SOFE4790 Distributed Systems (Fall 2020 - Dr. Q. Mahmoud)

Assignment #1

Honour code: By submitting this assignment report, I (name and banner ID# below) confirm this is my own work, and I have not asked any of my fellow students or others for their source code or solutions to complete this assignment, and I have not offered my source code or solutions for this assignment to any of my fellow students.

Name: Umar Qureshi

Banner ID#: 100591742

1. Application idea

My Client-Server Application is a .WAV File editor where the client inputs information of what needs to be edited in a GUI and the .wav is sent over to the client to be edited and finally the edited file is received. This idea was inspired by Dr. Q. who stated in a lecture about how someone edited an image, I thought of doing video but that seemed like too large of a task, so I went one level down to audio. It was also inspired by VideoEditBot on twitter which is a versatile bot where all the client needs to do is tag the bot and issue a command and the bot on Server side will edit the videos speed, pitch, resolution, replace the audio with a song from YouTube and even can place text on top of the video among many other features. It is an excellent example of a Client-Server application and it showcases how my simple program can be of use, can be scaled up, more features and polish can be added.

Commands for VideoEditBot:
https://github.com/GanerCodes/videoEditBot/blob/master/COMMANDS.md

2. Describe the two core functionalities

i) **Trimming the .WAV File.** A user on the GUI types in the portion of the audio they would like to keep, they type in the start and end time.

ii) **Altering the audio of the WAV File.** Clicking on the "spooky" radio button alters the sound of the Wav file, this was done by changing the Sample Rate of the audio file.

You are also able to use both functionalities at once

3. Describe the two novel features

i) **Simple GUI which doesn't close, can send multiple requests and has an easy to use file selector.** The GUI only closes when the user exits so it can send one request after another. The File Selector also only allows users to choose .WAV files

ii) **Audio Player within the GUI.** The user does not need to leave the program to listen to the edited audio, they simple select the edited file or any .WAV file to listen to.

4. Challenges and solutions

I had not anticipated how difficult this idea I came up with was. My original vision was to have more ways one could edit the WAV file such as increasing and decreasing amplitude/volume, speed, pitch and adding interesting filters on top of sounds. I spent hours searching, I was able to achieve most of them as I could hear them while they ran but there was no way for me to save the effects into a new .WAV File. I do not have much of a solution, some code I found I edited the Sample Rate and it altered how it sounded, it felt like the effect where you are outside and hearing music from inside a building. In the interest of time as my efforts ended up fruitless, I dubbed it the spooky sound effect and made it my main Feature. I think it still counts as a core functionality as the concept of editing how a file sounds are the same, Java sadly isn't built for editing and saving WAVs with effects.

When writing code to send the file back from Server Side I was having issues with it. I believe my problem was that I was using the same socket to send back the file which I used to receive it so what I did to solve this was to use a new socket and port number and pretend my Server was a client and I sent it back just like how the Client sent it to me, of course the client also received it as my Server did.

I had lots of issues with path names not working as I intended. I had saved my Server file in a different folder to showcase that my program is truly running on sockets and to keep things organised. To solve this, I found functions for path names and with a little of trial and error I found out how to do it correctly.

I have never used a Java GUI before, so it was a nice challenge to take on and use the opportunity to learn it. I watched a few videos on it and implemented the basics learnt with the video along with google for any other additions. Even with my simple, unpolished GUI I think that it does the job well.

When a file was selected the filepath that was saved was locked in the function, I was not able to use it in another function. I decided to make a file object that was global so I could easily save it and use it anywhere. I made other variables global as well to make life a bit easier.

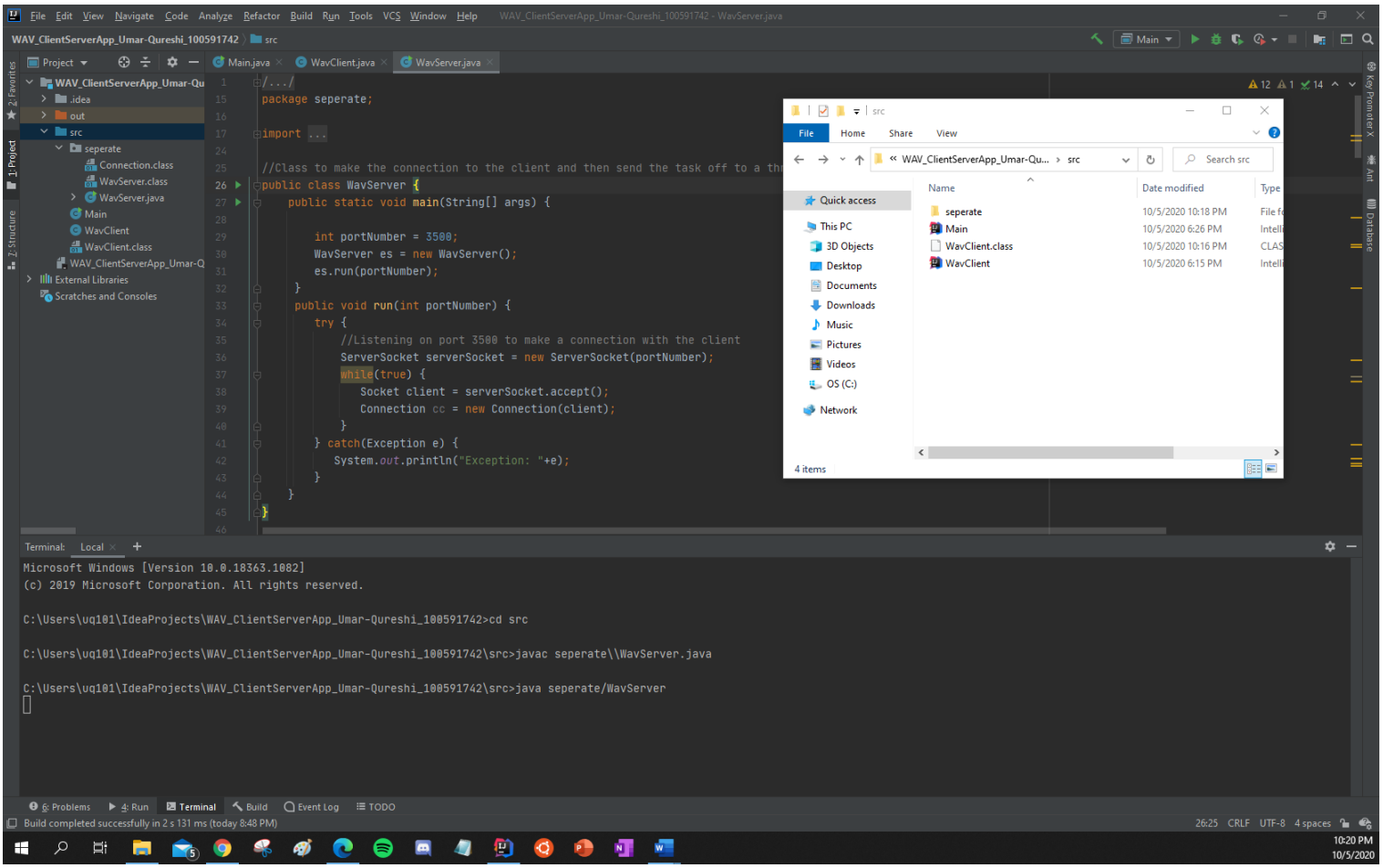
When I ran my GUI I was only able to save one file, If I saved another it would give me some sort of socket already in use message. I looked into it more deeply and realized that I had not closed my Socket or Buffer reader correctly, once I did I was able to save more than once except for one situation

When I would run the GUI, put on spook filter, save a WAV file, then turn off spook filter then try to save the new Wav file my program would crash. It gave me an error message stating a certain file had not been found when I specifically coded for a file to rename into the missing file. I decided to take a simpler approach where no matter what the spook filter was applied and then with an if statement, I sent the normal file or spook file as opposed to renaming files.

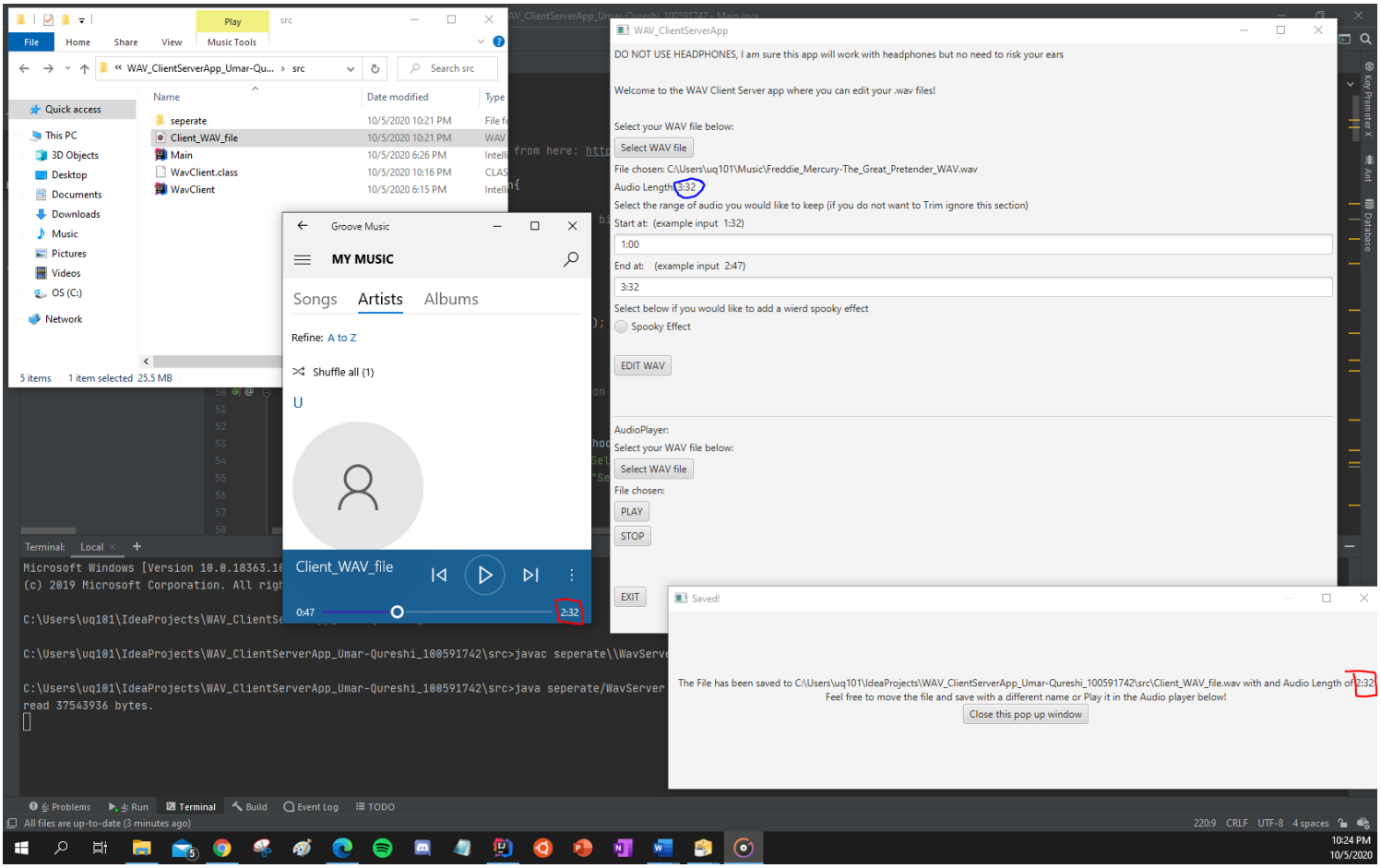
Overall this was a great learning experience as I am not too familiar with java, I got a chance to learn JavaFX GUI and get a better grasp on object oriented programming and of course hands on understanding of sockets.

5. Testing

Describe each test and include screenshots

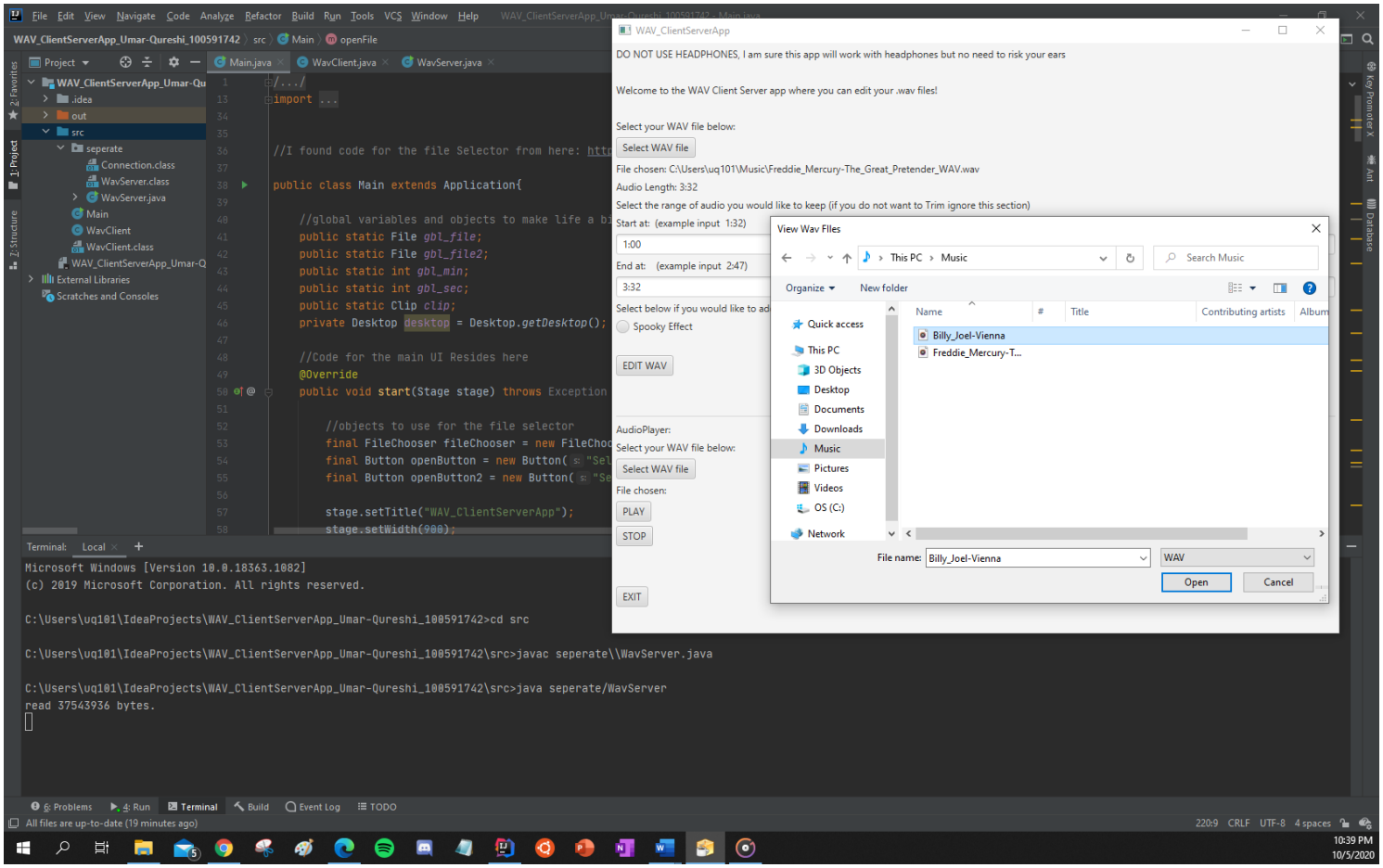


^Opening the Server and showing no WAV Files in the folder

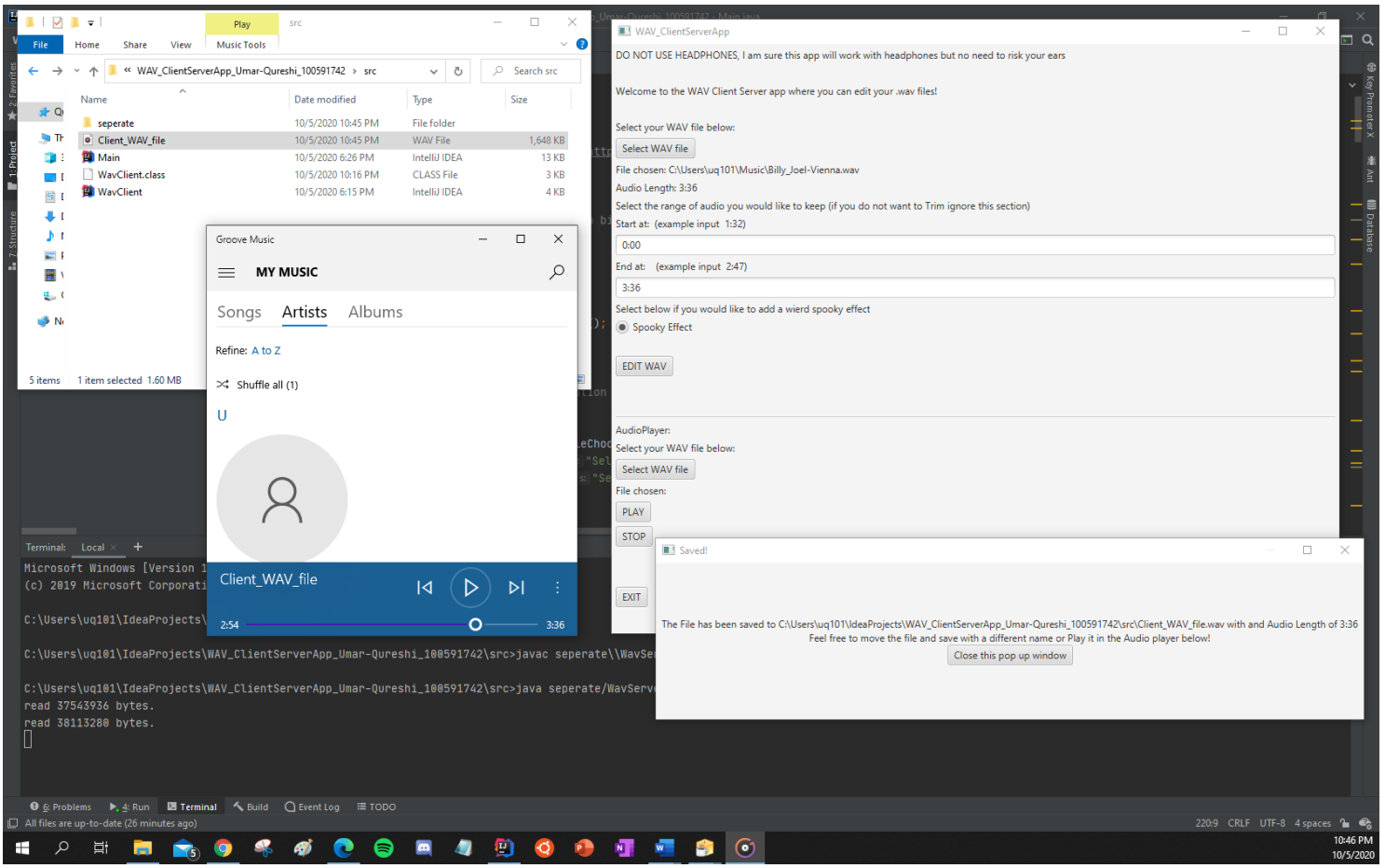


After running Main, entering info and clicking EDIT WAV, you can see the file has appeared and is one minute shorter as you can view in the red squares compared to the blue circle. You can also see Server outputting how many Bytes it has read.

Now lets Add Spooky effect to the song Vienna, first lets look at how the File Select Screen looks like:

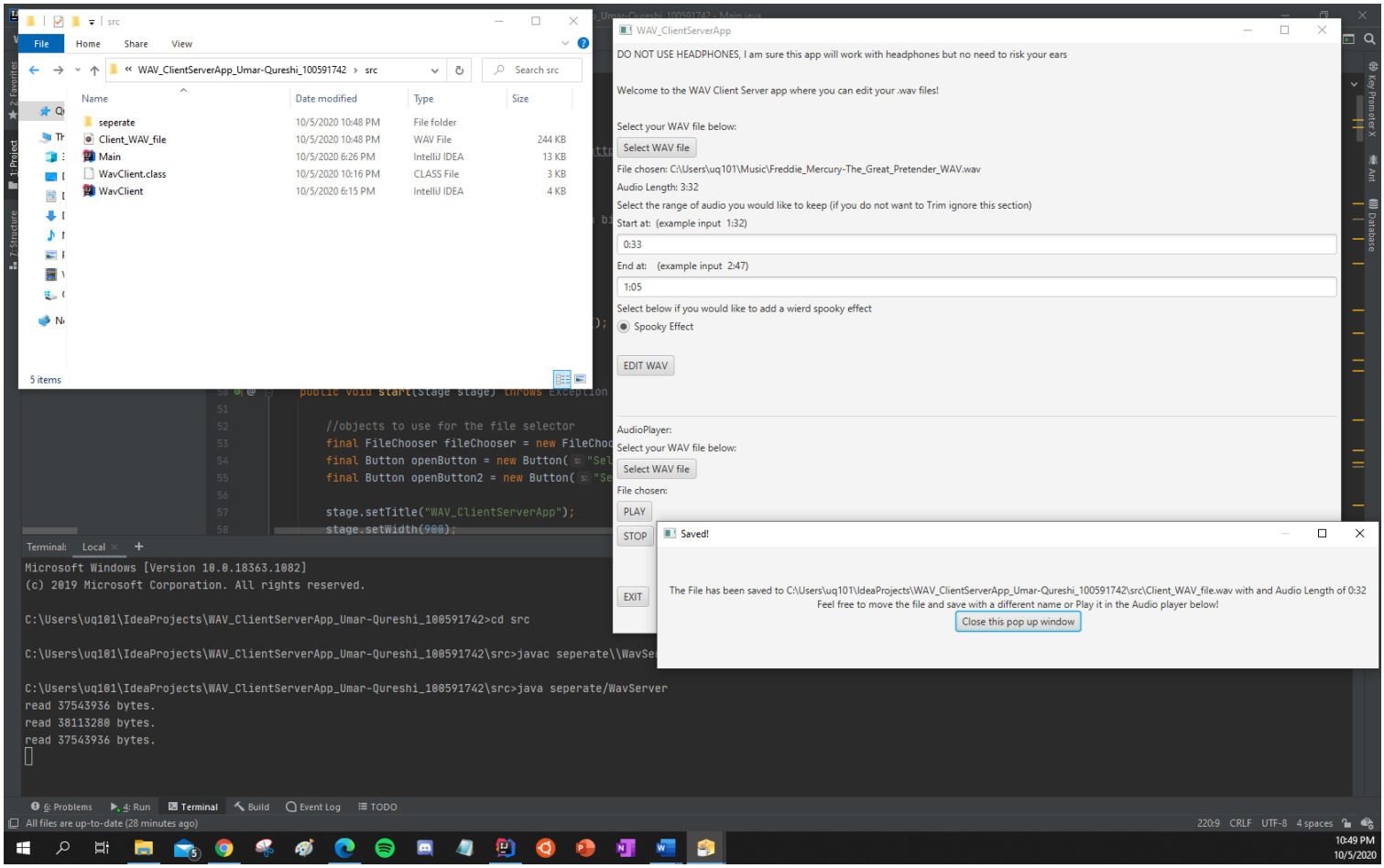


As you can see it only views and opens WAV files. Now lets add the effect.



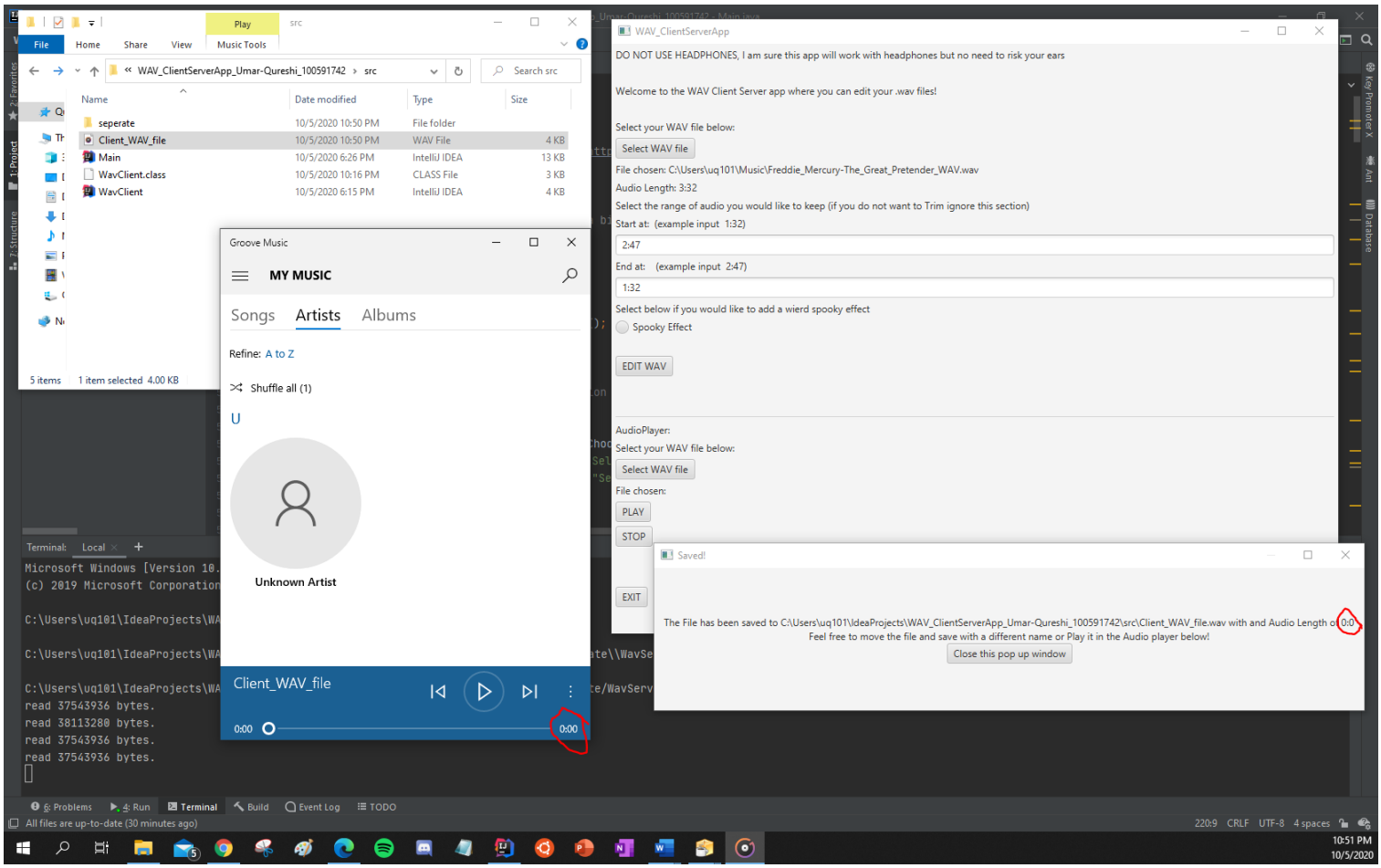
Effect has been added, although you cannot hear it.

Now lets trim and add spook effect, both at once:



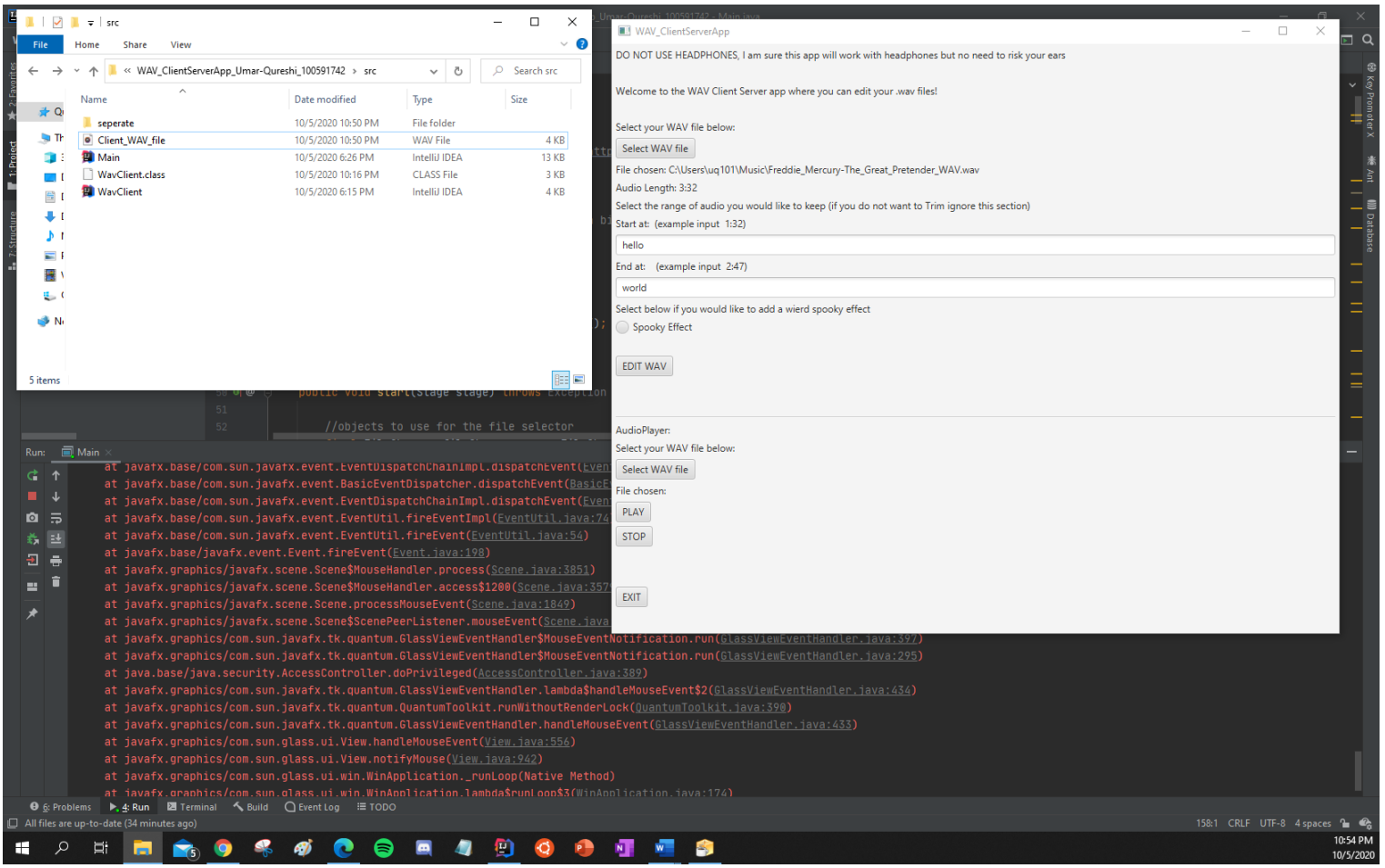
Notice the difference in file size in top left. Notice on bottom Left I have still not left the client yet

What happens if a user uses a bigger start then end:

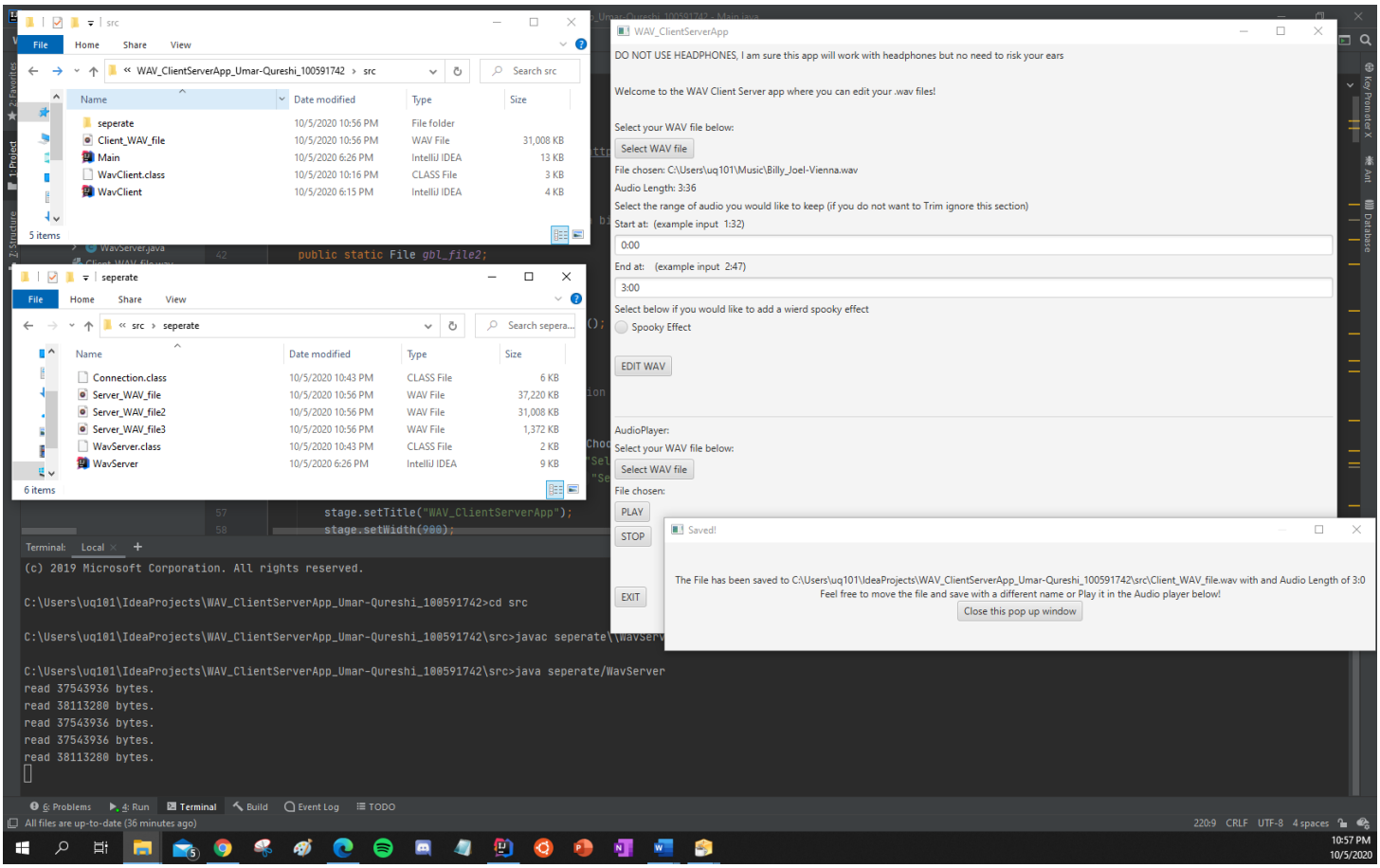


It saves but the music file has no length as seen by the red circles, program does not crash.

What if the user enters a string:



It does run WavClient and gives an error in run menu but does not crash the program/GUI



As you can see even after entering a string I am still able to use the GUI as I trimmed one more file. In the screenshot you can also look at what is stored in the separate folder

For the Audio Player you can see how I have selected the the most recent trimmed file, when you press play it play from the start and pressing stop stops the music. (Just note all these screenshots were taken without leaving the GUI at all)