# Iteration Five

## The goal

To implement add sound effects

### Tasks

|  |  |  |
| --- | --- | --- |
| **Task** | **Time Estimate** | **Actual Time Taken** |
| Analysis and Planning | 90 min | 90 min |
| Coding | 30 min | 30 min |
| Testing | 30 min | 30 min |

### Analysis and Planning

I will define a range of sound effects to use for various game events. Most will be added to the game class as public or hidden attributes.

Each level will have a different theme song. The Game.gameMusic source will be reset when loading a new level. Different music will play when the game is paused, won or lost.

### Class Diagram Before:



### Class Diagram After:



### Design

## Activity Diagram



## Planning a Complex Algorithm

Define the problem

*Want to play a sound effect when game is finished*

Inputs to the routine

*state*

Outputs from the routine

*Message to console*

Pre-conditions

Game has ended, status is known

Post-conditions

*Message is displayed in the browser console, sound effect is played*

Name the Routine

*Game.end()*

Think about error handling

*Default audio will be for a lost game*

### Pseudocode

Game.end (state) {

Print message to console  
 create audio element  
 set audio source to losing sound effect  
 if state’s status is won {

Redefine audio source as winning sound effect

}  
 play audio

}

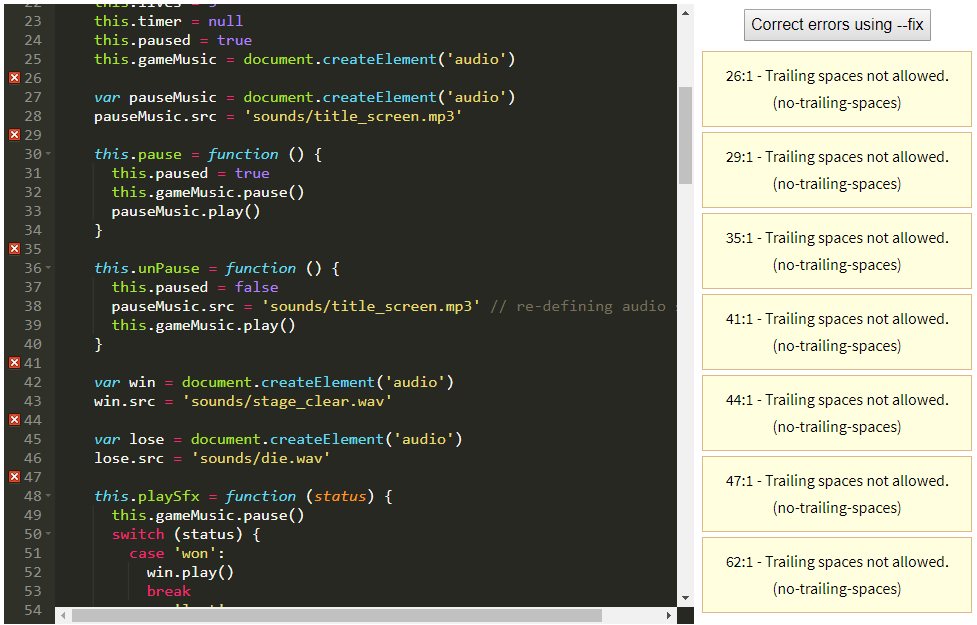
### Evaluation

I originally had my game launching when the page was loaded, and because the audio is set to play straight away I received an error message about unwanted media playing without user interaction.

To get around this error I added a button to launch the game. Once the user has clicked the button, they have interacted with the page and the browser plays the audio with no errors.

I ran my code through standardJS which complained about some minor style defects which were quickly fixed.

### standardJS Before



### standardJS After

