Project #1 - Audio Control System - Media Player

This project must be done alone. No groups allowed.

Program **Course**	Game Development - Advanced Programming INFO - 6046 Media Fundamentals - Fall 2023
Duration **Due Date**	**2 weeks** **October 15, 2023 11:59pm**
Total Marks **Weight**	70 Weight: 12-15% (Based on the number of projects (4-5)

Project Description and Purpose

The purpose of this project is to develop a system that allows users to efficiently manage and control audio playback in a multimedia application. This system is designed to provide a user-friendly interface for interacting with audio elements, enabling users to play, pause, stop, and adjust pitch, volume and pan of audio sources.

Make sure the audio files you submit are *appropriate* for an academic environment. When developing this project consider the possibility of adding it to your professional portfolio.

Note: These are maximum marks that can be possibly acquired for each section.

Note: Marks will not be awarded for a submitted project that does not meet the minimum criteria of the project which does not obviously demonstrate the purpose of the project.

Project Requirements

FMOD Bootstrap with C++ (15 marks)

- Integrate FMOD audio engine into your project and initialize it using C++.
- Properly initialize FMOD audio engine in your project.
- Provide error handling for the FMOD initialization and all calls to the FMOD api.

AudioManager that communicates with FMOD (25 marks)

- Implement functions to load and unload assets (sampled, and streaming sounds)
- Create methods to play, pause, stop, set pitch, set volume, and set pan for Audio Sources on their channel
- Implement error handling for audio resource management
- Establish communication with the FMOD engine for audio playback and manipulation
- Appropriately handles the audio resources.

MediaPlayer that communicates with the AudioManager (15 marks)

- Implement methods to control audio playback:
    - Play Audio
    - Pause Audio
    - Stop Audio
    - Adjust Pitch
    - Adjust Volume
    - Adjust Pan
- Ensure the MediaPlayer communicates with the AudioManager for these operations
- Implement error handling for user input and audio control

Controlling the MediaPlayer with User Input (15 marks)

- Create a user-friendly program that allows the user to interact with the MediaPlayer using various input methods.
- Design a graphical user interface (GUI) using an OpenGL project or use a command-line interface (CLI) for user interaction.
- Display controls for the user in the GUI or CLI.
- Incorporate user input to trigger MediaPlayer actions:
    - play, pause, stop, adjust pitch, volume, and pan
- Display relevant information such as currently playing audio, volume levels, and status
- Provide feedback for the users for successful or failed actions
- Ensure the program integrates with the MediaPlayer and AudioManager seamlessly.

Bonus

- Display the current playback position of the current audio source being played with a Bar that transitions across. (Not just the position as a value)

Submission Requirements

- Provide clear and concise documentation for the system.
- Ensure your code follows a consistent coding style, and naming conventions.
- Write a ReadMe describing how to build & run your project, and include what buttons to use to use your MediaPlayer.
- Properly document your variables, functions and classes.
- Include all required source files, resource files, and documentation within your project, and zip the project up labeled as *FIRSTNAME_LASTNAME_info6046_project01.zip* and upload to the appropriate project submission folder on FoL.

Plagiarism

While you may freely “borrow” mine (or anyone other) code but your code should be sufficiently” different.In other words, you cannot simply use an existing game engine (or part of a game engine) to complete this assignment; it should be either completely new or “significantly” modified.

Grading Scheme

- Normally a grade of zero (0) will be assigned to any assignment that is submitted late.
- However, certain rare exceptions apply according to the Infotech Policy on Missed Evaluations and Evaluation Deadlines.
- If you code does not even compile, I will not mark it. Period. This will get you a mark of zero (0).
- If you code does not build (i.e. linker error) and run (i.e. no crazy run-time crash that is unexpected), I may investigate this further, but only if there is some simple problem and/or very slight and/or very obvious (and easy to fix) configuration error.

Project Corrections

If any corrections or changes are necessary, they will be posted to the course web site and you will be notified of any changes in class. It is your responsibility to check the site periodically for changes to the project. Additional resources relating to the project may also be posted.