What we Achieved

during this group project we have achieved many new things such as

- creating a web engine with Java and java.fx
- separating the classes instead of having one block of code
- fully functioning prototype
- Simple UI

In order to create the volume mixer we had to install the latest version of java.fx which is “javafx-sdk-17.0.13” on all on our machines in order for it to work for everyone and enter the right configuration file on the launch.json file for it to work e.g "vmArgs": "--module-path \"C:\\Users\\dhavish\\Desktop\\java\\javafx-sdk-17.0.13\\lib\" --add-modules javafx.controls,javafx.fxml,javafx.web" with the right project main class name too. By installing this on VS code and importing it. We was able to start implementing our code into one big block of code which then we was able to launch our Gui project with a working web engine such as YouTube Vimeo and SoundCloud using java and JavaFX.

separating the classes instead of having one block of code

Once we had a fully functioning web engine, we then separated the classes into different files for more efficient work. The examples of the classes we separated are SoundCloudMixer, VimeoMixer, YouTubeMixer, VolumeSlider, and CombinedVolumeMixer, which extends the other application. Each class has its own individual purpose, such as Vimeo, YouTube, and SoundCloud. They have three separate web engines that link to YouTube, SoundCloud, and Vimeo, which are fully functional and working. The VolumeSlider class controls all the audio for all the web engines, including thickness and size, by giving each mixer class a tag called "player", e.g., getElementById('player'), which controls the volume for each individual web engine. The CombinedVolumeMixer class is the graphical user interface (GUI), which contains the entire layout. It extends the other classes, e.g., YouTube, Vimeo, and SoundCloud, and creates and displays the scene for all three web engines.

fully functioning prototype

We managed to get all the web engines to have full functionality by being able to control each web engine individually using the ID "player". We were able to choose whatever song or video we wanted on the web engine using the src link and changing the video ID link number for that specific video. We were able to put each of the web engines side by side without affecting the control of the volume or having very poor positioning of the web engines.

Simple UI

We managed to achieve a very vivid friendly user UI as all the web engine is next to each other in a orderly manner. We changed the colours of the display/scene to make it visually appealing for the audience. We made the button slider to a suitable size for its length and width and made the thickness of the volume percentage numbers more acceptable.