**S2110978**

**Mangaliso Linda Dlamini**

[m.dlamini@alustudent.com](mailto:m.dlamini@alustudent.com)

**Systems Programming**

**CW1 - User Documentation**

**User Documentation**

How to run

1. Server

* If you are running for the first time you will need to run the *make* command on your terminal from the server working directory to compile the source code
* Once the source code has been compiled an executable will be created on the server working directory.
* Run executable and start the server by running the command .*/server* in the server working directory.
* On start the server will listen for client connects on port 50031
* Exiting server - ctrl + c will close the server

1. Client

* If you are running for the first time you will need to run the *make* command on your terminal from the client working directory to compile the source code
* Once the source code has been compiled an executable will be created on the server working directory.
* Run executable and start the server by running the command .*/client* in the client working directory. Ensure that the server is started on a separate terminal window before starting the client.
* Starting the client will connect to the server and display a menu and prompt to enter the menu option
* Select option 1 to 6
* Exiting client
  + select option 6 on the menu
  + Ctrl + c

The program implements a text based menu interface that connects a client to a server to perform specific tasks. Each task is performed by selecting the corresponding option on the menu by entering a number between 1 and 6 at the prompt.

1. Returns the server IP address prefixed to the author's name and student ID.
2. Returns an array of random numbers between 100 and 500 generated by the server and whose sum is calculated by client
3. Returns the server’s system information or uname
4. Returns the list of files in the server’s ‘upload’ directory
5. Prompts the user to specify a file and copies it from the server's ‘upload’ directory to the client’s working directory.
6. Exits the clients

All menu options work as expected and also include error handling for cases where invalid menu option is selected, ‘upload’ folder does not exist or when the specified file to copy is not found in the upload directory. The upload directory must be a sub directory in the server’s working directory name “upload” The server implements gracefully exiting on receipt of SIGINT or ctrl + c which also work as expected. The server can accept multiple client connections. Identifying clients connections and communicating with the server was explored but not fully implemented.