Marist College

Server Client Network Programming in Java

MileStone

Professor: **Juan Arias**
By-
Prabuddha Banerjee

Milestone

Abstract
The Project is dealing with Networking as per the proposal. In this we deal with sending of Audio files via Server and client socket network. I wanted to make a communication system in which the messages are communicated among clients securely.

Introduction:
I decided to do this networking project because I am very much motivated in learning new things during my course work of Masters and this would be very good opportunity to learn new thing under the guidance of professor.

This project is focused on implementing an infrastructure that allows the distributed program components to communicate over a network in a reliable, efficient and generic way. The goal of the mechanism is to hide the distributed nature of remote objects as well as maintain privacy of the files which are sent. In particular, this project will communicate between clients via server network port and in order of communication it will maintain the privacy of message by encoding the file.
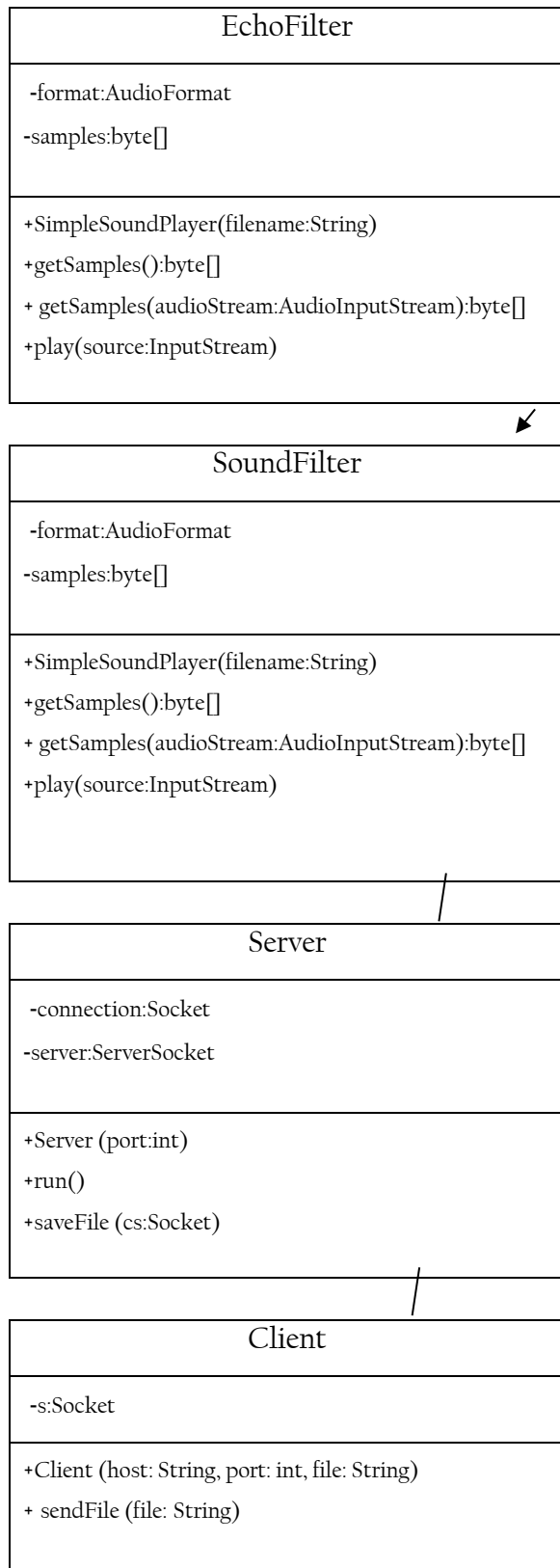
Detailed System Description

The system deals with maintaining transparency which is an important parameter in software design, it could simplify the end user's way of sending messages securely. In this project up till now I have created methods to encrypt the sound files and to decrypt it. Further I also built a method to send messages between client and server.

SimpleSoundPlayer
-format:AudioFormat   -samples:byte[]
+SimpleSoundPlayer(filename:String)   +getSamples():byte[]   + getSamples(audioStream:AudioInputStream):byte[]   +play(source:InputStream)

For the final write up, try to explain what these classes are for.

FilteredSoundStream
-format:AudioFormat   -samples:byte[]
+SimpleSoundPlayer(filename:String)   +getSamples():byte[]   + getSamples(audioStream:AudioInputStream):byte[]   +play(source:InputStream)

EchoFilter

-format:AudioFormat

-samples:byte[]

+SimpleSoundPlayer(filename:String)

+getSamples():byte[]

+ getSamples(audioStream:AudioInputStream):byte[]

+play(source:InputStream)

SoundFilter

-format:AudioFormat

-samples:byte[]

+SimpleSoundPlayer(filename:String)

+getSamples():byte[]

+ getSamples(audioStream:AudioInputStream):byte[]

+play(source:InputStream)

Server

-connection:Socket

-server:ServerSocket

+Server (port:int)

+run()

+saveFile (cs:Socket)

Client

-s:Socket

+Client (host: String, port: int, file: String)

+ sendFile (file: String)

[Handwritten note:] Are these two different classes? What's the difference? Maybe there should be a Filter parent class and these two as sub classes.

Requirements

The system is dealing with the modernizing communication facilities that we could use for maintain privacy among people. For the user to use the system one must have sound files in WAV form. The program will not work for any other format of music.

Literature Survey:
The other websites like java-forums.org have dealt with sending just sound files between clients and servers but they either seem to not cover enough or seem to focus on the sending sound file aspect too much to actually teach anything, and the system lack to be secured while sending/communicating between utilities.

User Manual:

For starting up with the system user must have any sound file in computer with WAV format. Then the user need build a connection between server and client. So firstly the user needs to run server class and then the client class to build the connection. Then the user may give the path for the file where the voice/sound file is stored. The user may see that the file got modulated (or encrypted in our case). Then the file will be sent back to client in decrypted way.

Conclusion:
The way this program should be used is one must have WAV file at the beginning. Then the user must know what the location of the sound file is. The user will then build connection and the file is communicated. The project is yet in its last stage to complete as up till now a connection is built between server and client and the file which needs to be transferred is modulated and ready to be sent. So, just a method that needs to be set up for sending of the file via server and client network.

References:
https://stackoverflow.com/questions/17044644/sending-audio-stream-over-tcp-unsupportedaudiofileexception

https://www.youtube.com/watch?v=-xKgxqG411c

Introduction to Java Programming And Data Structure by Y. Daniel Liang