Project #2 - Audio Control System - Interactive 3D Scene

<u>This project must be done alone. No groups allowed.</u>

Program **Course**	Game Development - Advanced Programming INFO - 6046 Media Fundamentals - Fall 2023
Duration **Due Date**	**3 weeks** **November 22, 2023 11:59pm**
Total Marks **Weight**	80 + 10 Weight: 12-15% (Based on the number of projects (4-5)

<u>Project Description and Purpose</u>

The goal of this project is to create an interactive 3D audio scene using FMOD, C++, and OpenGL. Design and implement a 3D environment of their choice, including sound sources, occlusion, and 3D listener attributes. Additionally, demonstrate the Doppler effect and incorporate various DSP effects to enhance the audio experience.

Make sure the audio files you submit are ***<u>appropriate</u>*** 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.

<u>Project Requirements</u>

Create a scene (Prerequisite)

- Choose a 3D scene to create (e.g., outdoor park, interior of a building, a fantasy world).
- Utilize OpenGL to render the 3D environment.
- Implement a first-person camera to navigate the scene.

3D Listener Attributes (10 marks)

- Update the 3D listener's position to match the camera or character's position within the OpenGL scene.
- Set listener orientation, velocity, and other relevant attributes.

User Interaction (10 marks)

- Implement user interaction elements (e.g., picking up objects, triggering events) that affect the audio scene in real-time.

3D Audio Setup with FMOD (10 marks)

- Use FMOD to manage and play audio within the scene.
- Position sound sources within the 3D environment to represent objects, ambiance, or interactive elements.

Object Based Occlusion (20 marks)

- Define objects within the scene that act as walls or obstacles to occlude sound.
- Implement audio occlusion, ensuring that sounds are muffled or blocked when the listener is behind occluding objects.

Doppler Effect (10 marks)

- Create at least one sound source that moves quickly within the scene (e.g., a fast-moving vehicle, bird, or spaceship).
- Implement the Doppler effect to simulate the change in pitch as the moving sound source approaches or moves away from the listener.

DSP Effects (20 marks)

- Apply at least four different DSP (Digital Signal Processing) effects to various audio sources within the scene. Examples of DSP effects we implemented in class include reverb, echo, pitch shifting, and distortion.
- Each DSP effect should be applied appropriately to enhance the realism or artistic expression of the audio scene.

Bonus (10 marks)

- Implement function callbacks for entering and leaving sections that trigger audio to start and stop. You can use a simple distance calculation from listener to the audio source to detect entering and leaving.

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.