Denny Sabu

John Sweeney

Michael Parrilla

**Operating Systems Assignment 3 Report**

**Server**

The server side of our project is handled from the serverSNFS.cpp file, in which its main function takes input from the command line to set the port and mount directory path. After the port and mount directory have been set main calls the serverStart function to accept socket addresses and create socket file descriptors. New sockets are created via threads from which the whatToDo function is called. It is from the whatToDo function that all other functions are called including get attribute, create, open, read, write, close, truncate, read directory, and unlink. The function for get attribute is called do\_getAttr and it attains the relevant attributes of a socket which are written to a character array. The create function is named do\_create and it opens paths as specified for sockets. The read directory function iterates through the directory at the given path and writes all files within that directory. There is also the do\_mkdir function that is used to make directories ate the given mount path. The opening of paths is handled from the do\_open function and the closing of paths is called from do\_close, both of which need to be called to use the sockets. Then the read and write functions, do\_read and do\_write, allow for the socket file descriptors to be seen and edited. Finally do\_truncate can remove parts if a path based on the desired size while do\_unlink can unlink a socket from its path.

**Client**

All client side functionalities are handled from the clientSNFS.c file, which connects to the server’s port as is given from the command line. Additionally the main function of clientSNFS implements a struct of FUSE arguments for its operations. These operations reflect the operations on the servers and are named as follows: hello\_unlink, hello\_truncate, hello\_release, hello\_mkdir, hello\_write, hello\_getattr, hello\_readdir, hello\_opendir, hello\_create, hello\_open, and hello\_read. Each of these functions has a corresponding fuse operation that is used to call them: .getattr, .readdir, .opendir, .open, .read, .mkdir, .write, .create, .release, .truncate, .unlink. From each of these functions the startConnect function is called to access the server functions so that the client can request the respective functionality from the server.