

Asset Store Package Documentation



Audio Manager

January 2

2017

by

[Oluwaseye Ayinla](#)

Overview

AudioManager is a Unity package for 2D games. It provides a simple way to manage and control your 2D game's background music and sound effects.

Features

- An audio manager component in inspector view
- Persistent singleton class call from code (no prefabs needed)
- Static function calls with callbacks
- 3 background music transition effects (Swift, Linear Fade & Cross Fade)
- Control of all sound effects in game without tags
- Integration with AudioManagerGroups
- Built-in sound pool for looping or repeating sounds
- Pool for loading audio assets from resource folder
- Fully commented code for understanding

Installation

Import the **Papae2D-AudioEngine- AudioManager.unitypackage** or copy the **Papae2D/AudioEngine/AudioManager** folder with its contents anywhere into your project folder and you are ready to go.

Usage

1. Drag and drop the **AudioManager.prefab** gameobject anywhere in the scene or hierarchy, edit any properties visible in the Inspector then call any API related function or attribute from code
2. Attach or add the **AudioManager.cs** class as a component to an empty game object in the scene, edit any properties visible in the Inspector then call any API related function or attribute from code
3. Just fire or call any API related function or attribute from code

Note that you have to import the namespace **Papae2D.AudioEngine** to use the AudioManager in script

Fade out the current music and fade in the next music within 4 seconds

```
AudioManager.PlayBGM(sound_clip, MusicTransition.LinearFade, 4f);
```

Play a sound clip for the duration of 10 seconds

```
AudioManager.PlaySFX(sound_clip, 10f);
```

Loop or repeat a sound at a particular location 5 times

```
AudioManager.RepeatSFX(sound_clip, 5, world_location);
```

Play a sound clip and track its progress from the repeat pool

```
AudioManager.PlaySFX(sound_clip, 120f, world_location, true);
```

Play a sound clip from the stored asset list once

```
AudioManager.PlayOneShot(AudioManager.GetClipFromAssetList("clip_name"));
```

Read the [API Reference](#) for more information.

Support

For enquiries, bug reports or suggestions contact oluwaseyeayinla@gmail.com