Aula 9

ALÉM DE JOGAR,

EU FAÇO JOGOS!

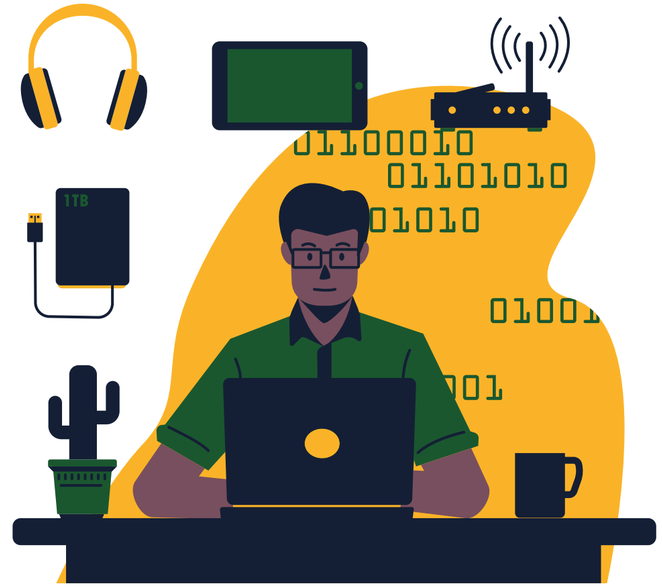
Instituto COOPERFORTE
20 anos transformando realidades

cpdi
Comitê para Democratização da Informática

Material desenvolvido pela professora Juliana Oliveira para o projeto além de jogar eu faço jogos. Divulgação proibida.

O que vamos aprender hoje?

- Áudio



Uso e funções de áudio

O sistema de áudio do Corona possui 32 canais distintos nos quais podemos reproduzir efeitos sonoros ou transmitir arquivos de áudio. Ele oferece uma configuração de volume excessiva para cada canal, além de um nível de volume principal para todos os canais.

É um sistema de "melhor esforço". Os sons solicitados serão reproduzidos o mais rápido possível, mas não há garantia de que eles começarão ou terminarão em horários exatos. Se estivermos transmitindo uma música e houver um buffer underrun devido ao estresse da CPU, o sistema de áudio continuará tocando assim que puder, mas a música poderá terminar de tocar mais tarde do que o previsto.

Formatos de áudio

O suporte de formatos de áudio varia de acordo com a plataforma

Plataforma	.wav	.mp3	.ogg	.aac	.caf	.aif
iOS	✓	✓		✓	✓	✓
Android	✓	✓	✓			
Mac OS	✓	✓	✓	✓	✓	✓
Área de trabalho do Windows	✓	✓	✓	✓		

Sobre os formatos

- Os arquivos de plataforma cruzada .wav devem ser descompactados em 16 bits.
- Para formatos altamente compactados como .mp3 e .aac (.mp4), o último é a melhor opção. .aac é o sucessor oficial de .mp3 do grupo MPEG. .mp3 teve vários problemas de patentes e royalties que expiraram em 2017, mas quando .aac foi ratificado, foi acordado que não haveria royalties necessários para distribuição.
- Ogg Vorbis (.ogg) é isento de royalties e patente , mas não é compatível com iOS.

Sobre os formatos

- Certos formatos - particularmente formatos com perdas e altamente compactados, como .mp3, .aac e .ogg - podem preencher/remover amostras no final de uma amostra de áudio e potencialmente quebrar um clipe de música com "looping perfeito" . Se estiver enfrentando falhas na reprodução em loop, tente usar .wav e certifique-se de que seus pontos inicial e final estejam limpos.
- Dependendo do formato — .mp3 em particular — a chamada audio.getDuration() pode retornar informações imprecisas, principalmente para arquivos carregados via audio.loadStream().

Som vs Transmissão

O sistema de áudio oferece duas funções para carregamento: audio.loadSound() e audio.loadStream. Não existe distinção obrigatória entre o arquivo que fornecemos para qualquer função, mas o método adequado deve ser usado de acordo com algumas diretrizes gerais:

Som vs Transmissão

- audio.loadSound() - Carrega um som inteiro na memória. Ele deve ser usado para arquivos de áudio mais curtos que podem ser usados repetidamente em todo o aplicativo.
- audio.loadStream() - lê em pequenos pedaços de um arquivo de áudio ao longo de sua duração. Deve ser usado para faixas de áudio mais longas, como música de fundo. Os arquivos transmitidos dessa forma podem ter um custo de latência e um custo de CPU ligeiramente maiores.

Canais e volume

Cada efeito sonoro ou faixa de áudio de streaming deve ser reproduzido em um canal distinto. Se não definirmos explicitamente um canal para reproduzir o áudio, o Corona irá localizar um canal livre para reproduzir o arquivo de áudio.

Em alguns casos, é útil reservar determinados canais para diferentes finalidades. Podemos por exemplo definir diferentes níveis de volume para música, fala e efeitos sonoros.

Canais e volume

Nesse caso, podemos reservar alguns canais na extremidade inferior da faixa e impedir efetivamente a atribuição automática nesses canais. Fazemos isso pelo comando audio.reserveChannels(), na qual bloqueamos os canais desejados.

Temos também várias funções para verificar o status de um canal. Isso inclui *audio.isChannelActive(), audio.isChannelPlaying()* e *audio.isChannelPaused()*

Controlando o volume

Dentro do sistema de áudio, temos um volume "mestre" e um nível de volume para cada canal. Tanto o volume mestre quanto os de canal individual podem ser controlados passando uma representação decimal de 0 a 100% para a propriedade audio.setVolume(). O volume principal não é necessariamente o mesmo que o volume interno do dispositivo, mas todos os níveis de volume são dimensionados proporcionalmente a esse volume interno.

Controlando o volume

Além de ajustar o volume geral ou de um canal específico, podemos também definir um volume mínimo ou máximo, onde qualquer volume abaixo e/ou acima do valor especificado serão reproduzidos no nível definido.

Alças de áudio

Quando carregamos um arquivo de áudio usando qualquer um dos dois comandos falados anteriormente, o Corona retorna um identificador para esse arquivo de áudio. Esse identificador pode ser usado para referenciar e reproduzir o arquivo de áudio, supondo que o identificador permaneça na memória.

Sound

Como já vimos, a função loadSound carrega um som inteiro na memória. Ele deve ser usado para arquivos de áudio mais curtos que podem ser usados repetidamente em todo o jogo. Depois de carregar o áudio para tocar o efeito sonoro é muito simples. Basta utilizar a propriedade audio.play e especificar qual o áudio desejado para reproduzir.

Stream

Já com o loadStream, não precisamos carregar o áudio antecipadamente. Basta carregar e reproduzir o arquivo também com a propriedade audio.play()

Controlando áudio

Temos algumas propriedades para controlar o áudio:

Função	Descrição
audio.pause()	Pausa a reprodução em um canal em reprodução ou em todos os canais em reprodução.
audio.resume()	Retoma a reprodução em um canal pausado ou em todos os canais pausados.
audio.rewind()	Retrocede o áudio para a posição inicial em um canal/identificador ativo ou em todos os canais ativos.
audio.seek()	Busca uma posição de tempo em um canal ativo ou identificador de áudio específico.
audio.stop()	Interrompe a reprodução em um canal ou em todos os canais e limpa os canais para que possam ser reproduzidos novamente.
audio.stopWithDelay()	Interrompe a reprodução em um canal ou em todos os canais após um período de tempo especificado.
audio.fade()	Fade um canal ou todos os canais para um volume especificado durante o período de tempo especificado.
audio.fadeOut()	Fade um canal ou todos os canais para o volume mínimo durante o período de tempo especificado.

Descartando áudio

Ao terminar de usar arquivos de áudio, precisamos descartá-los. Isso libera a memória alocada para esse arquivo. Para fazer o descarte, basta chamar a função audio.dispose() e passar a variavel do arquivo.

Importante

O áudio não deve estar ativo (reproduzindo/pausado) em nenhum canal ao tentar descartá-lo. Considere chamar audio.stop() antes de audio.despose() para garantir que o canal seja liberado. Além disso, não tente usar a alça depois que o arquivo de áudio tiver sido descartado e a memória liberada.