

COMP120 Tinkering Audio Contracts

There are FOUR contracts.

It is anticipated that areas of development will be varied, but have some overlap between them. The four contracts differ by their focus on diegetic ('in the world') or non-diegetic ('outside of the world') sound. They also differ by their focus is on particular effects driven by the player (e.g., character actions, navigating menus), or is an accompanying ambience (e.g. environmental sounds, game music, etc.).

Contract #1 - Sound Effect Generation (Diegetic Audio)

You are currently been tasked with creating a tool which will generate sound effects, these should consist of simple tones. The sound effect should be clearly tied to an action the player makes in the game (e.g., picking up an item, attacking, walking over a trap, or so on). An element of procedurality is expected to vary the tones.

Contract #2 - Ambient Platformers Audio Generation (Diegetic Audio)

As you are working on a platformer in a fantasy setting, it has been decided that you need a tool which will generate ambient audio tracks to make the player feel like they are in different areas of the fantasy setting. This should be configurable by a designer and is expected to vary by area of the realm.

Contract #3 - Melody Generation (Non-Diegetic Audio)

You are tasked with creating a tool which procedurally generates and exports a music track for the game. It will be important to research some rules to see which tones go well with each other, to inform the design of your procedural generator. The style of music is up to you as the developer, but should fit the original publisher's brief and the style of your particular game, but is anticipated to be akin to an 8-bit style game.

Contract #4 - User Interface Audio (Non-Diegetic Audio)

You need to develop a tool to generate audio for the user interface. These should be short tones to feed back to the player that they are successfully navigating the user interface and configuring the game's settings. This should be somewhat consistent across the interface, but the tone should be modified in a systematic way to indicate success or failure.