For the first part of this project, our main work is on parsing a configuration file, generating a uuid and updating the file when needed, adding neighbors and killing the server. With regard to my design, I create 2 helper classes which are CurrentNode and PeerNode to represent the current node and its neighbors respectively. By parsing the configuration file, information about current host and default neighbors can be accessed and stored in a CurrentNode instance. If no default uuid is found in the file, a new one would be generated using functions from the library *libuuid* and the file would be updated. Then the server would listen and get command from the keyboard. For now it only supports the “neighbors” command, which would print out all neighbors of the host, the “addneighbors” command with information about a new neighbor, and the “kill” command to terminate the server. If a new neighbor is successfully added, the program would print “Neighbor added!” along with the new node’s information.

The program is written in C/C++. An executable file *contentserver* can be easily generated by the make command with a Makefile.