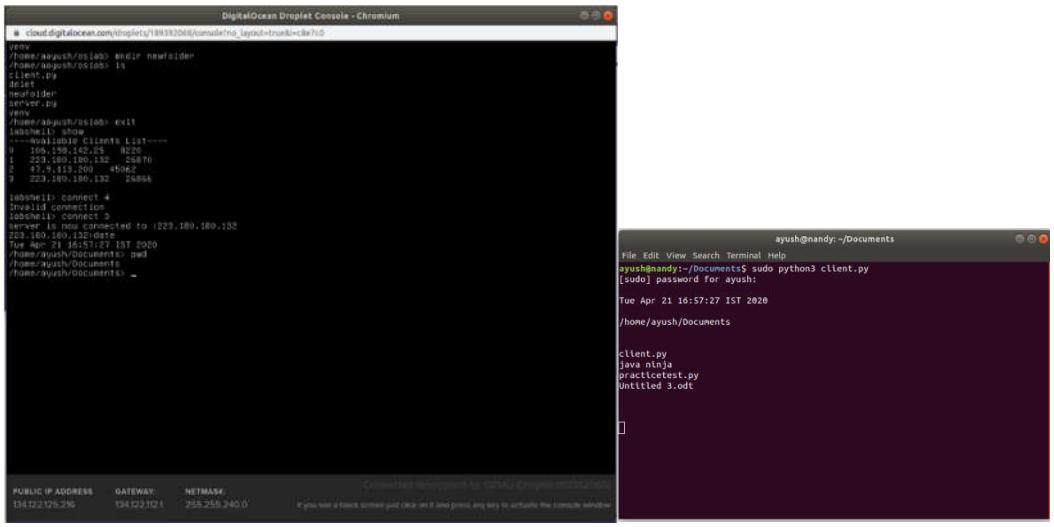
Starting up Online Server.py



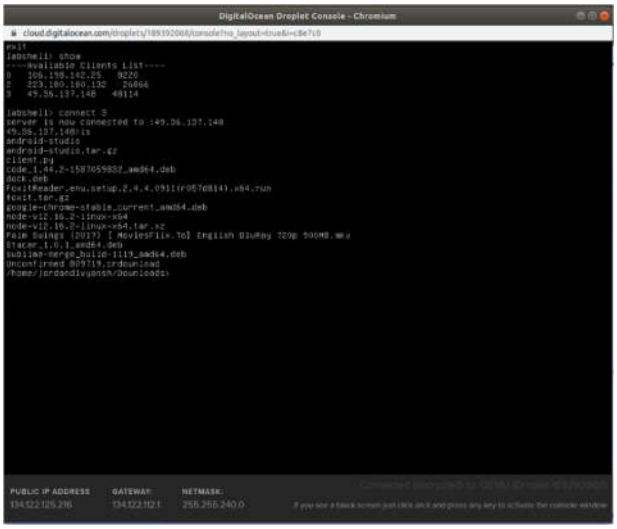
Accessing first client's Operating System



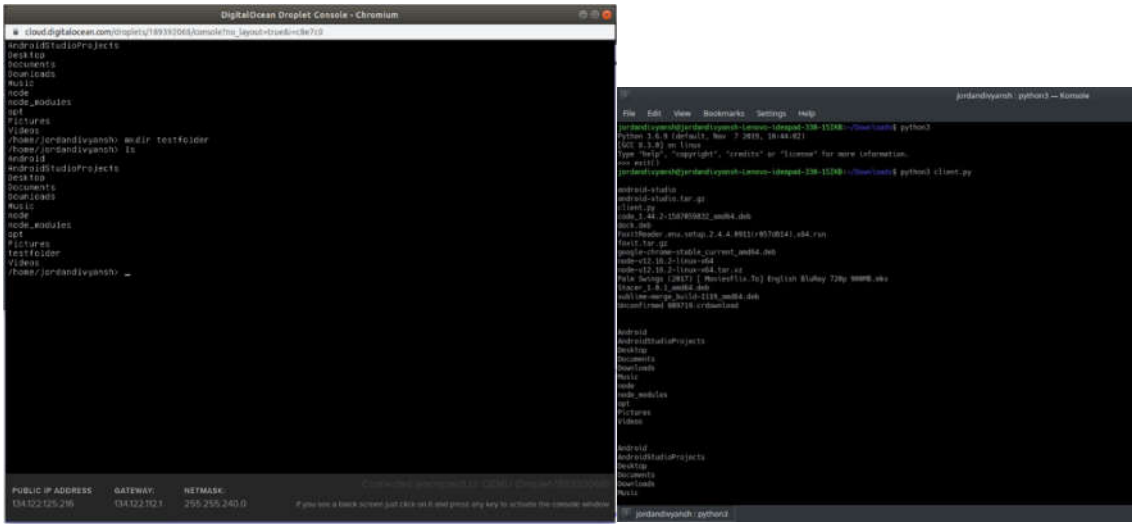
Adding more clients while accessing first client (Multithreading)



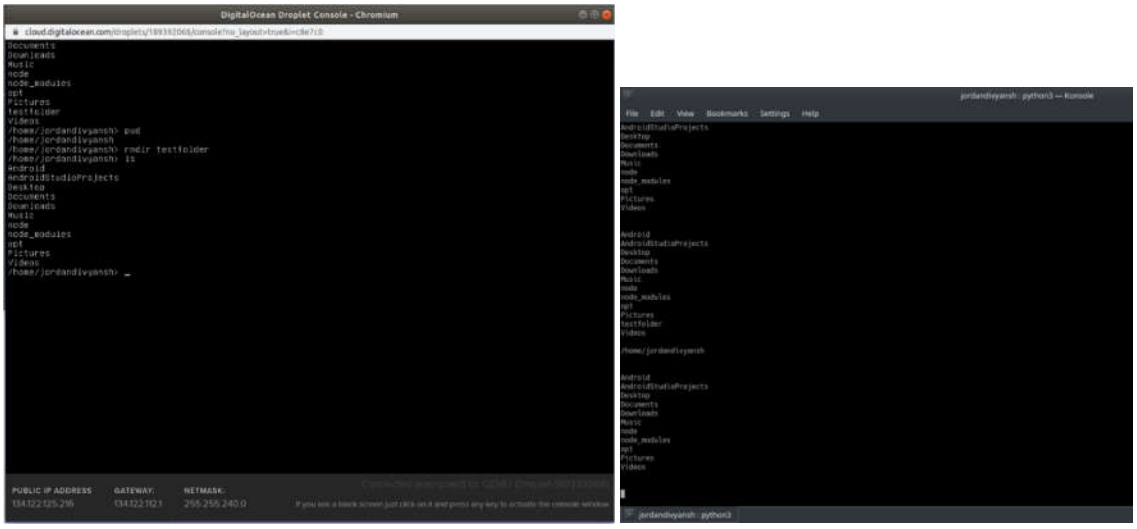
Making changes to third client’s PC



Listing all the connected clients



Accessing fourth client's files from server



Conclusion

This project was full of oppurtunities, we got to learn and implement new concepts of Operating Systems and Socket Programming. We polished our programming skills and were able to make this project successful by completing our objective of implementing the concept of multithreading in a web server which could simultaneously accept requests from new clients without halting the operations performed on the target client. The python server shown was hosted on a public ip-address which could have been accessed from any part of the world, this was done using an online cloud service called digital ocean. The project is fully functioning and it can be run on any system by following a set of steps.

References

- https://realpython.com/python-sockets/
- https://docs.python.org/3/library/socket.html
- https://www.geeksforgeeks.org/socket-programming-python/
- https://www.geeksforgeeks.org/socket-programming-multi-threading-python/
- Stack Overflow
- https://realpython.com/intro-to-python-threading/