

Universidade de Brasília  
Curso de Graduação em Design

EDUARDO ZOLHOF

Relato sobre diálogos entre o design  
gráfico e o sound design em jogos  
digitais

Brasília  
2017

EDUARDO ZOLHOF

# Relato sobre diálogos entre o design gráfico e o sound design em jogos digitais

Trabalho apresentado no curso  
de Graduação em Design da  
Universidade de Brasília.

Orientador: Prof. Tiago Barros Pontes e Silva

Brasília

2017

## Resumo

O projeto “Relato sobre diálogos entre o design gráfico e o *sound* design em jogos digitais” se trata de uma experimentação no campo do *sound* design, visando estabelecer paralelos com o design gráfico.

Nas universidades brasileiras pouco é discutido o universo acústico no contexto do design. Até mesmo em cursos como jogos digitais, onde pode-se considerar imprescindível o conhecimento em *sound* design, o tema é abordado muito superficialmente, com pouca ou nenhuma profundidade técnica.

Este fato ocorre devido à falta de profissionais capacitados e à baixa produção teórica e acadêmica na área. Por consequência, encontra-se pouco material de estudo disponível para que as pessoas com interesse em *sound* design possam aprofundar os seus conhecimentos. Como pode-se observar, tal relação estabelece um ciclo que dificulta o progresso na área.

Dado o conhecimento e repertório do autor, Eduardo Zolhof, na área de áudio, e tendo este, experiência internacional em projetos de diversos portes e participação em várias conferências acerca do tema, e sua inquietação referente ao pouco conteúdo disponível nas universidades brasileiras sobre *sound* design, foi então proposto por meio deste projeto de graduação, um diálogo entre os universos acústico e gráfico, de tal forma a informar e alertar sobre a importância do áudio no contexto do *design*, além de criar um material literário acadêmico que fique como referência para gerações futuras.

Neste relato temos os principais conceitos referentes ao universo acústico, cuja compreensão se faz necessária para o entendimento do conteúdo apresentado nos textos subsequentes.

Também se faz presente uma apresentação do processo geral de *sound* design de acordo com a experiência do autor e os livros presentes nas referências bibliográficas. Nele, são mostrados os principais aspectos aos quais se deve atentar ao longo de projetos de áudio para jogos digitais. Expõe-se os principais pontos recorrentes e é feita uma rápida análise sobre o processo criativo.

Tendo em vista a experiência do autor, foram analisados dois projetos os quais o mesmo havia participado como *sound designer*: *Stoker* e *Nova Empire*. Ambos os jogos tiveram desenvolvimento independente à criação deste relatório, tendo suas análises realizadas após a finalização de suas produções. Tais escolhas devem-se simplesmente ao fato de o autor do relatório ter participado de suas produções.

Estes projetos foram escolhidos pela diferença acentuada em suas identidades visuais, um tendo estilo *8-bits* e outro *sci-fi*. Além disso, a diferença entre o escopo da produção dos projetos, um sendo de pequeno porte e outro de médio porte, também foi fator determinante.

Por meio da análise entre ambos os projetos e o processo geral, é possível perceber com clareza as principais diferenças no áudio final de cada jogo e o quanto da natureza do projeto, inclusive referente à parte gráfica, tem influência sob o produto final.

Por último há uma retomada dos pontos analisados ao longo do relatório e são feitas as considerações finais, onde verifica-se o sucesso do projeto, frente aos objetivos gerais propostos em sua introdução.

Palavras-chave: sound design, sound designer, design, design gráfico, jogos, games, jogos digitais, áudio, sonoplastia.

## Sumário

01 INTRODUÇÃO .....	01
02 PRINCIPAIS CONCEITOS .....	03
03 PROCESSO DE SOUND DESIGN.....	12
04 CASO 01 – PROJETO STOKER .....	18
05 CASO 02 – PROJETO NOVA EMPIRE .....	33
06 CONSIDERAÇÕES .....	39
07 REFERÊNCIAS BIBLIOGRÁFICAS .....	41

## Capítulo 01: Introdução

Para muitos *designers*, o design ainda é uma área que se limita apenas à produção gráfica e de produto, com poucas vertentes. Entretanto, este pensamento é incorreto, tendo em vista a existência de diversas outras vertentes dentro do design, tais como o design de interface, *game design*, *web design* e, a área de nosso interesse para este relatório, *sound design*.

O áudio é uma ferramenta extremamente poderosa que é constantemente deixada de lado ou esquecida em projetos de design. O objetivo não é defender o seu papel dentro do design, uma vez que sua importância já não é mais questionada. O estudo e compreensão do *sound design* faz-se essencial, pois sua integração com outras vertentes pode potencializar a capacidade e resultado do produto final e melhorar a experiência dos usuários finais, principalmente por meio da natureza psicológica da qual pode-se fazer proveito utilizando elementos sonoros.

Contudo, em projetos de design de jogos, nos quais existe um ferramental – inclusive nas próprias universidades brasileiras – voltado para a sua dimensão gráfica, pouco discute-se a dimensão acústica. Nesse sentido, faz-se pertinente uma investigação sobre esse campo e um estabelecimento de um diálogo entre as duas partes.

Tem-se como objetivo geral, a investigação do campo do *sound design*, novamente, estabelecendo um diálogo com as práticas do design gráfico e criando uma literatura acadêmica sobre o assunto, a qual pouco se faz presente nos acervos nacionais.

Dentro dos objetivos específicos, propõe-se:

- Investigar o campo do *sound design* no âmbito teórico-metodológico;
- Estabelecer um diálogo com as práticas frequentes no âmbito do *sound design*;
- Analisar dentro de projetos específicos de design digital os elementos que foram investigados;

Portanto, tem-se por meio deste projeto, uma forma de evidenciar um campo pouco abordado dentro do design, usando um projeto de bacharelado como uma possibilidade de demonstrar o repertório teórico-metodológico do autor do mesmo, o qual se justifica dada a sua experiência internacional e pioneirismo no campo do *sound* design para jogos digitais no Brasil, de forma a auxiliar tal investigação.

O produto final a ser entregue será uma monografia acerca do tema, com o objetivo de compartilhar todo o conhecimento adquirido ao longo do projeto e do repertório do autor.

A estrutura utilizada para este relatório será a de esclarecer os conceitos referentes ao universo acústico, para que o leitor não tenha dúvidas com relação a nenhum dos termos utilizados ao longo dos capítulos.

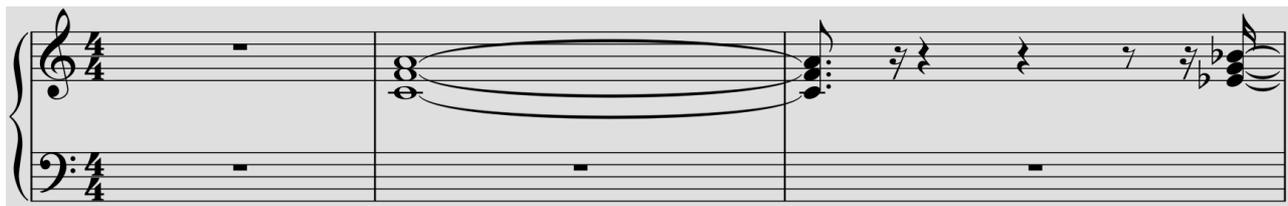
Entender-se-á como funciona o processo geral de *sound* design dentro do desenvolvimento de jogos. Neste capítulo serão dadas linhas gerais as quais costumam ser seguidas ao início de cada projeto novo, de forma a funcionar como guia dentro do desenvolvimento - o qual, muito provavelmente, será alterado a fim de se adaptar melhor às necessidades de cada jogo e equipe.

Analisar-se-á dois casos reais de projetos em que eu, o autor, estive envolvido na produção do áudio. O primeiro projeto se chama *Stoker*, projeto o qual o escopo da produção de áudio não foi muito extensa em comparação a jogos existentes atualmente no mercado. O segundo projeto se chama *Nova Empire*, projeto cujo escopo em termos de áudio foi consideravelmente maior do que o primeiro, e em termos de mercado, foi um jogo de média produção.

Por fim, serão feitas as observações e conclusões finais, por meio de uma retomada dos objetivos propostos na introdução e através de colocações adicionais, as quais serão frutos das observações feitas ao longo dos capítulos.



Figura 2 – Harmonia.



Fonte: do autor.

No sistema musical ocidental, tomamos como definição a existência de doze notas musicais distintas. A sua justificativa se dá por meio de cálculos matemáticos e um conhecimento da física das ondas, acompanhados de um contexto histórico, os quais a compreensão não se faz necessária para o entendimento da leitura. Consideremos, então, as seguintes notas: dó, dó sustenido (ou ré bemol), ré, ré sustenido (ou mi bemol), mi, fá, fá sustenido (ou sol bemol), sol, sol sustenido (ou lá bemol), lá, lá sustenido (ou si bemol) e si. A partir da nota si, repete-se a nota dó, iniciando o ciclo novamente.

Sabendo disso, podemos prosseguir para compreender o conceito de distância entre as notas. A menor distância entre duas notas no sistema musical ocidental (também chamado de bem-temperado), recebe o nome de semitom. Uma distância de um semitom é a distância entre uma nota e uma outra nota imediatamente próxima à primeira. Para fins de simplificação, chama-se de tom a distância equivalente a dois semitons.

Das doze notas citadas acima, são chamadas de notas naturais as sete notas equivalentes às teclas brancas de um piano. São elas: dó, ré, mi, fá, sol, lá e si. Chamam-se de notas acidentadas, as demais notas presentes, que tem o termo sustenido ou bemol próximas ao seu nome. Por consequência, dentro da linguagem musical, sustenidos e bemóis são chamados de acidentes.

Essencialmente, uma nota que recebe um acidente faz com que ela se desloque um semitom a sua direita, caso seja um sustenido, ou um semitom a sua esquerda, caso seja um bemol. Se levarmos em consideração as doze notas dadas como exemplo, percebemos que a distância entre a nota dó e a nota ré é de dois semitons (ou um tom), portanto, podemos dar um “passo” menor que o de um tom, saindo de uma nota para a outra. Para fazermos isso, devemos adicionar um acidente à nota, transformando o dó em

dó sustenido, para andarmos um semitom para a direita, ou fazendo com que o ré vire ré bemol, andando um semitom para a esquerda.

O conceito de intervalos musicais é importante pois cada distância possível entre duas notas, nos traz sensações e sentimentos distintos. Como consequência, para cada distância possível entre duas notas foi dado um nome diferente:

- (Intervalo de) segunda menor: (distância de) um semitom
- Segunda maior: um tom
- Terça menor: um tom e um semitom
- Terça maior: dois tons
- Quarta justa: dois tons e um semitom
- Quarta aumentada (ou quinta diminuta): três tons
- Quinta justa: três tons e um semitom
- Sexta menor: quatro tons
- Sexta maior: quatro tons e um semitom
- Sétima menor: cinco tons
- Sétima maior: cinco tons e um semitom
- Oitava justa: seis tons

Figura 3 – Acidentes e intervalos.

0	1	2	3	4	5	6	7	8	9	10	11	12
Dó	#/b	Ré	#/b	Mi	Fá	#/b	Sol	#/b	Lá	#/b	Si	Dó

Fonte: do autor.

Levando em consideração a nota dó como referência, perceba que ao chegar na oitava justa, após andar uma distância de seis tons, chegamos novamente na nota dó, fazendo com que o ciclo se repita.

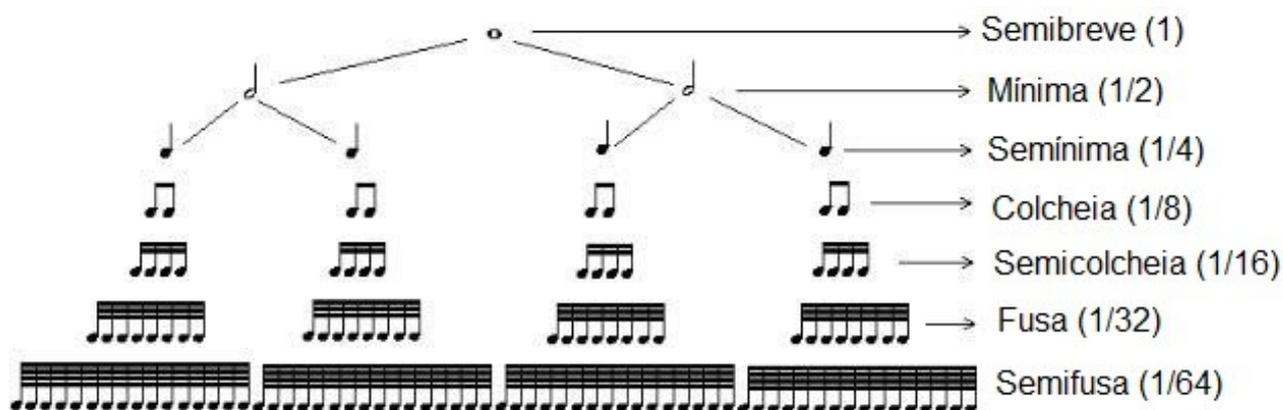
Outro conceito fundamental a ser compreendido é o de ritmo. Ritmo é a distribuição ordenada e a relação entre as durações das notas executadas sucessivamente (Teoria da Música, 1996). Ou seja, quando falamos que uma passagem musical tem ritmo acelerado, significando que a sucessão das notas e/ou pausas se dá de forma igualmente acelerada. A percepção de ritmo acelerado e lento se dá de maneira semelhante à linguagem verbal,

ao ato de falar rápido ou devagar. Nossa ideia de velocidade é baseada em nossa formação cultural e na maioria das vezes, dentro do universo sonoro, se dá pela referência da música popular daquela determinada época.

Dentro do ritmo, temos as figuras rítmicas que nada mais são do que valores de tempo atribuídos às notas ou pausas. Na música ocidental temos 7 figuras rítmicas:

- Semibreve: (duração de) 1 compasso 4/4
- Mínima:  $\frac{1}{2}$  de compasso
- Semínima:  $\frac{1}{4}$  de compasso
- Colcheia:  $\frac{1}{8}$  de compasso
- Semicolcheia:  $\frac{1}{16}$  de compasso
- Fusa:  $\frac{1}{32}$  de compasso
- Semifusa:  $\frac{1}{64}$  de compasso

Figura 4 – Figuras rítmicas e valores.



Fonte: do autor

Antes de seguirmos em frente, vamos esclarecer o conceito de compasso, tempo e métrica. O compasso é um bloco de notas cujos números de notas dentro de si é definido pela métrica. A métrica é a formalização do ritmo com auxílio do tempo. Como o ritmo é subjetivo a quem executa a música, a métrica e o tempo servem como ponto de referência. O tempo é definido pela unidade de bpm, isto é, batidas por minuto. Uma passagem musical que esteja a 60 bpm significando que serão escutadas 60 batidas em 1 minuto, ou seja, 1 batida por segundo. A métrica serve para definir o valor desta batida dentro do contexto musical. A métrica é sempre definida pela seguinte proporção: número

de figuras rítmicas em um compasso por figura rítmica de referência. Se em uma determinada música a 60 bpm, temos que a métrica é de 4/4, isto significa que dentro de 1 compasso teremos 4 notas com valores de semínimas.

Dentro do universo do *sound design*, vamos definir os conceitos referentes às propriedades acústicas do som.

Para começar, temos a frequência dos sons. Elas determinam se os sons são agudos (de alta frequência) ou graves (de baixa frequência). Sons agudos possuem ondas sonoras mais curtas, enquanto sons graves possuem ondas longas. Além disso, os sons agudos necessitam de menor potência para serem reproduzidos, enquanto sons graves precisam de maior potência – este fato explica o porquê de aparelhos celulares serem muitas vezes incapazes de reproduzir frequências baixas. Uma última propriedade é que, dado o fato de que sons graves possuem ondas sonoras longas, elas normalmente precisam de uma distância mínima entre o ouvinte e a reprodução do som para que sejam ouvidas com clareza. Em inglês, o termo referente à frequência se chama *pitch*.

Figura 5 – Onda sonora de frequência alta.



Fonte: do autor

Figura 6 – Onda sonora de frequência baixa.

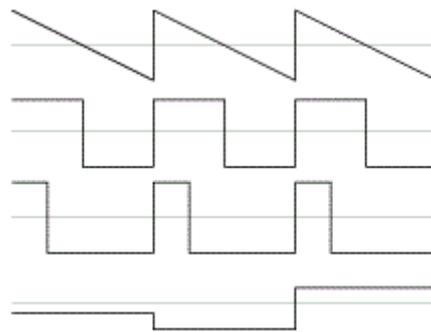


Fonte: do autor

Temos também o conceito de volume, o qual faz referência à potência com a qual um determinado som é reproduzido. A terminologia correta a ser utilizada junto ao volume de som, diz respeito à quantidade de som presente em um determinado espaço, portanto, o correto a se dizer é que há um grande ou pequeno volume sonoro. Entretanto, popularmente é dito que um volume é alto ou baixo, então, ainda que o termo esteja incorreto para questões técnicas, vamos utilizá-lo de acordo com sua colocação informal para os textos redigidos nos capítulos seguintes.

Outro termo importante é timbre. Tecnicamente, diferentes timbres acontecem devido a diferentes formatos que as ondas sonoras podem ter. Cada formato de onda sonora diferente produz um som diferente e traz em si uma característica única. Por exemplo, um piano soa diferente de um violão por ambos produzirem ondas sonoras diferentes, fazendo assim com que tenham uma identidade própria. Instrumentos que têm sonoridade parecida, em muitas vezes, são instrumentos que possuem uma construção, materiais e formatos semelhantes, fatores que afetam diretamente como som é gerado.

Figura 7 – Timbres: diferentes ondas sonoras.



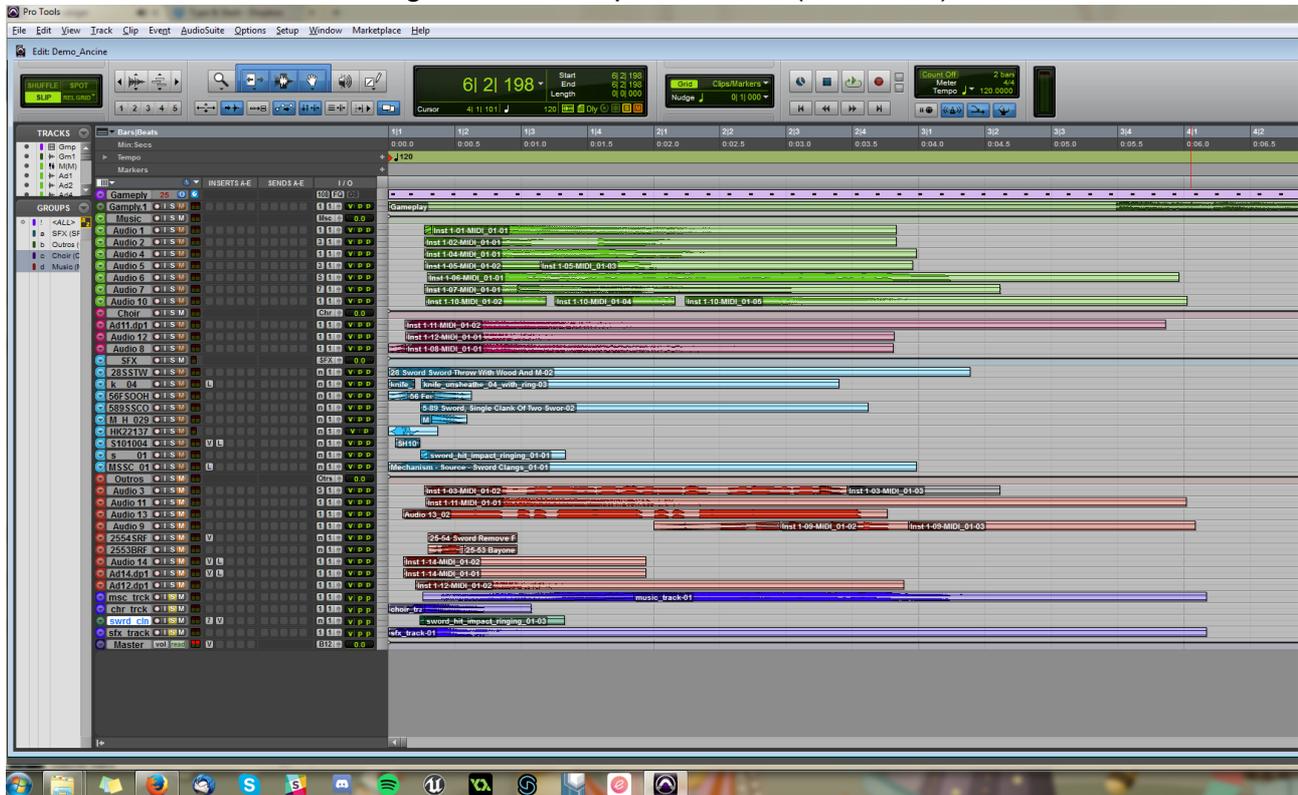
Fonte: do autor.

Sons sintetizados são sons que são criados a partir de impulsos elétricos e sistemas eletrônicos. Em sua grande maioria são timbres criados a partir de teclados digitais e analógicos, chamados de teclados sintetizadores.

Ao que diz respeito à produção de áudio, temos também diversos conceitos muito importantes para a compreensão do *sound design*.

O primeiro termo importante que se deve ter conhecimento é o de DAW, sigla em inglês para *Digital Audio Workstation*. Fazendo equivalência ao design gráfico, uma DAW é um software de produção de áudio da mesma forma que o *Photoshop* e/ou *Illustrator* funcionam como um canvas digital para artistas gráficos.

Figura 8 – Exemplo de DAW (*Pro Tools*).



Fonte: do autor.

*Sound assets* são os arquivos de áudio do computador que são gerados ao longo do desenvolvimento de um projeto. O termo *asset* é utilizado junto a diversas outras áreas, como por exemplo, no departamento de arte, temos os *art assets*.

*Chip sounds* são sons provenientes de placas de áudio capazes de reproduzir sons de até 16-bits (*Super Nintendo*), entretanto, na literatura de áudio, o termo é comumente utilizado para fazer referência a sons 8-bits (*GameBoy*). Também podem ser chamados de chip tunes, entretanto, quando chamados assim, costumam fazer referência exclusivamente à trilha sonora.

O processo de equilíbrio das frequências em um som é chamado de equalização. Por exemplo, o processo de equalizar uma música faz referência ao ato de fazer alterações nos volumes das diferentes frequências presentes.

A mixagem de um jogo, de uma música, de uma produção de áudio é o processo de equilibrar e balancear (muitas vezes por alterações nos volumes) todos os sons presentes naquela mídia, de forma que soem de forma agradável ao ouvinte e de acordo com o contexto da produção.

Modulação é o nome que se dá a variações feitas em um *asset* de áudio, em alguma de suas propriedades. Por exemplo, a modulação de *pitch* diz respeito a um som que terá suas frequências alteradas ao longo de uma linha do tempo, a fim de criar um determinado efeito ao ouvinte. Também recebe o nome de automação.

Bibliotecas de áudio são diversos *assets* de áudio que são vendidos, normalmente em conjuntos, com licenças comerciais livres ao comprador para que ele possa utilizá-los em seus projetos. Por exemplo, um *sound designer* que precise utilizar sons de leões em algum projeto e não tenha como gravar o áudio de um leão, tem como opção comprar uma biblioteca de sons de leões gravados por outros *sound designer*.

Temos também o conceito de *Engine*. Uma *engine* de desenvolvimento de games é um software otimizado para a produção de jogo. Ele atua juntando todos os *assets* presentes de todas as partes envolvidas na produção, o código do jogo e compila criando um arquivo executável, que é o jogo em si.

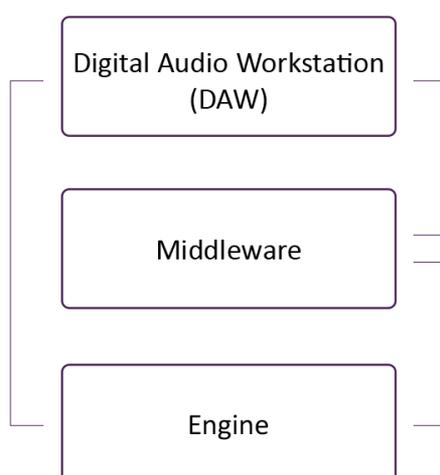
*Middlewares* são *softwares* que atuam entre a DAW e a *Engine*. São *softwares* específicos para a parte de áudio, que visam facilitar a implementação dos *assets* de áudio no jogo, de tal forma que não exija do artista de áudio um conhecimento a fundo de programação para fazê-lo.

Por último, a parte das definições supracitadas, devemos deixar claro as definições de *sound design* e composição musical.

O *sound design* é a solução de problemas através da utilização de elementos e conceitos exclusivos do universo acústico. Há quem discuta que o campo da composição musical não se encaixe dentro da definição de *sound design*, tendo em vista que a música pode ser criada com o único intuito de satisfação pessoal. Se considerarmos uma análise mais filosófica desta afirmação, podemos dizer que a satisfação pessoal surge como solução para o problema da necessidade de o indivíduo se expressar. Entretanto, de forma mais objetiva, ainda que se considere como verdadeira esta afirmação, pode-se dizer que o *sound design* ainda se faz presente nos elementos da composição musical, tanto em questão de contexto e temática (onde será reproduzido) , estruturação (harmonia, melodia e tempo) e características sonoras (efeitos e texturas através da manipulação dos sons).

Os problemas para os quais o *sound design* propõe solução podem surgir dentre os mais diversos contextos e situações. No nosso dia-a-dia nos deparamos com o *sound design* ao ouvir os *beeps* emitidos por semáforos quando estes estão com a luz vermelha acesa (vinculado ao design inclusivo), nos sons de alarmes (vinculado ao design de ambientes, pois considera-se a natureza do som e sua relação com os elementos físicos - tais como refração, dispersão, alcance, etc), aplicativos de celular (vinculado ao design de interfaces), filmes e *games* (design gráfico, narrativa), entre outros. Tendo em vista o imenso alcance do *sound design*, neste capítulo iremos nos limitar apenas a sua relação com os elementos visuais.

Figura 9 – *Flow* da produção de áudio em jogos digitais.



Fonte: do autor.

## Capítulo 03: Processo de Sound Design

Assim como em qualquer outra área dentro do design, não existe receita de bolo para o processo de *sound design* dentro de um projeto. Entretanto, existem linhas gerais dentro das quais é possível criar um molde que pode ser utilizado como referência e ponto de partida ao início de cada novo projeto.

Neste capítulo, vamos detalhar e compreender melhor o processo geral de *sound design*, para então podermos analisar nos capítulos seguintes, na prática, em projetos reais, como este molde se comporta e de que forma ele acaba sendo remodelado a fim de atender às expectativas de um determinado projeto.

O primeiro momento de um projeto vai depender de como foi a aproximação do *sound designer* ao desenvolvedor e ao projeto em si. A abordagem pode ocorrer de diversas maneiras: é possível que o próprio *sound designer* tenha descoberto o jogo em desenvolvimento, e então tenha entrado em contato com o desenvolvedor para oferecer os seus serviços, ou seja, ele muito provavelmente já tem algum conhecimento sobre o jogo. É possível que o desenvolvedor procure o *sound designer*, por ter referências de seus trabalhos anteriores, e portanto já conheça o seu trabalho. Também é possível que tanto o desenvolvedor quanto o *sound designer* tenham recém-descoberto o trabalho um do outro e tenham interesse em trabalhar juntos.

Cada caso fará com que a abordagem inicial do projeto seja diferente. Normalmente, quando o *sound designer* faz o primeiro contato com o desenvolvedor, ele costuma tomar uma posição de empregado, o qual deve seguir o direcionamento imposto por outras pessoas no projeto. Cabe então ao próprio profissional, saber dialogar com a equipe envolvida e explicar que mais do que alguém contratado apenas para preencher as lacunas de áudio, sua formação em design também permite que ele faça análises construtivas e seja capaz de dar sugestões que melhorem a experiência do usuário durante o jogo.

Quando é o desenvolvedor que procura o *sound designer*, na maioria das vezes, por conhecer melhor o seu trabalho antes de efetivar a sua participação no projeto, a equipe como um todo já deposita uma maior confiança em seu trabalho, não só valorizando, mas também ativamente buscando sua opinião em diversos aspectos do design do jogo,

fazendo com ele também tenha um impacto na produção do jogo, colocando outros profissionais de outras áreas a sua disposição para eventuais alterações no jogo.

Independente de como o primeiro contato é feito, o importante é que o primeiro passo dentro de um projeto é conhecê-lo a fundo. A conversa inicial com o desenvolvedor é fundamental para que o *sound designer* possa ter as primeiras noções de como o projeto será, portanto é seu dever conseguir o máximo de informação possível a respeito do projeto. As informações desejáveis normalmente são: conceito do jogo, número de participantes da equipe, escopo geral do projeto, prazos de produção e as opiniões do desenvolvedor em relação a como o áudio deve ser - tanto a sua visão em relação ao universo acústico do jogo em termos de estética sonora quanto à colocação dos sons em si dentro do jogo.

É de responsabilidade do *sound designer* criar junto ao desenvolvedor uma lista inicial com os prováveis *assets* que existirão no jogo. Para tal, é interessante que se tenha disponível uma *build* prévia de desenvolvimento, uma lista inicial com as impressões apenas do desenvolvedor e quaisquer possíveis documentos que auxiliem na alocação de recursos sonoros dentro do jogo, sejam eles relacionados a game design ou até mesmo a conceitos de arte.

Figura 10 – Exemplo de lista de *assets*.

A	B	C	D	E	F	G	H	I	J
Nome do Arquivo	Variáveis	Event	Bank	Descrição	Pathing	Trigger	Parâmetro	Observações	Status
1	watchbot_kick_weak_turbo_burst_#	watchbot_kick_weak	mecha_watchbot	Início do chute normal	MechaWatchbot	Animação de início do movimento de chute	Sim	Parâmetro por CHARGE	-
2	watchbot_kick_weak_turbo_loop_#	watchbot_kick_weak	mecha_watchbot	Burst turbo fraco	MechaWatchbot	Burst do turbo fraco	Sim	Parâmetro por CHARGE	-
3	watchbot_kick_weak_turbo_loop_#	watchbot_kick_weak	mecha_watchbot	Hold turbo fraco	MechaWatchbot	Segurar o chute com o turbo fraco	Sim	Parâmetro por CHARGE	-
4	watchbot_kick_weak_turbo_loop_#	watchbot_kick_weak	mecha_watchbot	Burst turbo forte	MechaWatchbot	Burst de transição entre turbo fraco e forte	Sim	Parâmetro por CHARGE	-
5	watchbot_kick_weak_turbo_loop_#	watchbot_kick_weak	mecha_watchbot	Hold turbo forte	MechaWatchbot	Segurar o chute com o turbo forte	Sim	Parâmetro por CHARGE	-
6	watchbot_kick_weak_#	watchbot_kick_weak	mecha_watchbot	Chute fraco sem impacto	MechaWatchbot	Animação do chute fraco	Sim	Parâmetro por CHARGE	-
7	watchbot_kick_weak_#	watchbot_kick_weak	mecha_watchbot	Chute forte sem impacto	MechaWatchbot	Animação de chute forte	Sim	Parâmetro por CHARGE	-
8	watchbot_kick_weak_#	watchbot_kick_weak	mecha_watchbot	Impacto do chute fraco ao acertar algo	MechaWatchbot	Final da animação do chute fraco ao colidir com algo	Sim	Verificar necessidade de diferenciar acerto outro mecha ou buildings if	-
9	watchbot_kick_weak_#	watchbot_kick_weak	mecha_watchbot	Impacto do chute forte ao acertar algo	MechaWatchbot	Final da animação do chute forte ao colidir com algo	Sim	Parâmetro por CHARGE	-
10	watchbot_kick_weak_#	watchbot_kick_weak	mecha_watchbot	Chute baixo fraco sem impacto	MechaWatchbot	Animação do chute baixo fraco	Sim	Parâmetro por CHARGE	-
11	watchbot_kick_weak_#	watchbot_kick_weak	mecha_watchbot	Chute baixo forte sem impacto	MechaWatchbot	Animação de chute baixo forte	Sim	Parâmetro por CHARGE	-
12	watchbot_kick_weak_#	watchbot_kick_weak	mecha_watchbot	Impacto do chute baixo fraco ao acertar algo	MechaWatchbot	Final da animação do chute baixo fraco ao colidir com algo	Sim	Verificar necessidade de diferenciar acerto outro mecha ou buildings if	-
13	watchbot_kick_weak_#	watchbot_kick_weak	mecha_watchbot	Impacto do chute baixo forte ao acertar algo	MechaWatchbot	Final da animação do chute baixo forte ao colidir com algo	Sim	Parâmetro por CHARGE	-
14	watchbot_kick_weak_#	watchbot_kick_weak	mecha_watchbot	Chute aéreo fraco sem impacto	MechaWatchbot	Animação do chute aéreo fraco	Sim	Parâmetro por CHARGE	-
15	watchbot_kick_weak_#	watchbot_kick_weak	mecha_watchbot	Chute aéreo forte sem impacto	MechaWatchbot	Animação de chute aéreo forte	Sim	Parâmetro por CHARGE	-
16	watchbot_kick_weak_#	watchbot_kick_weak	mecha_watchbot	Impacto do chute aéreo fraco ao acertar algo	MechaWatchbot	Final da animação do chute aéreo fraco ao colidir com algo	Sim	Verificar necessidade de diferenciar acerto outro mecha ou buildings if	-
17	watchbot_kick_weak_#	watchbot_kick_weak	mecha_watchbot	Impacto do chute aéreo forte ao acertar algo	MechaWatchbot	Final da animação do chute aéreo forte ao colidir com algo	Sim	Parâmetro por CHARGE	-
18	watchbot_kick_weak_#	watchbot_kick_weak	mecha_watchbot	Início do soco normal	MechaWatchbot	Animação de início do movimento de soco	Sim	Parâmetro por CHARGE	-
19	watchbot_kick_weak_#	watchbot_kick_weak	mecha_watchbot	Burst turbo fraco	MechaWatchbot	Burst do turbo fraco	Sim	Parâmetro por CHARGE	-
20	watchbot_kick_weak_#	watchbot_kick_weak	mecha_watchbot	Hold turbo fraco	MechaWatchbot	Segurar o soco com o turbo fraco	Sim	Parâmetro por CHARGE	-
21	watchbot_kick_weak_#	watchbot_kick_weak	mecha_watchbot	Burst turbo forte	MechaWatchbot	Burst de transição entre turbo fraco e forte	Sim	Parâmetro por CHARGE	-
22	watchbot_kick_weak_#	watchbot_kick_weak	mecha_watchbot	Hold turbo forte	MechaWatchbot	Segurar o soco com o turbo forte	Sim	Parâmetro por CHARGE	-
23	watchbot_kick_weak_#	watchbot_kick_weak	mecha_watchbot	Instabilidade mecânica ao carregar o soco	MechaWatchbot	Segurar o soco com o turbo forte	Sim	Parâmetro por CHARGE - Automação por VOLUME/TEMPO	-
24	watchbot_kick_weak_#	watchbot_kick_weak	mecha_watchbot	Baço danificado por overheat do soco forte	MechaWatchbot	Soco forte carregado por tempo excessivo	Sim	Parâmetro por CHARGE	-
25	watchbot_kick_weak_#	watchbot_kick_weak	mecha_watchbot	Soco fraco sem impacto	MechaWatchbot	Animação do soco fraco	Sim	Parâmetro por CHARGE	-
26	watchbot_kick_weak_#	watchbot_kick_weak	mecha_watchbot	Soco forte sem impacto	MechaWatchbot	Animação de soco forte	Sim	Parâmetro por CHARGE	-
27	watchbot_kick_weak_#	watchbot_kick_weak	mecha_watchbot	Impacto do soco fraco ao acertar algo	MechaWatchbot	Final da animação do soco fraco ao colidir com algo	Sim	Verificar necessidade de diferenciar acerto outro mecha ou buildings if	-
28	watchbot_kick_weak_#	watchbot_kick_weak	mecha_watchbot	Impacto do soco forte ao acertar algo	MechaWatchbot	Final da animação do soco forte ao colidir com algo	Sim	Parâmetro por CHARGE	-
29	watchbot_kick_weak_#	watchbot_kick_weak	mecha_watchbot	Apagar	MechaWatchbot	Animação apagando-se	Não	-	-
30	watchbot_kick_weak_#	watchbot_kick_weak	mecha_watchbot	Levantar	MechaWatchbot	Animação levantando-se	Não	-	-
31	watchbot_block_stance_in_#	watchbot_block_stance_in (NOVO)	mecha_watchbot	Entrada da posição de defesa	MechaWatchbot	Início da animação de defesa	Não	-	-
32	watchbot_block_stance_out_#	watchbot_block_stance_out (NOVO)	mecha_watchbot	Saída da posição de defesa	MechaWatchbot	Final da animação de defesa	Não	-	-
33	watchbot_block_hl_#	watchbot_block_hl	mecha_watchbot	Defesa de hl	MechaWatchbot	Colisão frontal enquanto na posição de defesa	Não	-	-
34	watchbot_block_#	watchbot_block	mecha_watchbot	Dado	MechaWatchbot	Animação de dado	Não	-	-
35	watchbot_jump_takeoff_quick_#	watchbot_jump_takeoff	mecha_watchbot	Pulo rápido (opera mecânico)	MechaWatchbot	Animação de pulo rápido	Sim	Parâmetro por CHARGE	-
36	watchbot_jump_landing_quick_#	watchbot_jump_landing	mecha_watchbot	Alemasagem rápida (opera mecânico)	MechaWatchbot	Animação de alemasagem rápida	Sim	Parâmetro por CHARGE	-

Fonte: do autor.

Uma vez que as colocações iniciais tenham sido feitas e após os contratos terem sido assinados, é iniciada a pré-produção. Ela é importante para a execução do projeto e na maioria esmagadora das vezes, é negligenciada pelos desenvolvedores. Uma pré-produção bem consolidada permite que a produção ocorra de forma contínua e minimiza

consideravelmente a possibilidade de erros, e por consequência, a perda de tempo e até mesmo dinheiro ao longo da produção.

Nela, fazemos o planejamento de toda a programação da produção, com todas as datas e prazos, já organizando níveis e hierarquias dos elementos de produção, para que, caso seja necessário realizar alterações em função de eventos que fogem ao controle do *sound designer*, ele possa ter uma reação rápida e se adaptar às circunstâncias.

Além disso, também é na pré-produção que ocorre a pesquisa, estudos e busca por referências. Dependendo do projeto, é possível que o *sound designer* seja colocado fora da sua área de conforto. Por exemplo, caso o mesmo profissional precise também atuar como compositor do projeto, a chance de ele ter de compor uma trilha sonora que não necessariamente se encaixe com o que está acostumado, não é baixa, pois a dinâmica de elementos e níveis existentes em videogames faz com que em um mesmo jogo, possa existir diversos cenários e ambientes, completamente diferentes entre si, o que exige do compositor uma grande versatilidade. Ao que diz respeito ao *sound design* em si, é possível que o universo acústico do jogo também fuja ao de costume o qual o profissional está acostumado a atuar - alguém que tem experiência em jogos realistas, provavelmente enfrentará dificuldades ao ter que se adaptar a um jogo em pixel art. A parte da pré-produção permite que o *sound designer* tenha tempo para descobrir, conhecer e se sentir confortável dentro do universo sonoro do jogo que será produzido. Muitas vezes, ela corresponde a um terço do tempo disponível para a realização do projeto.

Uma vez encerrada a pré-produção, começamos a produção em si. A princípio, ela deve ocorrer de forma fluida sempre que a pré-produção for bem realizada, entretanto, dado o caráter criativo da profissão e das funções exercidas pelo profissional, a produção muitas vezes acaba se tornando um processo de exploração, tentativa, erro e acerto.

Diversos fatores subjetivos afetam a capacidade criativa de um indivíduo, mas existem abordagens que permitem a qualquer profissional manter um fluxo criativo e produtivo de certa forma constante.

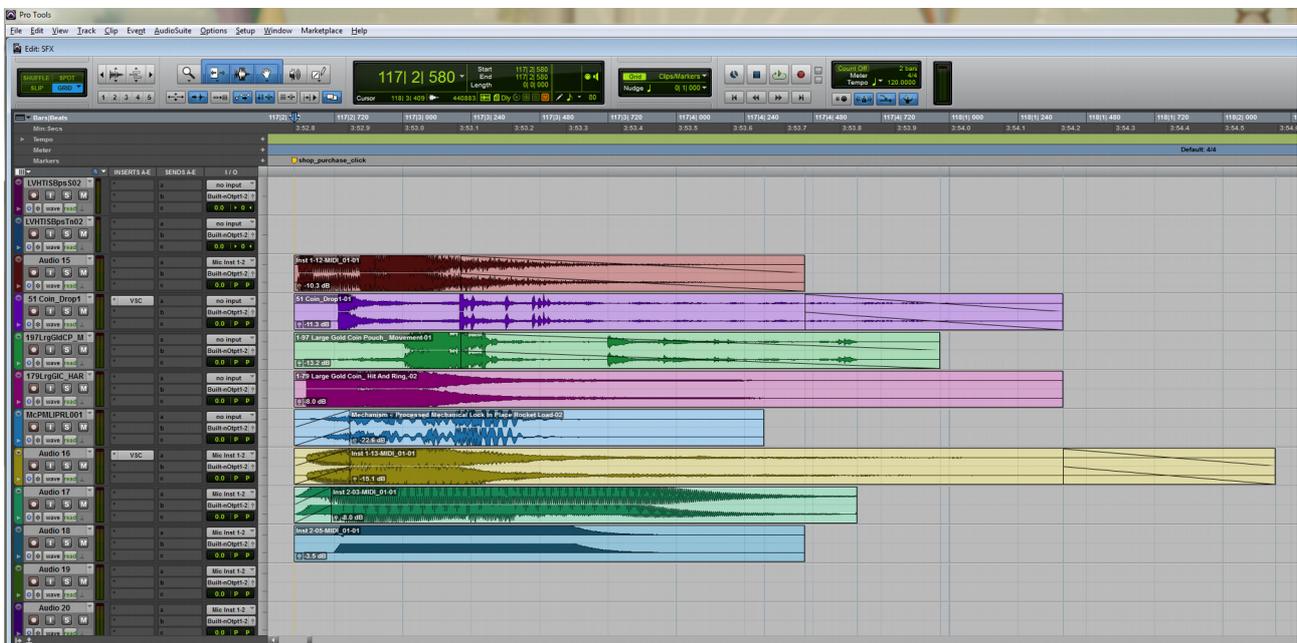
Dentro do *sound design*, é muito mais fácil corrigir algo do que criá-lo do zero. Psicologicamente, a barreira de se criar algo é muito maior do que a de corrigir algo que já havia sido feito. Portanto, ao início de toda a produção é recomendável que o profissional não se prenda a detalhes e tente preencher todos os *assets* da produção o

mais rápido possível, mesmo que os sons não se encaixem totalmente num primeiro momento ou sejam de má qualidade.

Apesar de parecer perda de tempo, este passo permite com que o *sound designer* saia do estágio de ter um canvas em branco a sua frente, e passe a ter algo que ele possa analisar se está funcionando ou não. Este pontapé inicial, apesar de parecer pequeno e até mesmo não muito apropriado, permite com que o *sound designer* saia do seu estado de inércia e apreensão, para um estado de criação e reflexão, onde ele já coloca o seu trabalho em movimento.

Ao que diz respeito à criação dos efeitos sonoros em si, no geral, o *sound designer* deve sempre tentar ser o mais direto possível. Deve-se pensar que cada som conta em si uma história, tendo início, meio e fim, no entanto, isso não quer dizer que todo som deve ser complexo e de difícil produção - pelo contrário: um *sound designer* eficiente sabe resumir a história e contá-la através de uma utilização mínima de recursos. No geral, a ideia da produção de um som é tentar, através de uma única layer, chegar o mais próximo possível ao resultado desejado, e a partir de então, realizar as correções e adições necessárias para que efeito sonoro soe completo.

Figura 11 – Construção de um efeito sonoro por *layers*.



Fonte: do autor.

## Áudio 1 – Efeito sonoro da Figura 11.

<https://www.dropbox.com/sh/rjveq20bd53joel/AADappEsZe0EQBpLXYx2gChba?dl=0>

Fonte: do autor.

Na música, cabe ao compositor criar uma trilha sonora que esteja de acordo com o jogo. Pode parecer óbvio, mas cabe ao profissional saber interpretar as necessidades do jogo e criar ambientes sonoros que, em sua grande maioria, não deve se destacar do jogo em si, mas sim fazer parte dele, junto a ele, de forma a não chamar atenção nem de mais nem de menos. A trilha deve ter um equilíbrio perfeito pois ela ajuda a estabelecer o estado emocional do jogador, sem com que o jogador perceba isso - a menos claro, que seja o papel de uma determinada música em um certo momento, chamar a sua atenção.

É muito comum ao longo da produção, serem criadas diversas versões para um mesmo *asset*, em busca da perfeição. É muito importante que o profissional de áudio mantenha registros de versões anteriores e anotações dos feedbacks dados pelos desenvolvedores. Não é incomum, após várias tentativas de criar um mesmo som, analisarmos as tentativas anteriores e somente então perceber que parte da resposta que procuramos está nessas tentativas, pois por serem fruto de ideias que não carregavam em si conceitos e julgamentos, ao contrário de versões já revisitadas, muitas vezes possuem elementos que estavam corretos desde o início, porém mal executados. Também não é incomum que o próprio desenvolvedor mude de opinião ao longo da produção e passe a gostar de uma versão inicial de um efeito sonoro, por isso a importância de manter tudo devidamente registrado.

A forma como é dada a implementação dos efeitos sonoros dentro do jogo também é muito importante. É necessário ir testando ao longo da produção os efeitos sonoros, a medida que estes vão sendo criados - em hipótese alguma deve-se deixar para implementar os *assets* ao final da produção. O áudio do jogo pode ser completamente diferente quando analisado apenas em seus *assets* individuais e quando analisado dentro do contexto do jogo, onde o áudio se torna dinâmico e não há controle sobre como ele será reproduzido. Um bom trabalho de implementação, verificando se todos os sons estão dando *trigger* no momento certo e tendo ideia de antemão sobre o ambiente sonoro e quantos elementos de áudio poderão ser ativados em um único momento.

Após o processo de produção, vem a pós-produção e finalização do projeto. Esta parte cuida da correção de *bugs* e da mixagem do som.

A correção de *bugs*, como o nome sugere, é a procura exaustiva por quaisquer erros nos sons, sejam eles de implementação ou de problemas nos arquivos. Também envolve “abusar” das mecânicas do jogo, de forma a tentar antecipar situações extremas dentro do jogo, como certos comandos sendo repetidos diversas vezes, algo que não aconteceria em um *gameplay* normal.

A mixagem cuida de balancear o volume de todos os assets de áudio presentes no jogo. Quando diversos arquivos de áudio são tocados ao mesmo tempo, o volume de todos eles se somam, podendo criar caos ao longo do *gameplay*, portanto é importante mixar corretamente o som de forma a impedir que isto aconteça. É extremamente desconfortável para um jogador estar jogando e em um momento de ação dentro de um jogo, seu som “estourar” e o volume ficar muito alto. Além de retirá-lo de imersão do jogo, também pode causar danos físicos ao equipamento de áudio (*hardware*) que está reproduzindo o som do jogo.

Com isso, finalizamos o processo geral de *sound design*. No próximo capítulo vamos analisar o primeiro caso de um projeto real e começar a entender quais partes deste processo se mantêm e quais se alteram na produção de um jogo.

Figura 12 – Mixagem.



Fonte: do autor.

## Capítulo 04: Caso 01 – Projeto Stoker

O projeto Stoker foi um projeto relativamente pequeno, em quantidade de *assets*, quando comparado a outros projetos. Entretanto, tendo em vista o universo acústico escolhido para representar os efeitos sonoros presentes no jogo, a experimentação e a pré-produção não se deram de maneira breve. Nele, fui responsável pela parte de *Sound Design* e mixagem através do *middleware* FMOD.

O jogo é um *racing game* sobre trens tendo o *mobile* como plataforma principal e PC como secundário. Ter conhecimento de quais plataformas o jogo será lançado é extremamente importante, pois o Hardware que reproduzirá o áudio irá variar muito para cada plataforma. No caso do Stoker, o foco *mobile* fez com que eu constantemente optasse por sons que tivessem maior destaque nas frequências médias e altas, tendo em vista que os *speakers* de celulares não costumam reproduzir com fidelidade frequências baixas. Outra forma de reproduzir sons graves dá-se por um aumento no volume do som. A dificuldade em reproduzir sons de baixa frequência em *speakers* de celular e, em muitas vezes, *speakers* de baixa qualidade, ocorre devido ao fato de as ondas sonoras consideradas graves serem muito longas e exigirem maior energia para serem escutadas.

Figura 13 – Projeto Stoker.



Fonte: do autor.

Minha participação no projeto aconteceu em um dos últimos momentos do desenvolvimento. O *sound design* começou a ser pensado e realizado após a trilha sonora, o design gráfico e o game design estarem praticamente finalizados. Esta costuma ser uma grande falha no flow de desenvolvimento, e infelizmente, muito recorrente. Tem-se como falha porque, ao incluir o *sound design* apenas ao final do processo, o desenvolvedor acaba cortando grande parte da participação do *sound designer* no desenvolvimento do projeto. Como já sabemos, o *sound design* não é apenas uma adição casual de sons, portanto ao ter um profissional da área atuando desde a fase conceitual do projeto, pode trazer soluções de natureza acústica para diversas dificuldades que costumam ocorrer durante o processo. O *sound design* pode não só atuar como uma solução, mas ele pode agir em conjunto com outras áreas do design de tal forma a fortalecer a experiência do usuário.

Sabendo disso, percebemos que desenvolvedor não usou o seu orçamento da maneira mais eficiente possível, pois deixou de ter as opiniões e sugestões de um profissional de design em seu projeto, o qual estava disponível sem custos extras (algo muito comum entre os profissionais da área).

Ao entrar no projeto, grande parte dele já estava com todos os aspectos definidos, incluindo o direcionamento da parte de áudio. Tendo em vista que o jogo tenta reproduzir um estilo gráfico semelhante ao que se vê em jogos de *GameBoy*, o desenvolvedor optou por uma trilha sonora que acompanhasse essa mesma estética, inclusive replicando a questão técnica de *hardware*, no que diz respeito às limitações técnicas do chip de áudio do *GameBoy*, o qual consistia de apenas quatro layers de áudio, duas das quais necessariamente reproduzindo um pulso de áudio, uma terceira reproduzindo uma *waveform* (formato da onda sonora) e uma quarta reproduzindo noise. Ou seja, a qualquer momento do jogo, a princípio, poderiam ser tocados apenas 4 sons simultaneamente.

Apesar destas definições, o desenvolvedor ainda estava em dúvida em relação a qual seria a solução adotada, em termos acústicos, para os efeitos sonoros. O motivo da dúvida, curiosamente, era dado devido ao fato de o próprio desenvolvedor ter se aventurado na implementação de efeitos sonoros, apenas como *placeholders*, que não eram sons com qualquer relação com *chip tunes*, pelo contrário, eram sons realistas. O desenvolvedor achou a sonoridade interessante e acabou ficando em dúvida sobre qual estilo adotar.

Portanto, menos do que uma restrição, foi-me passado um relato por parte do desenvolvedor e suas incertezas com relação ao direcionamento da estilística dos efeitos sonoros. Isto acabou sendo extremamente benéfico para o desenvolvimento do áudio, pois me permitiu uma maior exploração criativa com relação às possíveis soluções para o jogo, e também um leque maior de opções sonoras que poderia ser utilizadas de tal forma a suprir e auxiliar quaisquer elementos de design do jogo.

Neste momento, foi feita uma pausa no questionamento da estilística, pois antes de definir a paleta acústica, fazia-se necessário definir em quais momentos do jogo deveria-se adicionar sons de modo a aperfeiçoar a experiência do jogador. Para tal, requisitei ao desenvolvedor que me enviasse a *build* mais recente do jogo junto com uma lista de efeitos sonoros que ele próprio imaginava que deveriam estar presentes. Este é um passo muito importante na pré-produção, pois apesar de o *sound designer* ser o profissional de maior conhecimento e autoridade no âmbito dos efeitos sonoros dentro da produção de um jogo, ninguém da equipe tem um conhecimento maior sobre o jogo como um todo do que o desenvolvedor-chefe. Com isso, na lista que é fornecida, é possível verificar vários aspectos de design que não estariam necessariamente claras numa *build* de produção do jogo, muitas vezes por não ter algumas mecânicas e recursos visuais ainda produzidos e implementados.

A lista final de *assets* acaba sendo uma combinação dos esforços e discussões entre o *sound designer* e o desenvolvedor. Um com o conhecimento técnico pleno em áudio e o outro com total conhecimento do projeto. No projeto Stoker, além disso, ainda foi fornecido um esquema de UI/UX que continha todos os elementos do jogo que não necessariamente estavam presentes na *build* atual, mas que estariam presentes na versão final do jogo.

Com essas informações em mente, foi criada uma lista preliminar com todos os *assets* que aperfeiçoariam a experiência do jogador. Essa lista inicial continha 37 *assets* (27 únicos, 10 variações).

A lista não apenas tinha os nomes dos *assets* sugeridos, mas também a descrição de cada um, os eventos sonoros aos quais fariam parte, quais eventos dentro do jogo dariam *trigger* no som, o número de variações necessárias, se irá necessitar de parâmetros, um espaço para referências e outro para comentários do desenvolvedor. Cada um dos itens

da lista é muito importante pois facilita e agilizar o *workflow* e a comunicação entre o *sound designer* e qualquer membro da equipe de desenvolvimento.

Nomear os *assets* utilizando uma lógica sequencial, em vez de utilizar um “nome título” para cada efeito sonoro acelera, o processo de produção pois deixa muito claro a qualquer pessoa que vê a lista a lógica por trás da nomenclatura e organização dos arquivos, desde os primeiros momentos do desenvolvimento do áudio. Como o projeto Stoker não foi muito extenso, a organização dos nomes dos *assets* não tem um impacto tão grande, mas ao passarmos para um projeto com uma quantidade um pouco maior de *assets*, já fica um pouco mais difícil de manter um número maior de *assets* organizados.

A descrição dos sons também é muito importante, pois muitas vezes, ao analisar o jogo pela primeira vez e tentar dissecar todos os momentos em que se julga necessário ter um elemento sonoro, desenvolvemos determinados raciocínios complexos que justificam certos sons e que às vezes temos dificuldades de recuperar e/ou lembrar em um momento posterior, e vai ficando cada vez mais difícil a medida que o número de *assets* começa a aumentar. Uma breve descrição facilita muito futuras visitas de qualquer integrante do projeto, ao facilitar a compreensão e localização de cada *sound asset*.

O item “os eventos sonoros aos quais fariam parte”, é uma sugestão que o *sound designer* deixa para os programadores como uma forma de dizer que diversos *sound assets* fazem parte de um único evento sonoro. Especificamente no caso do Stoker, temos um *asset* para “carregar” o carvão (encher a pá de carvão ao segurar o dedo na tela), um outro *asset* para a ação de arremessar o carvão e um terceiro *asset* para o carvão sendo queimado na fornalha, ou seja, temos três elementos sonoros, entretanto, todos fazem parte do evento “jogar carvão”. Este tipo de informação facilita as coisas para o programador, o qual passará ter uma ideia melhor de como os elementos sonoros serão representados durante o *gameplay* e como os *assets* reagem às mecânicas do jogo.

A definição de quais eventos dentro do jogo vão acionar cada som também é importante. A ideia neste momento é ser o mais específico possível ao invés de falar em termos gerais. A razão por trás disso, é que ao definir em palavras exatas o momento do *trigger*, o programador saberá implementar o efeito sonoro no momento certo da ação ou animação. É bem comum por parte de programadores sem muita experiência com a parte de áudio ficarem confusos com o momento correto de o *asset* ser acionado, pois muitas vezes ele irá tentar sincronizar com a informação visual e por não ter acuidade em *sound*

design, acabará colocando o áudio levemente fora de sincronia com a parte gráfica. Como bons produtos finais se constroem nos detalhes bem-feitos, uma quantidade grande de sons pouco fora de sincronia acabam se tornando muito evidentes, não só desfavorecendo a estética como um todo, mas atrapalhando a jogabilidade, tendo em vista que um dos sentidos do jogador estará falho.

No geral, é bem difícil definir o número de variações de um determinado asset antes de escutar alguma versão deste mesmo asset em ação no jogo. Na natureza e no nosso dia a dia como um todo, sons iguais ou muito semelhantes não são recorrentes. Até mesmo os elementos que num primeiro momento se parecem, acabam por possuir alguma característica distinta entre suas várias "versões", dada a grande variação dos sons que nos rodeiam e de como eles interagem com o ambiente, o qual muitas vezes está em constante mudança. Por ter isso em mente, um *sound designer* costuma querer adicionar uma grande variedade de sons, mas acaba não sendo tão simples assim. Há algumas razões que justificam tal dificuldade. A primeira e mais óbvia, é que um grande número de assets muitas vezes se torna inviável em termos de armazenamento e processamento. A segunda é que devemos considerar a existência do áudio dinâmico por meio de middlewares ou da própria engine do jogo, na qual podemos adicionar pequenas automações em algumas características sonoras, tais como volume e pitch, de tal forma que acabamos randomizando e gerando por meio de análise combinatória, diversas variações. É verdade que elas ainda assim soam parecidas com o asset original, mas essa medida já é de grande ajuda e consegue amenizar consideravelmente a percepção de repetição de uma mesma fonte sonora.

No caso do Stoker e em muitos jogos que são produzidos, é comum o *sound designer* adotar como primeiro passo criar um único som para a maioria dos *assets*, em seguida tentar amenizar o efeito de repetição por meio áudio dinâmico e por último chegar a conclusões sobre quantos sons de fato deverão ter variações. Obviamente, num primeiro momento precisamos definir um número de variações que imaginamos que deverá se aproximar da realidade por motivos de definição contratual. Essa percepção inicial muitas vezes vem junto com a experiência de anos de trabalho, mas a quem falta experiência profissional, deve-se utilizar a experiência pessoal e atualizar a tabela com os números corretos assim que possível.

A questão dos parâmetros também visa auxiliar o programador, mas também pode-se considerar como o primeiro momento criativo do efeito sonoro. É um dos primeiros

momentos no qual imaginamos o áudio sendo tocado junto ao jogo e como ele exatamente deve reagir. Os parâmetros servem para quando se faz necessário um som responsável a um evento visual e não como uma ação de *trigger* único. No caso do Stoker temos como exemplo o som de carregamento do carvão na pá, o qual depende de por quanto tempo o jogador está apertando o dedo na tela. No caso, pode-se criar um parâmetro de tempo que é relacionado a uma alteração de *pitch* do *asset*. Por quanto mais tempo o jogador pressionar o dedo, o áudio irá aumentando o seu *pitch* e caso o jogador solte o dedo, o áudio poderia ir diminuindo o *pitch* e caso o jogador pressione novamente o dedo contra a tela, o *pitch* poderia voltar a aumentar a partir do ponto que estava antes, ou seja, de forma dinâmica e responsiva, em contrapartida a um evento estático o qual simplesmente teria dado o *trigger* do início do arquivo de áudio, perdendo assim a sensação de continuidade.

O espaço reservado para referências é de grande serventia à comunicação entre o *sound designer* e o desenvolvedor, e essas referências podem ocorrer de diversas formas. É possível que um determinado evento do jogo que com certeza terá um sound asset associado a ele ainda não esteja disponível ou sequer tenha sido criado, já que o jogo ainda não foi completamente finalizado. Neste caso, o desenvolvedor pode referenciar diversas imagens de outros jogos que expressem uma sensação semelhante à sensação final do asset gráfico que estará presente no jogo. O desenvolvedor pode também acrescentar referência sonoras de outros jogos e até mesmo outras mídias, neste espaço. É extremamente comum uma situação na qual o desenvolvedor não consiga expressar verbalmente o som que desejaria para determinado evento, e acaba por fazer referência a outro som que tenha alguma semelhança, cabendo ao *sound designer* o trabalho de tradução e compreensão da visão do desenvolvedor para com o jogo.

Por último temos o espaço para comentários. Ele costuma ser mais utilizados em projetos realizados remotamente, mas também pode ser muito útil quando o projeto é realizado num mesmo ambiente. A quantidade de informação que é passada e repassada entre o desenvolvedor e o *sound designer* costuma ser imensa e é muito fácil esquecer ou perder registros anotados casualmente. Um espaço próprio para essas anotações ajuda muito na comunicação e no fluxo do desenvolvimento, pois evita que ideias e feedbacks sejam perdidos ao longo de longas conversas, e-mails e conversas de softwares como Skype, Slack, etc.

Após esta lista inicial proposta pelo *sound designer*, vem o calendário do projeto.

Curiosamente, o prazo dado para finalizar o Stoker não foi muito curto: aproximadamente um mês e meio. Isso permitiu que toda a investigação (pré-produção) e desenvolvimento fossem realizados com calma, permitindo assim resultados mais precisos e satisfatórios, tendo em vista que muito do processo de *sound design* se dá por meio de exploração dos processos criativos. Nesta parte foram definidas as datas de entrega de cada *batch* do projeto e os prazos para revisões.

Com toda esta parte inicial definida, o contrato entre empregador e empregado foi finalmente firmado. O número de *assets*, o valor negociado (o qual não cabe a discussão neste momento), os prazos e os pormenores legais (que também não vêm ao caso) são as informações mínimas necessárias para que se possa assinar o contrato.

Tendo em vista a dimensão do projeto Stoker e do tempo disponível para a sua realização, foi possível realizar uma pré-produção um pouco mais extensa que o de costume, algo que foi muito benéfico para o jogo, pois permitiu que tanto eu, o *sound designer*, quanto o desenvolvedor pudessemos investigar, dialogar e procurar as soluções acústicas e estéticas apropriadas para a proposta do jogo, a qual o próprio desenvolvedor estava em dúvida desde o início do processo de *sound design*.

O primeiro passo foi ir em busca do óbvio: procurar por jogos de *GameBoy* ou com uma limitação de até 16 bits que tivessem cenas marcantes de trens ou, melhor ainda, fossem com a temática de trem. Este foi um momento muito curioso, pois desmanchou completamente nossas expectativas. A maioria dos jogos que averiguamos, mal tinham sons relacionados aos trens que se faziam presentes. O elemento mais forte do jogo Stoker se passava praticamente despercebido. Após pensar e realizar testes com sons do próprio chip do videogame, foi possível confirmar o porquê de isso acontecer. Primeiro, há a limitação do próprio som do console: tendo em vista que o *GameBoy* reproduzia apenas quatro layers de sons simultaneamente, era inviável manter uma layer ocupada constantemente com o som de um trem. E em segundo, havia a questão de como o som do trem deveria ser representado. Pelo fato de a característica do som de um trem ser extremamente repetitivo e dada a paleta de sons disponíveis em chip *sounds* antigos, qualquer elemento sonoro acabaria ficando muito em evidência e causaria um grande cansaço ao usuário em um curto tempo, comprometendo assim a experiência do jogo.

Após testar algumas diferentes opções e variações, acabamos optando por um meio termo na solução: os efeitos sonoros seriam uma mistura de sons reais (e outras fontes que não fossem necessariamente até 16 bits) e sons com a característica do *chiptune* clássico. Dessa forma, os elementos reais e mais atuais poderiam amenizar a sensação de desgaste em momentos repetitivos, ao mesmo tempo que os sons possuiriam elementos de *chiptune*, mantendo assim, não apenas uma harmonia com o visual, mas também com a trilha sonora (feita por outra pessoa), a qual obedecia as limitações do *GameBoy*.

Com a parte da pré-produção finalizada, passamos então adiante com a produção. O grande desafio da produção em sons de 8 ou 16 bits, consiste entre escolher uma representação fiel e real dos elementos visuais ou de tomar a liberdade criativa e criar um som um pouco mais distante, dada a impossibilidade de representar as características de um determinado som no *chip*. Por exemplo, no jogo Pokémon, temos que os sons dos próprios pokémons são representações abstratas de sons de animais, tendo em vista que apenas poucas características de sons de animais tinham como ser representadas através de *chip sounds*.

Apesar de parecer que estas escolhas deveriam ter sido feitas durante a pré-produção, elas se encaixam melhor na produção em si, pois cada efeito sonoro necessita de uma análise única e específica e que só é possível em meio a testes, ou seja, na própria produção em si.

Vamos analisar a produção de acordo com os blocos principais da UX/UI do jogo: o menu principal (também chamado de lobby), a garagem (onde acontece toda a customização dos trens e tela pela qual se passa, necessariamente, antes de começar a corrida em si, para que o jogador escolha qual trem será utilizado na corrida), a tela de busca por outros jogadores, a corrida em si e o resultado da corrida (esta última, podendo fazer com que o jogador retorne ao menu principal, à corrida - como uma opção de restart - e à garagem).

No menu principal do jogo, temos presentes sons de cliques de abertura para outros menus e outras telas. Para estes sons, ficou combinado com o desenvolvedor, de que usaríamos o mesmo som de click para a maioria dos botões, com exceção de cliques que levassem a mudanças entre os blocos principais da UX/UI e cliques que levassem à opção de compras dentro do jogo. Ao que diz respeito entre as mudanças de blocos, os efeitos sonoros têm como papel, muitas vezes, reforçar determinadas ações realizadas

pelo jogador e como ele deve se sentir durante a experiência que o leva do menu principal à tela do jogo em si. Se analisarmos jogos como Mario Kart e Super Smash Bros (ambos da Nintendo), é possível perceber que existe uma progressão dos sons entre cada tela saindo do menu principal, passando pela seleção de mapas e jogadores, até a chegada na parte de *gameplay*. Cada som subsequente, saindo do menu principal, é um efeito sonoro cada vez mais elaborado, que vai crescendo em número de elementos e layers, muitas vezes feito com elementos musicais, seguindo uma progressão numa escala, tendo como momento final, ao clique de ir para a tela do *gameplay*, o que seria a resolução sonora. Uma forma mais clara de entendermos este conceito, é ao pensar em uma sequência de acordes numa música e o acorde final que fecha a música, sendo aquele que dá uma sensação de “alívio” e finalização. Este é o mesmo conceito aplicado em diversas progressões em jogos, e no caso que estamos analisando, na progressão de menus.

Com isto em mente, criei sons que não apenas representassem a entrada em uma determinada tela em si, mas que tivessem uma representação em um contexto maior, considerando todas as telas que levam ao *gameplay*.

Já para a parte de compras dentro do jogo, a tela de shopping carrega uma das maiores responsabilidades do *sound design*: a de criar uma sensação e uma resposta psicológica de prazer, vinculada ao ato de comprar algo. A experiência como um todo deve ser prazerosa e carregar em si uma positividade que seja muito evidente ao jogador. Da mesma forma que shopping centers na vida real devem ter uma temperatura adequada, e muitas vezes um cheiro específico (em carros novos o cheiro é bem evidente, por exemplo), o *sound design* ser capaz de proporcionar uma representação acústica que faça o usuário se sentir recompensado por estar tomando a decisão de entrar no shop do jogo. Para tal, utiliza-se elementos musicais (escalas maiores que trazem em si um sentimento de alegria e conforto), sons que representem dinheiro (tais como moedas se chocando e caixas registradoras) e também a sensação de progressão entre o ato de entrar no shop e o ato de realizar a compra (esta última devendo ser a mais recompensadora para o jogador). Existe também a possibilidade de ter um som específico para a desistência da compra, com o intuito de o jogador se sentir desapontado consigo mesmo por não finalizar a compra, entretanto, esta última não foi aplicada no Stoker, tendo em vista que sua interface é mais simples e um som específico para o

cancelamento de uma compra poderia ficar muito evidente ao usuário, algo que poderia ser visto com maus olhos pelo jogador.

Vídeo 1 – Efeitos sonoros de compra e *upgrade*.

<https://www.dropbox.com/sh/j58jsfr1yvhi01y/AADNE09HXZHpQ5HcUR5-3lr8a?dl=0>

Fonte: do autor.

Para finalizar o bloco do menu principal, um som que acabou sendo deixado de fora do jogo, por escolha do desenvolvedor mas que, em minha opinião, deveria ter sido incluído, foi o efeito sonoro de volta dos menus. Como todos os cliques de menu são iguais, com exceção dos que foram mencionados previamente, a falta de um som que diferenciasse o clique de ida e o de volta, faz muita diferença no *flow* das telas. É um detalhe relativamente pequeno, mas que ajuda a criar uma sensação de localização ao jogador, em qualquer momento que ele precise retornar de alguma outra tela ou menu.

Para o segundo bloco, temos a tela da garagem, onde é possível customizar os trens, fazendo melhorias em sua performance ou comprar skins (pinturas) de outros modelos de trem. Nesta tela, temos alguns elementos visuais que pediram uma resposta acústica específica para criar uma melhor experiência de navegação, sendo estes, cada um dos elementos de seleção que poderiam receber um upgrade ou uma melhoria e as skins dos trens.

Os elementos de upgrade são "parte" do trem, sendo elas: motor, fornalha, suspensão, pá e rodas. Em um primeiro momento, o desenvolvedor se mostrou relutante em ter um som específico para cada uma das seleções, pois achava que seria desnecessário, sendo apenas um som o suficiente para representar todas as partes. Apesar de ser viável a utilização de apenas um efeito sonoro, principalmente, levando em conta a estética do jogo, a utilização de um som para cada parte torna a experiência do usuário mais rica e completa, principalmente se levarmos em conta a paleta de sons limitada pelo *chip* do *GameBoy*, sendo assim mais interessante mostrar uma variedade de efeitos sonoros sempre que houve abertura para tal, para proporcionar uma maior dinâmica e mais movimento aos elementos do jogo.

De fato, é bem difícil criar sons que, utilizando em sua maioria chip sounds, apresentem características muito distintas entre si, e no caso, que conseguissem ao mesmo tempo representar e diferenciar cada uma das partes do trem.

Para o motor, foi utilizado como base o som de um motor de trator dando partida. Apesar de na realidade um trem não apresentar esse tipo de motor, o som da ignição de um carro é extremamente característico e portanto representa bem o ícone de um motor.

Na fornalha foi utilizada uma mistura entre sons de fogueira, para representar o ato de queimar algo, e uma camada de noise longo de *GameBoy* que também representa a característica de "chiado" que ouvimos no ato de chamuscar, semelhante ao som de uma grelha.

Para a suspensão, foi utilizado majoritariamente um *chip sound* que alternasse rapidamente entre uma nota de frequência baixa e outra alta, criando assim a sensação de pulo e alternância, o qual condiz com a representação visual de uma suspensão, que carrega em si a sensação de pulo e elasticidade.

Para as rodas, uma mistura entre um som real de um *whoosh*, como por exemplo, o som do ar sendo cortado por uma corda sendo rotacionada, o qual traz em si a característica de rotação junto a um som 8-bits grave, curto e repetitivo para acentuar a representação de uma roda girando.

Para o som da pá, utilizou-se uma mistura de sons reais de uma pá cavando na neve, o qual é extremamente característico, junto a sons curtos de noise, que também soam similar a uma pá cavando carvão. Neste caso, como os dois sons são semelhantes, eles se adicionam de tal forma que o som do 8-bits prevalece porém tendo o som da pá real como reforço em sua composição, facilitando a sua compreensão.

Por último, o som da pintura consiste de uma layer de som de uma escova com cerdas grossas em uma superfície áspera, junto a uma camada de noise longo, em uma ideia semelhante ao som do pá, onde a característica de semelhança entre ambos os sons é tamanha que os tornam sinônimos, prevalecendo o som de *chip* e trazendo em si um toque de realidade que ajuda em sua distinção.

Ainda na garagem, temos também o efeito sonoro referente ao ato de fazer upgrade em cada uma das peças. Novamente, ele segue a linha de raciocínio dos elementos sonoros que devem proporcionar uma sensação de satisfação ao jogador sempre que a ação for realizada. Assim sendo, sempre que o jogador realiza um upgrade em uma peça, ele tem um som de uma escala maior, numa sequência de notas ascendentes, de tal forma a consolidar a ação de upgrade. No Stoker, cada peça pode receber até 5 upgrades e portanto o quinto upgrade acaba sendo o último. Uma forma de incentivar o jogador a

querer realizar todos os upgrades disponíveis, é recompensando a sua ação de chegar ao último upgrade com um efeito sonoro específico para aquele momento, deixando bem claro ao usuário de que ele está avançando no jogo e tendo atitudes positivas.

Passando em frente, a próxima tela é a de busca de partida, na qual basicamente o jogador deve esperar encontrar um adversário para jogar contra. Neste caso, a música tem um papel mais forte do que os demais efeitos sonoros, pois a busca por jogadores muitas vezes pode ser lenta, então a música deve ser agradável de ouvir e deve ser animada o suficiente para deixar o jogador animado para que a partida comece logo. Além disso, ela deve ser longa o bastante para não ser cansativa caso ela precise entrar em loop. Com relação aos efeitos sonoros, os mais importantes para a tela são o de oponente encontrado e o de cancelar a busca. Estes sons devem manter um contraste evidente entre si, quase como antônimos sonoros. O som de cancelar a busca, neste caso, foi composto por duas notas tocadas simultaneamente em um intervalo de segunda menor (o qual representa tensão) e também uma layer com um áudio de arranhão, com a frequência modulada para baixo, soando como se fosse um glitch, erro. Em contrapartida, o som de busca por jogador sucedida é um som bem positivo, novamente tendo elementos musicais como sua base principal.

Finalmente chegamos na tela de *gameplay*. A dinâmica e mecânica do jogo são bem simples: há uma contagem regressiva e ao fim dela o jogador deve tocar na tela repetidas vezes para jogar carvão na fornalha e fazer com que o trem ganhe velocidade. Entretanto, ele deve tomar cuidado pois se ele jogar mais carvão do que o trem aguenta, ocorre um superaquecimento e o trem perde velocidade. O jogador tem a opção de dar diversos cliques curtos e rápidos ou então segurar um pouco o dedo na tela, a fim de carregar a pá com mais carvão e portanto ganhar mais velocidade de uma única vez.

Vídeo 2 – Progressão horizontal das telas.

[https://www.dropbox.com/sh/ahim7f7wekiyict/AADuIIIRSoURE\\_npdksV3O8KJa?dl=0](https://www.dropbox.com/sh/ahim7f7wekiyict/AADuIIIRSoURE_npdksV3O8KJa?dl=0)

Fonte: do autor.

Primeiro, devemos listar os sons presentes: temos a sequência de sons da contagem regressiva, a pá cavando em cliques curtos e em cliques longos, o carregamento do pá, o carvão sendo queimado em poucas e grandes quantidades, o aviso de superaquecimento e a movimentação do trem em si.

Desta lista, o mais fácil de ser criado é a contagem regressiva. Culturalmente conhecida por uma sequência sonora de 3 beeps curtos e idênticos e 1 beep longo, uma oitava acima, tal efeito sonoro é facilmente recriado em sons de GameBoy e soa perfeitamente natural ao jogador, tendo em vista que tal contagem já se fazia presente em jogos de eras anteriores a de 8-bits. Além disso, historicamente, sua característica acústica se dá por meio de teclados sintetizadores, ou seja, sons eletrônicos, representação muito mais confortável frente a estética pixel, em oposição à dificuldade encontrada quando tentamos recriar sons mais orgânicos.

Os sons de carvão foram um processo de tentativa e erro. De início, o importante era criar variações, tendo em vista que os sons seriam repetidos diversas vezes ao longo do jogo, portanto, de forma alguma, poderiam cansar o ouvido do jogador. Mesmo antes de os sons serem criados, esse foi o primeiro pensamento em mente. Com relação à produção dos sons em si, o desafio era diferenciar cavadas curtas e longas, e queimadas curtas e longas. Após tentar algumas alternativas, percebi que sons referentes à ações curtas e rasas têm sua qualidade acústica por volta de frequências altas, enquanto ações longas e de peso se associam bem a sons mais graves. A explicação por trás disso se deve ao fato de que sons graves possuem ondas sonoras mais longas e portanto demandam mais energia para serem reproduzidas, frente a ondas curtas que são reproduzidas com maior facilidade, com menos “esforço” (Designing Sound, 2010), portanto, faz sentido associarmos ações mais robustas a sons graves e ações leves a sons agudos.

Um erro que teve de ser corrigido para que os efeitos sonoros de cavada funcionassem bem, foi referente à programação de áudio. Em uma primeira versão, os sons eram resetados a cada toque na tela, isto é, os sons eram interrompidos antes de serem reproduzidos em sua totalidade. Isto fazia com que os sons parecessem que estavam com algum problema. Ao mudar o código e permitir que os sons fossem completamente reproduzidos, independente da velocidade da ação de pressionar a tela por parte do usuário.

Outra questão que teve que ser resolvida foi referente ao equilíbrio entre os sons de cavar e de queimar o carvão. Em termos de design, um som era a resposta para o outro, portanto eles deveriam soar de tal forma que se completassem. Na primeira tentativa, ao ter ambos os sons com um volume próximo, eles acabavam confundindo os jogadores, pois ao clicar rapidamente na tela, não ficava claro de onde cada som vinha. Ao abaixar o volume do som do carvão queimando, o problema foi resolvido. A razão por trás disso, é

que o segundo som acabava criando uma sensação de eco, como se ele fosse legitimamente uma resposta do segundo som. Pelo fato de a animação do jogo consistir de um personagem arremessando o carvão à fornalha, o segundo volume baixo também representava a sensação de distância entre os elementos.

O som de carregamento da pá foi relativamente simples. Ele deveria ser um som que permitisse um loop, pois não há controle por parte do jogo de por quanto tempo o jogador irá pressionar a tela. Como observado anteriormente, sons de pá cavando são bem recriados através de uma layer de noise 8-bits. Este som apresenta naturalmente uma característica de loop, então bastou utilizá-lo como a layer principal e de maior destaque do efeito sonoro.

Por último, temos o som de superaquecimento. Por ser um som de aviso de que o jogador errou, ele deve soar quase como um som punitivo e não apenas repreensivo. A intenção é fazer com que o jogador sinta desconforto sempre que errar, fazendo com que ele tente evitar cometer tais erros. Semelhante ao som de falha na busca por outros jogadores, este efeito sonoro consiste de uma layer musical com um intervalo de segunda menor, uma outra layer com característica, áspera, arranhada (também chamado de glitch) e por último um som de pane mecânica, similar a engrenagem de um motor travando. Além disso, é importante que o volume do efeito sonoro seja um pouco mais alto que o som dos demais, de tal forma que "corte" a mix e seja certeza absoluta de que o jogador escutará o som repreensivo.

Vídeo 3 – Efeitos sonoros na tela de *gameplay*.

<<https://www.dropbox.com/sh/ew9aor65o4dh62k/AADGND9uzGDJnYC5PNmOdl97a?0>>

Fonte: do autor.

O último grande bloco de telas é a tela de final da partida. É uma tela simples com 3 opções disponíveis: a de voltar para o menu principal, a de retornar para a garagem e a de jogar novamente. Tendo apenas 3 opções de clique, todos utilizam o mesmo som de clique regular do menu principal. Em um primeiro momento, foi considerado utilizar os mesmos efeitos sonoros referentes ao clicar para entrar em cada uma das telas de acordo com o flow original de UX/UI, entretanto, estes mesmos sons acabam ficando fora de contexto se utilizados novamente na tela de fim de jogo. Como foi colocado anteriormente, o som para cada uma das partes mencionadas é fruto de uma sequência da progressão de telas, sendo o último (que antecede o *gameplay*), o efeito sonoro mais

positivo, a fim de animar o jogador para a partida. Se o mesmo fosse reutilizado na tela de fim da partida, quando o jogador clicasse pra jogar novamente, por exemplo, ficaria muito evidente que o jogo estaria tentando forçar tal positividade, portanto, acaba sendo mais natural utilizar apenas o som de clique regular. Os sons de maior importância para a tela de final de partida, no caso, são temas musicais de derrota e vitória, sempre lembrando que o tema de derrota não deve soar entristecedor a ponto de desmotivar o jogador, mas sim esperançoso ou até mesmo animado, a fim de mantê-lo interessado em jogar uma próxima partida.

Finalizamos então, o caso do projeto Stoker.

## Capítulo 05: Caso 02 – Projeto Nova Empire

O segundo caso que iremos analisar será o jogo Nova Empire. Diferente do projeto Stoker, este projeto foi aproximadamente 4 ou 5 vezes maior em termos de número de *assets* e também contou com a presença de Voice Over, então sua produção ocorreu de forma diferente a do projeto Stoker. Neste projeto também fiquei responsável pela parte de *Sound Design* e mixagem de som, entretanto, a implementação não foi realizada diretamente por mim e como mencionado anteriormente, também houve necessidade de uma parte de dublagem.

Nova Empire é um jogo sobre exploração espacial, onde o jogador controla e melhora sua frota enquanto coleta recursos, forma alianças e batalha outros jogadores online. É um jogo completamente voltado para o mercado - pode-se dizer que é uma versão espacial de jogos como Clash of Clans e Travian.

Figura 14 – Projeto Nova Empire.



Fonte: do autor.

O início do processo se deu de uma forma um pouco diferente. O primeiro contato com o desenvolvedor foi referente apenas à questão da dublagem. Ou seja, o desenvolvedor já fez um primeiro contato com uma lista de em quais *assets* o voice over seria necessário e com os textos já redigidos.

O primeiro passo foi a correção dos textos. Da perspectiva do *sound designer*, ele deve analisar o texto que será dublado e considerar em qual situação o *asset* será utilizado. No caso do Nova Empire, os áudios seriam utilizados como mensagens de aviso no jogo, portanto era importante levar em consideração a extensão de cada frase, possíveis cacofonias, encontros consonantais abruptos ou finais ou inícios de palavras que impedissem que a frase fosse pronunciada com fluência e clareza. Uma mensagem de aviso deve ser transmitida de imediato, e soar quase como um efeito sonoro, a ser interpretado não necessariamente pela linguagem em si, mas pelo aspecto sonoro. Como consequência, a intonação da frase também é muito importante para a transmissão da mensagem. Uma frase referente a um evento comum no jogo deve soar sem exaltação, enquanto uma mensagem de batalha deve soar com tom de animação.

Após as correções dos textos que seriam dublados, eu e o desenvolvedor começamos a conversar sobre os momentos do jogo onde eles seriam colocados, pois ele gostaria de ter uma segunda opinião. Esta conversa acabou levantando a questão dos efeitos sonoros, onde o desenvolvedor expressou a necessidade de ter alguém que os fizesse e desse direcionamento ao áudio do projeto (com exceção da trilha sonora que já estava definida).

Porém, antes de discutirmos os efeitos sonoros, faltava definir o ator que faria a dublagem. Primeiro, tivemos que definir se a voz seria feminina ou masculina. Para tal, fiz uma análise dos textos: todos retratavam um diálogo entre um operador de voz com o seu comandante (no caso, o jogador), portanto deveriam ser levados em consideração os níveis de hierarquia entre o locutor e o ouvinte. Após pesquisar algumas referências, historicamente em filmes e jogos, a maioria da tripulação em naves espaciais são pessoas do sexo masculino, e quando há mulheres presente, elas costumam ocupar cargos de baixa hierarquia. Após conversar com o desenvolvedor, achamos que seria interessante mudar um pouco esta perspectiva e utilizar apenas vozes femininas, independente do grau de hierarquia dos personagens. Não há exatamente um motivo lógico por trás desta escolha, a não ser de propor algo diferente. Também achamos que

caso fosse uma escolha "errada" do ponto de vista comercial, ela não teria um grande impacto.

A escolha da atriz se deu de forma muito simples: eu já tinha uma lista de contatos de dubladores e deste, eu já conhecia uma atriz acostumada a fazer vozes de aviso em tripulações, portanto foi bem fácil organizar a gravação.

Com a parte do voice over resolvida, passamos então a pensar nos efeitos sonoros. Como de costume, pedi ao desenvolvedor uma lista com os sons que ele imaginava que precisariam estar no jogo e uma *build* de desenvolvimento do projeto. A lista inicial do desenvolvedor tinha cerca de 20 efeitos sonoros. Ao final das revisões e conversas que tivemos, chegamos a mais de 100 *assets*. Esta diferença considerável reforça a importância de um *sound designer* ter voz ativa em um projeto, podendo contribuir com o seu desenvolvimento.

Em sequência, com contratos assinados, passamos a fase de pré-produção, que neste caso, devido à limitação de tempo, acabou sendo bastante curta. A pesquisa foi dividida entre jogos de estilo semelhante ao Nova Empire, e filmes e outros jogos que se passavam em naves espaciais, dentro de um contexto de exploração espacial. Em geral, sons de naves espaciais são praticamente sons eletrônicos (proveniente de sintetizadores), sons de maquinaria pesada e sons de armas e explosões (tendo em vista que parte do jogo envolve batalhas espaciais). É importante frisar que no espaço, devido ao vácuo, não há propagação de som, entretanto, seria extremamente danoso ao jogo e todo o seu design se o "correto" fosse seguido à regra. Portanto, não apenas deste ponto de vista do design, mas também do entretenimento e experiência como um todo, entende-se a necessidade de quebrar tais regras e criar um mundo de sons onde eles não deveriam estar presentes.

Uma curiosidade em relação ao Nova Empire, foi a decisão por parte do desenvolvedor de não ter variações nos sons, mesmo aqueles que seriam repetitivos. Muito disto se deve ao fato de que ao invés de criar variações, ele deu preferência ter sons completamente diferentes para cada uma das partes do jogo. Entretanto, em momentos como os de batalha, onde sons de tiros e explosões são ouvidos diversas vezes, ele optou por utilizar variações dentro da própria engine do jogo, modulando o *pitch* e o volume dos efeitos sonoros.

Passando para a parte de produção, diferente do projeto anterior, o Nova Empire não possui exatamente um *flow* definido de UX/UI. Pode-se dizer que todas as telas têm um nível de hierarquia e importância equivalentes, portanto não houve necessidade em se preocupar com a progressão dos sons, exceto em momento óbvios que deveriam representar algo positivo ou negativo para o jogador.

Os efeitos sonoros foram divididos por tela, sendo estas: tela de carregamento, tela de compra, popups de sim ou não, menus, submenus, chat, sistema(solar), galáxia, estação espacial, detalhes da estação espacial, construções, batalha e sons que poderiam estar presentes em quaisquer telas.

Como é possível notar, o escopo deste jogo foi consideravelmente maior que o do jogo anterior. Tendo em vista a quantidade de telas possíveis, era importante fazer com que cada efeito sonoro de clique fosse único, para dar uma sensação de localização melhor ao jogador que pode se sentir sobrecarregado de informações nas primeiras vezes em que for jogar. Ou seja, o *sound design* tem um papel fundamental em atenuar este primeiro choque entre um jogador e um jogo de maior complexidade. Em outras palavras, sons de transições únicos ajudam o jogador a criar um mapa mental das telas do jogo e navegar entre eles com maior facilidade. Isto também ajuda a criar uma sensação de conexão do usuário com o jogo, algo extremamente importante, tendo em vista o caráter comercial do jogo. Tal sensação acaba criando um sentimento de comprometimento por parte do jogador, fazendo com que ele tenha a sensação de que está melhorando no jogo rapidamente, construindo algo naquele universo, criando quase uma dependência por parte do jogador.

Vídeo 4 – Transição entre telas.

<<https://www.dropbox.com/sh/ru85bbskr158mkc/AAAybOvTk-ruyz9Y8P2zxbwAa?dl=0>>

Fonte: do autor.

Novamente, dado o caráter de mercado do jogo, um som extremamente importante foi o de compra de boxes do jogo e sua abertura, a qual revela algum item ou bônus que serve para beneficiar o jogador. Um estudo chamado Skinner box, realizado por B.F Skinner em 1930, constatou que o estímulo psicológico e de prazer para uma pessoa é muito maior quando ela não sabe o que esperar do que está por vir, por isso o conceito de comprar uma caixa que irá revelar um item aleatório é muito mais interessante do que a compra de um item específico, onde o jogador já sabe o que irá receber. Sabendo disso, foi muito

importante não apenas ter um som positivo para a compra de gold dentro do jogo (unidade monetária dentro do jogo que permite ao jogador que este compre as caixas) - seguindo toda a questão de uma layer musical em notas ascendentes dentro da escala maior, assim como no projeto Stoker, mas o som principal e de maior importância acabou sendo o de abertura da caixa. Era muito importante ter um som extremamente gratificante ao jogador, de tal forma vincular a surpresa de não saber qual item ele irá receber, a um *asset* sonoro que também traz essa sensação em si.

Vídeo 5 – Abertura de pacotes.

<https://www.dropbox.com/sh/jil32yrm3ppcu99/AAB11Y6cc-45XjDBfVlyJ8A2a?dl=0>

Fonte: do autor.

Os sons de batalhas, apesar de não terem muitas variações para cada som, tiveram diversos sons específicos de cada unidade de batalha do jogo, ou seja, cada nave espacial deveria ter um som único para suas armas e suas explosões. Este processo foi particularmente difícil, pois não foi fácil criar sons que de fato soassem diferente mas tivessem uma característica única em si.

Parte da dificuldade se dá ao fato de ter sons originais a partir dos quais é possível trabalhar para diferenciar os sons. Dada a impossibilidade de gravar sons de explosões, eu precisei utilizar sons de bibliotecas de terceiros como ponto de partida, e quando há uma grande quantidade de *assets* a serem produzidos, os sons destas bibliotecas de áudio "acabavam" muito rápido.

No geral, a solução foi criar sons modulados e fazer bastante uso da questão de que os sons devem contar uma história. Como cada som deve ter um início, meio e fim, os sons de tiros e explosões possuíam esta característica, portanto foi feito um trabalho criando sons modulares e fazendo uso de análise combinatória, onde poucos sons combinados, eram capazes de gerar uma grande variação de sons únicos. Um detalhe interessante, era que os próprios *assets* visuais não eram muito diferentes entre si, então acabou sendo de responsabilidade do *sound design* proporcionar uma maior distinção entre os eventos.

Vídeo 6 – Exemplo de batalha.

<https://www.dropbox.com/sh/29no5kwjrus9c0o/AAAS68BSqll-hwxriwd81kZRa?dl=0>

Fonte: do autor.

Por último, chegamos a parte de pós-produção, na qual, erroneamente, os sons foram implementados pela primeira vez e ocorreu a mixagem. Por escolha do desenvolvedor de não trabalhar com *middlewares* e nem permitir que eu tivesse acesso ao projeto dentro da engine, ficou fora do meu controle a parte da implementação do som. Como foi dito nos capítulos anteriores, é extremamente importante ir implementando os sons a medida que eles vão sendo criados, algo que não foi possível neste projeto. A consequência da falta de controle por parte do *sound designer* veio rapidamente: nas cinco primeiras builds de versão beta do jogo, em nenhuma delas todos os sons que foram produzidos já estavam presentes e devidamente implementados ao mesmo tempo que o próprio desenvolvedor havia confirmado que todos os sons já haviam sido implementados, o que pode ser considerado uma falha gravíssima de produção. O que aconteceu foi que, devido ao grande número de sound assets, o desenvolvedor acabou se confundindo e desorganizando em relação a quais sons eram os que eu havia produzido e quais eram os placeholders que estavam presente em builds de desenvolvimento. Ainda que para mim a diferença fosse gritante, para o desenvolvedor, que não está acostumado com o processo de *sound design*, ele deixou por assim mesmo. Tal erro pode ser considerado gravíssimo, pois se eu não tivesse conferido todos os sons do jogo, muitos deles estariam hoje em sua versão final, o que seria péssimo para a minha imagem pessoal, para a imagem do jogo e a sua qualidade e por consequência, sua posição no mercado e retorno comercial.

Finalizamos assim, a análise do projeto Nova Empire.

## Capítulo 06: Considerações

Após analisarmos os casos em dois projetos distintos e o processo geral de *sound design*, foi possível compreender melhor a diferença na produção e o papel diferente que cada elemento tanto de *sound design*, quanto de composição musical podem assumir. Como o título sugere, foi de fato realizado um relato sobre diálogos entre o design gráfico e o *sound design* em jogos digitais.

No projeto *Stoker*, foi possível observar que mesmo sendo um projeto de proporções menores, ele apresentou uma dificuldade razoável por ter seu universo acústico em uma zona fora do meu conforto. Além disso, ficou evidente como pode ocorrer a relação entre o design gráfico e o *sound design*. Neste caso, a arte pixel serviu como guia para a proposta sonora do jogo, a qual em muitos casos, apesar da limitação do *chip* de som do *GameBoy*, teve que ir além da limitação e considerar o que era melhor para o áudio do jogo, dado o contexto atual no qual o jogo foi produzido. O áudio nem sempre deve se sacrificar em função da arte do jogo, da mesma forma que o oposto também válido. Sempre que possível, eles tentarão atuar como uma unidade, mas no final, deve sempre prevalecer aquilo que for melhor para o design do jogo como um todo. Este mesmo raciocínio pode-se aplicar também para outras mídias, tendo em vista o estudo de referências feitos em ambos os projetos.

O segundo caso – *Nova Empire* -, de cunho comercial e proporcionalmente, muito maior que o primeiro caso, teve um foco maior em manter uma qualidade comercial, fazendo muitas vezes o uso do *sound design* junto a estudos e percepções da psique humana, de tal forma incentivar o consumo dentro do jogo e colocá-lo em uma posição confortável no mercado. Além disso, todos os elementos do design, tanto o gráfico, quanto o de áudio e o de game design deveriam funcionar de forma harmônica e completamente natural do que se esperaria de um jogo líder de mercado. De certa forma, é um *sound design* menos inovador e menos criativo, mas, por outro lado, se torna extremamente técnico e funcional, onde a maioria das escolhas são centradas de tal forma a potencializar a experiência do jogador, prendê-lo no jogo em um curto período de exposição e deixá-lo completamente confortável no ambiente do jogo.

Ao que diz respeito aos objetivos apresentados na introdução do relatório, é possível confirmar o sucesso do trabalho, o qual se propunha a criar uma literatura de caráter acadêmico abordando o *sound design* como elemento principal.

Como conclusão deste trabalho de graduação, gostaria de dar destaque à importância do conhecimento do universo acústico dentro do mundo do design. O *sound design* é uma ferramenta que não deve ser pensada à parte, mas sim como parte do projeto, pois ele é capaz de solucionar problemas tanto em conjunto com o design gráfico (e outros tipos de design) quanto por conta própria. Muitas pessoas dizem reconhecer isto, mas se esquecem do áudio como alternativa viável. Espero que por meio deste projeto, àqueles que o lerem, tenham compreendido melhor a sua importância e de que forma ele pode melhorar consideravelmente a execução de um projeto.

## Referências Bibliográficas

- CROSS, Mark. **Audio Post Production for Film and Television**. Boston: Berklee Press, 2013.
- DONELLY, K.J et al. **Music in Video Games**. New York: Routledge, 2014.
- FARNELL, Andy. **Designing Sound**. Cambridge: The MIT Press, 2010.
- KOMPANEK, Sonny. **From Score to Screen** Sequencers, Scores & Second Thoughts Th New Film Scoring Process. New York: Schirmer Trade Books, 2004.
- MARKS, Aaron. **The Complete Guide to Game Audio** For Composers, Musicians, Sound Designers, and Game Developers. Lawrence: CMP Books, 2001.
- PHILLIPS, Winifred. **A Composer's Guide to Game Music**. Cambridge: The MIT Press, 2014.
- SOMBERG, Guy. **Game Audio Programming** Principles and Practices. Boca Raton: CRC Press, 2017.
- SONNENSCHNEIN, David. **Sound Design: The Expressive Power of Music, Voice, and Sound Effects in Cinema**. Saline: MWP, 2001.
- STEVENS, Richard; RAYBOULD, Dave. **Game Audio Implementation: A Practical Guide Using the Unreal Engine**. Boca Raton: CRC Press, 2016.