EECE 2140

Iteration 4 outline

**Team Objectives for this week:**
**1. Skip track**
**2. Create playlist**
**3. Create data structure**
   **a. Class: Song**
      **i. file:myFile.mp3**
      **ii. name:"BackInBlack"**
      **iii. lengh…**

**Objective:**
Begin coding the foundational components of your project, **ensuring that at least a basic version of each core functionality** is implemented and can be tested.

# 1. Implementing Core Functionalities

- Select at least **three essential functionalities** from Iteration 3 and **develop a working version**.
- Implement **input handling, processing logic, and output generation**.
- Ensure that the functionality follows the system overview defined in the previous iteration.
- Code should be well-structured(OOP), readable, and follow best practices.

# 2. Unit Testing and Debugging

- Identify and fix any **errors, logical issues, or inconsistencies** in the implementation.
- Utilize **print statements, or debugging tools** to track the flow of execution.

# 3. Updating System Overview (if necessary)

- Adjust the project scope **if needed**, ensuring feasibility for the next iterations.

# Deliverables for this Iteration:

**Code for at least three core functionalities**, properly structured and commented.
**Bug fixes and improvements** based on initial testing.
**Updated system overview diagram** (if needed).

**Push** the code to your GitHub account.

# Example: College Class Registration System

If this project was the focus, students should:

- Implement **student registration, course listing, and enrollment functionalities**.
- Test that student information is correctly stored and retrieved.
- Ensure that students can successfully enroll in a course.
- Debug issues related to data processing and system logic.

**Code for at least three core functionalities**: Upload MP3 files, play files, Stop file.
**Bug fixes and improvements** based on initial testing.
**Updated system overview diagram** (if needed).

Installing Pygame (for playing audio files:

### Method 1: Using Anaconda (If you're using Spyder from Anaconda)

1. Open **Anaconda Prompt** (search for it in Start Menu).
2. Run the following command:

conda install -c conda-forge pygame
Wait for the installation to complete, then restart Spyder.

Program text:

```
import pygame

pygame.mixer.init()
pygame.mixer.music.load("Acdc-Dc-BackInBlack.mp3")
pygame.mixer.music.play()

# Keep the program running while audio plays
while pygame.mixer.music.get_busy():
        pass
```