

Jeff Foreman, Nimitt Tripathy

CSE 160 Project 3 Discussion Questions/Write Up

November 18th 2017

PROJECT 3 DESIGN

There are two halves to our Project, branching off of the completion of Project 2: *TransportP* (along with its respective headers) and *Node.nc*. For this project, we are dealing with the transport layer in which we define sockets for a node where each socket is given a state, port, address, flag, and read/write buffers. These sockets send and receive SYNACK, FIN_SENT, and FINEND, and data packets. These packets are utilized for three-way-handshakes along with the actual distribution of data throughout the network. In this project, a socket will send a SYN towards a certain socket, where the socket that received the SYN will reply with a SYN_ACK. Once the SYNACK has been received, the node that received the SYN_ACK will reply with an ACK that the entire connection has been established among the corresponding sockets. Now that a connection has been established, a sender node can write its data packet towards its sockets out-buffer, which will send to the socket of the receiving node. This socket will read it onto its in buffer as now the node will be able to process this data as it pleases. Once the sender node is done sending data, it will send a FIN_SENT to the receiver socket. When the socket has completed sending out its data packet, it will send a FIN_SENT packet as well. The receiver node, gets the data and reply with a FIN_RECIEVED, causing both sockets to close. We created a file called TransportP which defines the set of sockets that a node has and all of their functions such as read, write, open, close, and connect. We implemented a series of conditions in Node.nc accounting for the SYN packets being sent out and received, the ACK's being sent out and received, and the data packets. Overall, our project implementation meets the requirements of a functioning Transport Layer.