Chris Klassen – Melvin Loho – Jonathan Chu – Joel Mabbott

BCIT  3700 Willingdon Avenue

Windows assignment 4 design document

Table of Contents

[Requirements 2](#_Toc411875591)

[State Flow Diagrams 3](#_Toc411875592)

[Server Pseudo-Code 4](#_Toc411875593)

[Client Pseudo-Code 4](#_Toc411875594)

# Requirements

* Create a Windows application
* Use the TCP/IP protocol suite to transfer audio from a server to clients
* Play audio from the clients
* Client must be able to connect to a remote server
* Client must be able to save and retrieve sound files from the server
* The .wav format must be supported
* Two-way microphone support must be provided
* The server must use multicasting

# State Flow Diagrams

# Server Pseudo-Code

## Initialization

Main Function

{

Call the Setup Function

Call the Load Tracklist Function

Call the Open Socket Listener Function

If the mode is multicast

{

Call the Multicast Function

}

Else

{

Call the Unicast Function

}

Call the Teardown Function

If the Teardown Function fails

{

Print error message

exit

}

exit

}

Setup Function

{

Parse the command line arguments

If the command line arguments are invalid

{

Print the usage string

Return false

}

Set the mode to argument 1

Set the port to argument 2

Start Winsock

Return true

}

Load Tracklist Function

{

If the music folder does not exist

{

Print error message

Return false

}

If the number of audio tracks is zero

{

Print error message

Return false

}

For each audio track in the music folder

{

Add the file name to a music list

}

Return true

}

Open Socket Listener Function

{

Open a listener socket with the specified port and protocol

If the socket fails to open

{

Print error message

Return false

}

Bind the listener socket to an available port

If the bind fails

{

Print error message

Return false

}

Return true

}

## Teardown

Teardown Function

{

Close the control channel

If the control channel fails to close

{

Print an error message

Return false

}

Stop Winsock

Return true

}

# Client Pseudo-Code