Audio

(Feedback - Environment - Mood)

Case Study: Overwatch

Audio Feedback

All sounds in the game are dynamically mixed based on their importance to the player (enemy heroes are louder than friendly ones, enemy heroes most dangerous to player are loudest, etc.)

Each hero has very distinct footsteps and movement sounds, so they can be recognizable only by sound.

Heroes automatically call out important gameplay information (“My shield is low”, “Teleporter under attack, etc.”)

Enemy heroes have different voice lines if they are on the opposing team (sometimes in a different language).

https://www.youtube.com/watch?v=teun_wZ8_LI

DubWars

https://www.youtube.com/watch?v=VJFi3gl7j6w

Music and Sound Effects Resources

You do not have to spend money on your projects! Here are some resources I've used in my own games.

Music
https://incompetech.filmmusic.io/search/

Sound Effects
https://freesound.org

SDL Mixer

“SDL_mixer is a sample multi-channel audio mixer library.
It supports any number of simultaneously playing channels of 16 bit stereo audio, plus a single channel of music, mixed by the popular FLAC, MikMod MOD, Timidity MIDI, Ogg Vorbis, and SMPEG MP3 libraries.”

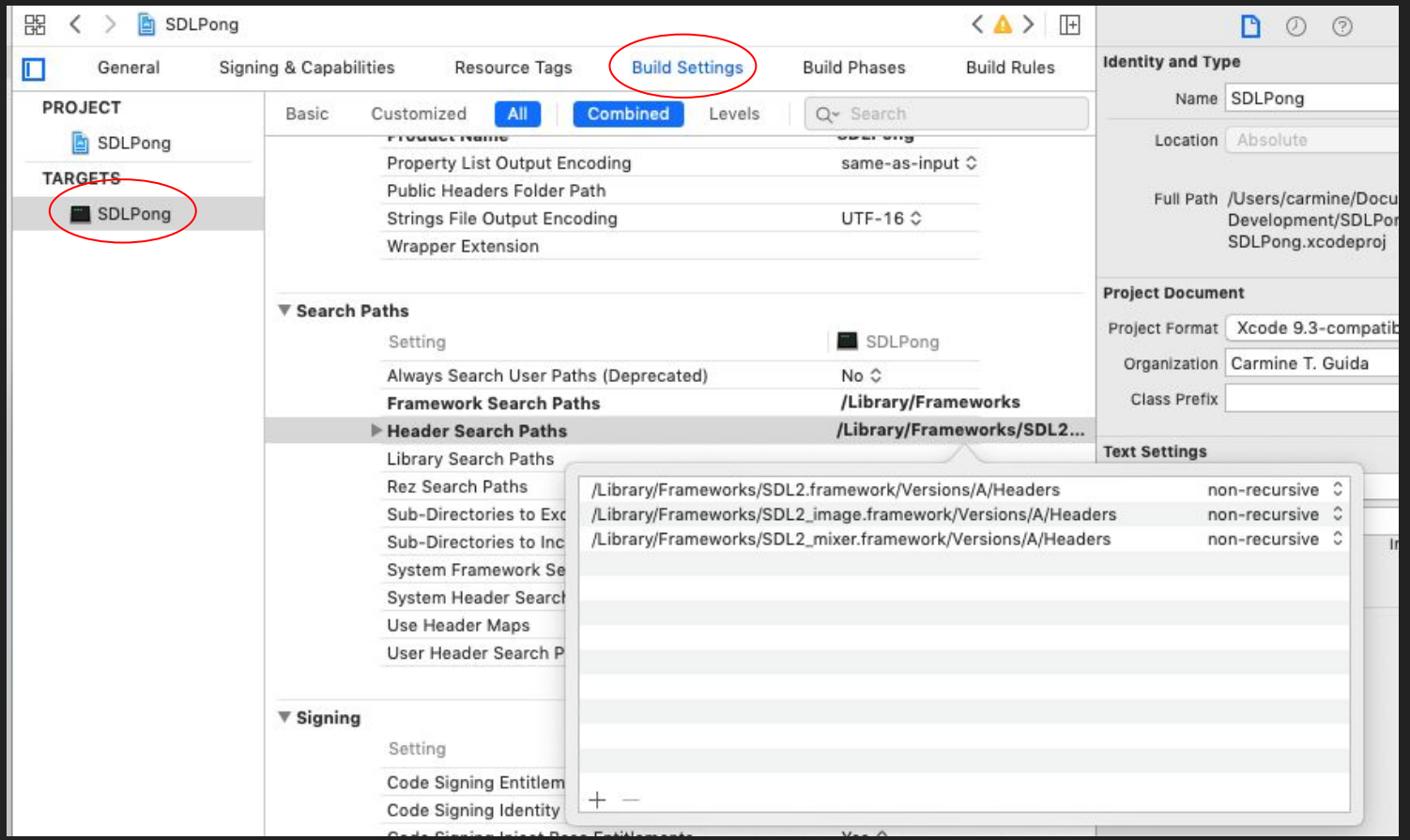
https://www.libsdl.org/projects/SDL_mixer/

SDL Mixer


COMP
EQ
HIGH
MID
LOW
AUX
MON
FX
PAN
BAL
MUTE
CLIP
1
2
3
4
5/6
7/8
9/10
11/12
MON
FX
2-TR
MAIN MIX
LEFT
RIGHT
MON SEND
FX SEND
2-TR/USB RTN
XPQ
USB
INTERFACE
FBQ
POWER +48 V
PHONES
L
R

Before we start...

SDL Mixer is not in the original Xcode template,
we need to add it.



Copy/Paste: /Library/Frameworks/SDL2_mixer.framework/Versions/A/Headers

Working with SDL_mixer

```
// Include required header file
// This is not in main.cpp template, you need to add it.

#include <SDL_mixer.h>
```

Working with SDL_mixer

```
// Initialize audio

void Initialize() {

    SDL_Init(SDL_INIT_VIDEO | SDL_INIT_AUDIO);
```

Working with SDL_mixer

This is the definition of the function to start audio.

```
int Mix_OpenAudio(int frequency, Uint16 format, int channels,
                                                 int chunksize)
```

How we will use it:

```
// Start Audio
Mix_OpenAudio(44100, MIX_DEFAULT_FORMAT, 2, 4096);
```

Music

(There is 1 channel for music)

Working with SDL_mixer

Music is similar to a texture where we load it and get a pointer to the data.

SDL Mixer has a built-in function for loading .mp3 files.

```
Mix_Music *music;

music = Mix_LoadMUS("music.mp3");
```

Working with SDL_mixer

We can play music using Mix_PlayMusic. The first parameter is the pointer to the mp3 we loaded and second parameter is the number of times to loop.

-1 means Loop forever

```
Mix_PlayMusic(music, -1);
```

Working with SDL_mixer

We can play set the music volume using Mix_VolumeMusic.
The volume range is from 0 to 128 (MIX_MAX_VOLUME)

```
// Set the music to half volume
Mix_VolumeMusic(MIX_MAX_VOLUME / 2);
```

Working with SDL_mixer

We can stop music playback using Mix_HaltMusic.

```
// Stop the music!
Mix_HaltChannel();
```

Sound Effects

(There are multiple channels for sound effects)

Your sounds must be a 16 bit WAV file.
You can use a free program such as Audacity
to get files into the proper format.

Working with SDL_mixer

Sound effects are similar to a texture where we load it and get a pointer to the data.

SDL Mixer has a built-in function for loading .wav files.

```
Mix_Chunk *bounce;

bounce = Mix_LoadWAV("bounce.wav");
```

Working with SDL_mixer

We can play a sound using Mix_PlayChannel.

Channel -1 means “use the first available”.

Loops is the number of times to loop.
-1 means Loop forever
0 means Play once (don’t loop)
1 means Play twice (play the first time, then loop 1 time)

```
Mix_PlayChannel(-1, bounce, 0);
```

Working with SDL_mixer

We can play a fade in a sound using Mix_FadeInChannel.

```
// Fade in (from 0 to full volume) over 1 second
Mix_FadeInChannel(-1, bounce, 0, 1000);
```

Working with SDL_mixer

We can play set the volume using Mix_Volume.
The volume range is from 0 to 128 (MIX_MAX_VOLUME)

```
// Set all channels to half volume
Mix_Volume(-1, MIX_MAX_VOLUME / 2);
```

Working with SDL_mixer

We can play set the volume of a sound (chunk) using Mix_VolumeChunk. The volume range is from 0 to 128 (MIX_MAX_VOLUME)

```
// Set the volume of the bounce sound to 1/4th
Mix_VolumeChunk(bounce, MIX_MAX_VOLUME / 4);
```

Working with SDL_mixer

We can stop playback on all channels using Mix_HaltChannel.

```
// Stop all sound effects
Mix_HaltChannel(-1);
```

Working with SDL_mixer

When our game exists, we need to cleanup the resources we allocated.

```
void ShutDown() {

    Mix_FreeChunk(bounce);
    Mix_FreeMusic(music);

    SDL_Quit();
}
```

Let’s Code!

Open your Pong assignment!

We’ll work on Lunar Lander after that!