Project 1 | Phase 2

DISTRIBUTED COMPUTING

Dhruval Darji

Deion Anthony

Mirela Memic

Table of Contents

[Introduction: 3](#_Toc433835594)

[Design Approach: 4](#_Toc433835595)

[Design Modules and Implementation: 5](#_Toc433835596)

[Client Sample Output: 8](#_Toc433835597)

[Collected Data: 14](#_Toc433835598)

[Conclusion: 16](#_Toc433835599)

# Introduction:

For Phase 2 of this project we designed and implemented a Peer-to-Peer system. The purpose of this phase was to design a Peer-to-Peer system that allows as many pairs of nodes as possible to exchange messages. As in Phase 1, a message that is sent in lowercase is to be converted to uppercase and returned. For the implementation of our Peer-to-Peer system, we modified and used our Client-Server programs from Phase 1. We used four main modules in the implementation of our system; TCPServerRouter, SThread, TCPServer, and TCPClient. We also used our custom statistics module, called Stats, to monitor and keep track of variable data, and our main module, called Main, to dynamically run the system. More detail on each of the modules, and how they were modified from Phase 1, will be found in the Design Modules and Implementation section.

# Design Approach:

We began Phase 2 of this project with our programs from Phase 1. The basic function of our system in this phase is the same as in Phase 1; to send a message, in the form of a text file, in lowercase, convert it to uppercase, and return it. With this in mind, we simply used the TCPServerRouter, SThread, TCPServer, and TCPClient modules from Phase 1 and modified them from the Client-Server system to a Peer-to-Peer system. We also used our Stats module for data collection and our Main module to run our system. In addition to the modules we used from Phase 1, we also added another module called RoutingInfo to hold and use all of the necessary router information. As in Phase 1, we used IntelliJ IDEA as our IDE and the GitHub remote storage in order to easily and efficiently make changes to and share all of our project files between each group member.

# Design Modules and Implementation:

# Client Sample Output:

# Collected Data:

# 

# Conclusion:

In Phase 2 of this project we implemented and analyzed a Peer-to-Peer distributed system that allows multiple pairs of nodes to exchange messages over a server-router bridge. By incorporating our programs from Phase 1, we were able to see the similarities and differences between the Client-Server system and the Peer-to-Peer system. The completion of this phase of the project has provided us with more knowledge and a better understanding of the Peer-to-Peer paradigm.