



**School of Computing**

**Bachelor of Science in Computing.**

**Programme Code: DT211/3**

**2012 – 2016**

Network Programming Testing Document

**Lecturer: Mark Deegan**

<b>Student's Name</b>	James Wilson
<b>Class Group</b>	DT211/3
<b>Assignment Number</b>	2
<b>Assignment Title</b>	Simple Server
<b>Date Issued</b>	17/1/2015
<b>Date Due For Return</b>	6/3/2015
<b>Date Returned</b>	6/3/2015

## Testing Document

I will be testing the outcome of various different types of inputs from command line by the user, in the Simple C & Java server, and C & Java client. I am testing for valid arguments that have been passed through the console, if the arguments are not a length of one then print an error. I also will be testing to see if the server is valid and if the ports are available to establish a valid connection between the client and the server.

We should also be testing for the co-operation between the client and server in the two different languages provided by the developer.

## Test Case Narrative

### Testing valid arguments

<b>Test Case Name:</b> Test the input from command line via the client
<b>Intent:</b> The client should be prompted with an error message
<b>Precondition:</b> Enter in an invalid number of arguments
<b>Example:</b> 44 or " " or localhost
<b>Dialog:</b> Step 1: Client (user) enters in the host and port number Step 2 : User hits the enter button
<b>Expected Results:</b> The Client should be able to write to the socket
<b>Test Case termination:</b> Hitting enter after entered the hostname and port number

### Testing if there is a valid server online

<b>Test Case Name:</b> Test to see if there is a server to connect to
<b>Intent:</b> The client should be see an error message
<b>Precondition:</b> Enters in correct credentials but there is no server
<b>Example:</b> localhost 4444
<b>Dialog:</b> Step 1: Client (user) enters in the host and port number Step 2 : User hits the enter button
<b>Expected Results:</b> The client should be noted that there is no server to connect to
<b>Test Case termination:</b> Hitting enter after entered the hostname and port number

### Check if no arguments were passed on server side

<b>Test Case Name:</b> Test to see if the server has entered a port number
<b>Intent:</b> the server should be notified invalid number of arguments where passed through
<b>Precondition:</b> Enters in correct credentials but there is no server
<b>Example:</b> " "
<b>Dialog:</b> Step 1: The server enters in a blank argument Step 2 : Hits the enter button
<b>Expected Results:</b> The server should see that there was no arguments entered
<b>Test Case termination:</b> After hitting the enter button and waiting for a reply from the console

### Check if too many arguments were passed on server side

<b>Test Case Name:</b> Test to see if there is a server has entered in more than one port number
<b>Intent:</b> The server should be issued an error
<b>Precondition:</b> Enter's in too many port values
<b>Example:</b> 4444 4444 4543
<b>Dialog:</b> Step 1: Server enters in too many port numbers in the console Step 2 : Server hits the enter button
<b>Expected Results:</b> The server should be noted that too many arguments were passed through
<b>Test Case termination:</b> Hitting the enter button after inserting port values

### Test that C Client can work with the Java Server

<b>Test Case Name:</b> Test to see if the C Client can connect to the java server
<b>Intent:</b> The client should be prompted to enter a message & the server should be able to check to see if they received a message from the C client
<b>Precondition:</b> Correct credentials are in place between the c client and the java server
<b>Example:</b> C Client: localhost 4444 Java Server: 4444
<b>Dialog:</b> Step 1: Server enters in the specific port number Step 2 : The client enters in the server to connect to and the same port number
<b>Expected Results:</b> The client and server should be able to work together

**Test Case termination:** Hitting the enter button after inserting port values