

Como entender e fazer

GAME DESIGN





Como entender e fazer Game Design

Thais Weiller

Capa usa uma screenshot do jogo Catpromize que eu fiz uns anos atrás.



Game Start

Primeira edição, 2023

Ficha catalográfica e Licença

W422 - Weiller, Thais

Como entender e fazer Game Design[livro eletrônico] / Thais Arrias Weiller. - Curitiba: 2022.

ISBN: 978-65-00-00004-7

1. Videogames – design 2. Jogos por computador – design

I. Título. II. Autor.

14-0215 CDD 794.81 CDU 79

Este trabalho é registrado sob a licença de [Atribuição-Compartilhelgual 4.0](#).

Você tem o direito de:

- Compartilhar — copiar e redistribuir o material em qualquer suporte ou formato

- Adaptar — remixar, transformar, e criar a partir do material para qualquer fim, mesmo que comercial.

O licenciante não pode revogar estes direitos desde que você respeite os termos da licença.

De acordo com os termos seguintes:

- Atribuição — Você deve dar o crédito apropriado, prover um link para a licença e indicar se mudanças foram feitas. Você deve fazê-lo em qualquer circunstância razoável, mas de nenhuma maneira que sugira que o licenciante apoia você ou o seu uso.
- Compartilhalgual — Se você remixar, transformar, ou criar a partir do material, tem de distribuir as suas contribuições sob a mesma licença que o original.

Sem restrições adicionais — Você não pode aplicar termos jurídicos ou medidas de caráter tecnológico que restrinjam legalmente outros de fazerem algo que a licença permita.

| | |
|---|-----------|
| Introdução | 7 |
| O que é um jogo? | 13 |
| Primeiros Estudos _____ | 15 |
| Game Studies _____ | 22 |
| Por que jogamos | 26 |
| Diversão _____ | 26 |
| Motivação _____ | 34 |
| Conciliação _____ | 38 |
| Game Design | 42 |
| As partes do Design _____ | 42 |
| Mecânicas | 48 |
| Core Loop _____ | 48 |
| Loops Auxiliares _____ | 52 |
| Mecânicas Concorrentes _____ | 54 |
| Loops de Feedback _____ | 55 |
| Riscos e Informações Assimétricos _____ | 57 |
| Gameplay Emergente _____ | 59 |
| Resumindo _____ | 60 |
| Progressão | 61 |
| Técnicas de construção modular _____ | 62 |
| Aprendizado _____ | 65 |
| Ritmo _____ | 66 |
| Linearidade e sensação de interação _____ | 68 |
| Resumindo _____ | 68 |
| Contexto | 69 |
| Leitmotiv _____ | 70 |
| Narrativa _____ | 73 |
| Estética _____ | 75 |

| | |
|--|-----------|
| Resumindo _____ | 76 |
| Sistemas | 77 |
| Sistemas determinísticos e não-determinísticos _____ | 78 |
| Grandezas _____ | 80 |
| Tabelas _____ | 82 |
| Sistemas em movimento _____ | 83 |
| Balanceamento _____ | 83 |
| Resumindo _____ | 84 |
| Documentação de Game Design | 85 |
| GDD _____ | 85 |
| Quando não documentar _____ | 86 |
| Tipos de GDD _____ | 88 |
| Linguagem técnica _____ | 92 |
| Palavras finais | 96 |
| Referências | 98 |

INTRODUÇÃO

Quando falamos em *game design* com pessoas que não são ou não tem intimidade com a área de jogos é comum alguma confusão sobre o quê, exatamente, estamos nos referindo a. Alguns vão pensar que estamos nos referindo a fazer o computador ou console desempenhar as ações necessárias para o jogo acontecer (o que é programação), outros vão associar a palavra *design* com estética e assumir que deve ter alguma relação a aparência do jogo (o que é arte).

Ambas estas áreas são partes essenciais da existência de um jogo, mas não são o tal do *game design*.

Game design é a disciplina que se dedica a entender e planejar o funcionamento de jogos. É semelhante ao trabalho de um arquiteto: o arquiteto faz a planta baixa levando em consideração o tamanho do terreno, a forma que será usada, a necessidade dos usuários. Resumidamente, o arquiteto é responsável por entender o funcionamento e uso do que vai construir e por planejar essa coisa considerando esse funcionamento e uso desejado. Ele não assenta a laje ou compra os móveis, esse é o trabalho de outros profissionais, e o mesmo é válido no nosso caso. O funcionamento de um jogo é essencial para a experiência de todos que vão o jogar de forma que o game designer existe como profissão tanto em jogos digitais como em analógicos.

É por isso, também, que eu gosto de chamar essa área de *projeto de jogos* e costumo traduzir game design para o português como **projeto de jogos**.

Mas antes de fazermos jogos, digitais ou analógicos, precisamos entender o que, na verdade, são jogos e o que já foi dito sobre eles. No primeiro capítulo, faremos essa revisão histórica e vamos também definir o que vai ser entendido por jogo durante todo esse livro. Parece uma questão redundante definir o que é jogo num livro de game design, mas na verdade é essencial para o design saber se a experiência que estamos fazendo vai ter objetivos claros ou não. Essa necessidade não fez sentido? Vai fazer depois que você ler o primeiro capítulo.

Depois, vamos falar um pouquinho sobre o porquê, como humanos, jogamos. Você com certeza já se perguntou isso e é uma pergunta fascinante e absolutamente sem uma resposta. Já estou imaginando a sua cara ao se perguntar “Se não tem resposta, por que vamos falar sobre isso?!?” e a resposta é muito simples: não é que não há resposta, na verdade existem várias! Não há uma única resposta definitiva que defina por que todas as pessoas do mundo jogam. É por isso que é ainda mais importante conversar, como conversaremos no segundo capítulo, sobre alguns dos principais motivos que levam os humanos a jogar. Isso vai te ajudar a entender um pouco melhor seus jogadores e o que eles querem com o seu jogo. Com sorte, vai te ajudar a entender um pouco melhor até você mesmo!

Se você já leu meu livro *Game Start* e lembra bem do que foi discutido lá, o assunto desses dois últimos capítulos deve parecer meio repetitivo. Você tem total razão, é repetido mesmo. Eu usei ele como base para desenvolver os dois primeiros aqui e embora ele traga as mesmas informações está escrito de uma forma mais amigável. Entretanto, apesar da melhor formatação, o conteúdo continua sendo o mesmo e, se você tem boa memória, você pode pular o resto deste capítulo e o próximo e começar direto no terceiro capítulo. No terceiro capítulo é quando realmente esse livro começa e vamos falar sobre desenvolvimento e jogos e qual o papel do game design nele. Vamos falar um pouquinho de como a área é dividida na indústria e como essa divisão inspirou o resto do livro.

Vamos abordar a parte central do game design, as mecânicas, no quarto capítulo e explorar alguns dos seus componentes e conceitos principais. As mecânicas são os principais fluxos de funcionamento de um jogo e, sem elas, o design e o jogo como um todo não existem.

Em seguida, vamos falar de como planejamos a jornada do jogador por meio da progressão no quinto capítulo. Conforme o jogo progride, se novos elementos ou mudanças não são introduzidos, podemos correr o risco de entediar o jogador. E se o jogador fica entediado, ele não vai ter a experiência completa que você gostaria que ele tivesse. A forma mais popular de progressão é o level design e também vamos falar sobre ele aqui.

Essas informações em geral são apresentadas com uma roupagem que lhe dá sentido, para a qual damos o nome de contexto e é o tema no sexto capítulo. O contexto explica para o jogador, por meio de palavras, sons, cores, todos ou nenhum dos anteriores, o por que ele está jogando e por que é importante no mundo do jogo. Isso pode ser desde o jogador controla um quadrado e por isso talvez seria legal se encaixar em outros quadrados até as motivações emocionais de por que o protagonista precisa ajudar a sua filha adotiva a chegar em um determinado lugar.

Todos eles elementos precisam de informações que quantifiquem as conexões entre eles. Essa coleção de informações é chamada de sistemas, que é o assunto do sétimo capítulo. Os sistemas são a parte mais lógica e direta de todo o trabalho do designer e é por meio dele que atingimos o sonho de ter um jogo bem balanceado, ou seja, um jogo em que todas as ações que o jogador pode tomar tem consequências proporcionais e que correspondem as demais dentro do universo do jogo.

Finalmente, depois de ter todas essas informações sobre os principais elementos do game design, você finalmente pode começar a pensar em formatar suas idéias em um formato que permite que você ou você e uma equipe possam começar a fazer um jogo. Documentamos essas informações em um Game Design Document (GDD) e ele pode ter vários formatos diferentes. No oitavo capítulo vamos sobre alguns desses formatos, as vantagens e desvantagens de cada um e se, sequer, você precisa mesmo de um.

Você deve ter reparado que eu não gosto muito de apresentar respostas prontas. Na maioria dos capítulos, eu vou te apresentar o problema ou a questão central, apresentar as principais idéias, quais as forças e as fraquezas de cada uma e deixar na sua mão a conclusão sobre. Eu faço isso por alguns motivos, o primeiro é que eu não acho que game design é uma ciência exata: existem boas práticas e coisas que nós sabemos que funcionam, mas a verdade é que nada na área é uma verdade absoluta e tudo é uma questão de contexto. Por exemplo, uma boa prática é sempre dar feedback claro sobre as ações do jogador e a reação delas do mundo, para que o jogador saiba claramente o que ele está fazendo de certo ou errado e possa mudar seu comportamento de acordo. Entretanto, existem jogos ou momentos dentro de jogos que você não quer deixar isso tão claro por motivos que fazem sentido para o jogo. Por exemplo, o protagonista está em uma situação nova e estressante e você quer comunicar isso para o jogador de uma forma que ele também se sinta assim, e para aumentar o estresse você não deixa claro quais as consequências das ações do jogador. É uma solução elegante de fazer o jogador se sentir como o protagonista sem falar para ele como se sentir e você atingiu isso indo contra uma boa prática.

Segundo, você tem que aprender a identificar o que ressoa com você e o que não pois vai ser uma habilidade essencial para descobrir o que está funcionando nos seus jogos e o que não está. Jogos se diferenciam das outras mídias por serem interativos, isso significa que eles só existem se estão sendo jogados por alguém. Se seu jogo não está sendo jogado, isso é um problema do jogo. Se o jogador não entendeu o não

consegue fazer algo, isso é um problema de design. **O jogador nunca está errado**¹, e como designer você tem que conseguir observar o jogador e entender onde está o erro para arrumar depois.

O terceiro e último motivo é que eu confio em você como designer para decidir o que faz mais sentido para você. Cada pessoa tem uma filosofia de design e é escolhendo o que você acredita que faz sentido que você vai achar a sua.

E tudo bem se você acha que não merece toda essa confiança ainda, é como você devia se sentir principalmente se você ainda não terminou um jogo, mas saiba que mesmo sem experiência nenhuma *terminando* um jogo, você já tem experiência o suficiente na sua vida jogando jogos e *fazendo, projetando e completando* inúmeras atividades semelhantes a jogos. Esse conhecimento não se traduz automaticamente a bons designs, você precisa pensar criticamente sobre essas experiências e ser capaz de ser crítico de seu próprio trabalho. Mas se você não se esquecer que autocrítica é a ferramenta mais importante no repertório de toda pessoa, em breve vai estar fazendo jogos incríveis!

¹ Exceto em raras, *muito raras*, ocasiões que nem vale a pena lembrar. Sempre assumo que o erro é seu, sério mesmo, sem meme. Se algum de vocês falar alguma coisa como "eu tinha feito um level genial mas os jogadores não entenderam e teve que ser cortado" que nem um ""chefe"" meu uma vez falou, considerem suas carteirinhas de game designers revogadas.

○ QUE É UM JOGO?

Se dê um tempo e tente definir o que é um jogo.

É sério, pare de ler e tente definir em palavras o que define um jogo. Quais características são essenciais para que uma atividade seja ou não um jogo.

Já pensou? Pare de novo e pense mais um pouco.

Acredito que você inicialmente lembrou dos bons momentos passados jogando e passou por conceitos de diversão, entretenimento, engajamento e desafio. Mas, daí, pensou em alguns videogames que te levaram por uma jornada linear e dramática e pensou em história, personagens e mensagem. Por fim, você lembrou que nem todos esses jogos foram exatamente divertidos e ainda sim você continuou os jogando, então talvez a definição esteja menos na diversão em si e mais para um conceito mais amplo de... experiência?

Se essa foi sua jornada ou não, se esses foram os conceitos que você pensou ou se os seus foram diferentes, não importa no momento. Jogos são todas essas coisas e muitas outras também, o jogo é algo que a gente olha e instintivamente sabe o que é mesmo sem saber exatamente o porquê é. Ou melhor, não sabemos *por enquanto*, pois hoje você vai aprender a definir um jogo. Mas antes de tentarmos delimitar com mais certeza o que é ou não um jogo, vamos separar alguns

conceitos centrais: de um lado, jogos e brincadeiras, e do outro o espírito lúdico.

Se você já teve algum contato, por menor que seja, com crianças ou animais, você deve ter percebido o quanto gostam de se manter entretidos. Contanto que se sintam confortáveis, eles brincam e jogam quase todo o tempo livre. Esse ímpeto criativo e de recreação é chamado de **espírito lúdico** e ele pode ser observado em quase todos os animais complexos (aves, mamíferos, alguns répteis e até animais não vertebrados como o polvo), principalmente em seus jovens, mas também em adultos.

Jogos e brincadeiras são a concretização deste mesmo espírito lúdico em uma atividade material, digital ou mental. Quando esta materialização do espírito lúdico não parece ter nenhuma função direta de sobrevivência ou prática e não é regida por regras claras a todos os participantes ou objetivos definidos, damos a ela o nome de **brincadeira**. Regras e objetivos são conceitos abstratos que requerem um certo nível de refinamento intelectual para serem definidos e aceitos entre mais de um indivíduo, logo enquanto brincadeiras são observadas em inúmeras espécies, jogos são um pouco mais raros, sendo comuns em criaturas dotadas de pensamento abstrato como nós humanos. Isso não significa que pessoas adultas são inteligentes demais para brincadeiras! Adultos são perfeitamente capazes de participar e se divertir em brincadeiras quando estão com vontade das mesmas, apenas que elas são mais democráticas entre espécies e indivíduos.

Finalmente, agora precisamos definir o que é um **jogo**. Mas antes disto, precisamos saber qual a história do entendimento de jogos até chegarmos na definição mais contemporânea e a que vamos usar como referência para o resto deste livro.

Primeiros Estudos



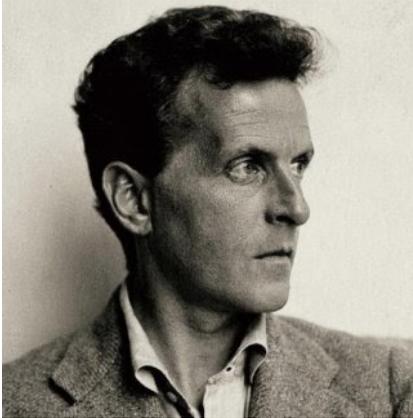
Johan Huizinga

O primeiro pesquisador a se aventurar nesse empreendimento que temos conhecimento foi **Johan Huizinga** (2000) em seu livro *Homo Ludens*, primeiramente publicado em 1938. Na obra, Huizinga, que era um historiador de profissão, faz uma retomada histórica de jogos e brincadeiras em várias sociedades e os usa de evidência para suas próprias teorias. Para ele, a atividade lúdica é necessidade humana e de outras espécies, inata a maioria dos animais, sendo portanto, também o fator primordial na criação da cultura humana. Huizinga também defende que, da mesma forma como os animais brincam instintivamente, desde quando são filhotes até a maturidade, o jogo e a brincadeira fazem parte do espírito e da natureza humana e são determinantes em qualquer tipo de criação feita pelo homem. Dessa maneira, para Huizinga, praticamente toda atividade cultural e artística humana, desde disputas de injúrias à alta literatura e música erudita, é residual ao jogo. O que Huizinga chamava de

atividade lúdica nós estamos chamando de espírito lúdico neste curso, embora não façamos as mesmas afirmações ambiciosas.

Podemos então observar que foi Huizinga quem cunhou pela primeira vez a divisão entre brincadeira e jogo e o estabelece o conceito de espírito lúdico. Outro ponto essencial da teoria do autor é que o jogo é “diametralmente oposto à seriedade”, “‘não-sério’ e exterior à vida habitual” e que é “impossível aplicar [a estas atividades] as noções de vício e virtude”. Para Huizinga, o jogo ocorre em uma realidade separada da nossa no que ele chama de Círculo Mágico, um conceito semelhante à suspensão de descrença da literatura. Enquanto jogando e, por consequência, dentro do círculo mágico, os jogadores agem regidos pela realidade do jogo, mas assim que o jogo acaba as ações do mesmo ficam contidas nele mesmo. Essa necessidade de o jogo ser livre e sem consequências para a vida real contradiz a nossa realidade contemporânea, com jogadores profissionais e mega eventos esportivos e e-sportivos, algo que só estava começando na época da publicação do livro e que Huizinga era absolutamente contra.

Alguns anos após o *Homo Ludens* de Huizinga, o filósofo **Ludwig Wittgenstein** desenvolve o conceito do jogo da linguagem em seu livro *Philosophische Untersuchungen* (Investigações Filosóficas). Grande parte dos conceitos deste livro já havia sido escrita e submetida para publicação em 1946, mas Wittgenstein retira o manuscrito e só o submete novamente anos depois, sendo publicado em 1953. Para o filósofo, a linguagem obedece a esquemas semelhantes aos de um jogo em determinadas formações, afirmando que grande



Ludwig Wittgenstein

parte da utilização de uma língua se dá na forma de uma atividade, de uma parte da vida que sofre grande influência lúdica. Embora em momento algum Wittgenstein arrisque definir o que entende como jogo, o conceito utilizado em seu estudo se assemelha muito ao de Huizinga, de um fluxo lúdico superior, aplicado pelo indivíduo na execução de

alguma atividade, qualquer que ela seja.

Roger Caillois (1986) é o próximo intelectual, cronologicamente, a fazer contribuições relevantes ao entendimento dos jogos. Assim como Huizinga e Wittgenstein, Caillois não dedicou sua vida acadêmica aos jogos e publicou apenas um livro sobre o assunto, o *Les jeux et les hommes* de 1957. Apesar de não corroborar com o entendimento de Huizinga de que o jogo é a base central da cultura humana (“o espírito do jogo é essencial para a cultura, mas, no percurso da história, jogos e brincadeiras são resíduos dela”), Caillois busca no autor a base de sua concepção de jogos, definindo-os como atividades livres, separadas da realidade, de desenrolar incerto, necessariamente improdutivas, regulamentadas e fictícias.

Dessa forma, Caillois cria limites mais estreitos para as atividades que podem ser entendidas como jogos, diferente de Huizinga que acreditava que quase todas atividades humanas



Roger Caillois

poderiam ser entendidas como jogos . Caillois entende que todos os jogos podem ser divididos nas categorias *Agon* (competições) como por exemplo, basquete, vôlei, lutas, entre outros, *Alea* (jogos de azar) podemos citar baralho, roleta bingo, par ou ímpar, *Mimicry* (imitações) como exemplo temos mímica, representações teatrais, fantasias e *Ilinx* (brincadeiras de disrupção, que visam criar vertigem e desorientação) atividades como andar de skate, asa delta, acrobacias, escaladas, entre outros. Além desta divisão em categorias, há também diferentes espectros que os jogos podem ter, indo de *Paidia* (atividades menos estruturadas e regradas, mais livres e espontâneas) a *Ludus* (jogos estruturados e com regras bem definidas), reforça novamente a divisão entre brincadeiras e jogos.

Após Caillois, a ciência do jogo passa a não mais ser composta basicamente por um monólogo (com um pensador influente por época) e torna-se processo de diálogo entre vários pesquisadores, em diversas áreas, na construção de um entendimento mais amplo. Durante esta época (aproximadamente entre as décadas de 1960 e 1990), os conceitos de jogos foram aplicados a outras áreas ou hibridados com outras teorias, ou ainda, apenas observados com enfoques diferentes. Uma das grandes diferenças permitidas por esse diálogo em relação à fase anterior é sua interdisciplinaridade; alguns desses teóricos são nativamente da biologia ou da matemática e, até mesmo, da dramaturgia, e com isso trazendo as visões únicas de seus áreas de origem e contribuindo com a diversidade de pontos de vista.

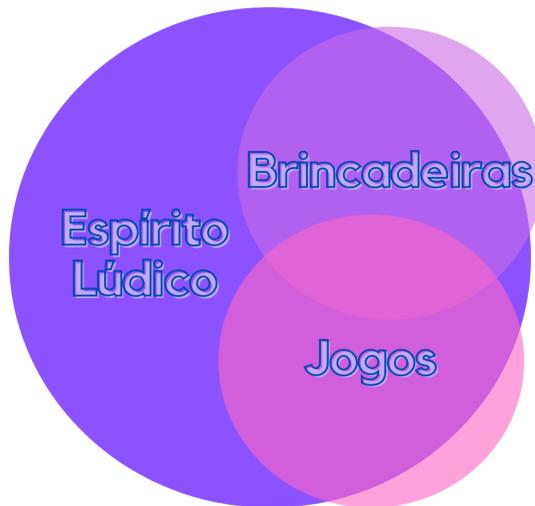
Mesmo com a inclusão de novos conceitos, a definição inicial de jogo, como uma atividade não séria e não relacionada com a realidade, imaginado pelos gregos e firmado por Huizinga e Caillois, continuava dominante. Bernard Suits, um dos autores desse período a estudar jogos, exemplifica perfeitamente este sentimento ao definir jogos como “a tentativa voluntária de superar um obstáculo desnecessário”.

Outro marco dessa geração foi a aparição dos videogames no contexto de pesquisa. **Chris Crawford** foi um dos primeiros a chamar atenção para a nova mídia e seu potencial em uma época em que a indústria, o mercado e as criações ainda eram tímidos. Crawford aponta como jogos (tanto digitais quanto analógicos) não eram considerados meios de expressão artística até aquele momento, mas que tinham potencial para tal. O autor não se aprofunda muito no conceito de jogos em si,



Chris Crawford

digitais ou não, mas pode ser considerado uma espécie de visionário, ao observar essa possibilidade em 1982, época em que videogames pareciam não passar de poucos pixels espalhados na tela com alguma movimentação, e que atraíam os mais jovens.



Imagine o espírito lúdico como um sentimento que incentiva o indivíduo a jogar ou brincar. A maioria dos jogos e brincadeiras ocorrem dentro dos limites do espírito lúdico, mas também podem existir sem essa relação, como quando você se sente obrigado ou pressionado a participar de um jogo. É como seu espírito criativo e um desenho: seu espírito criativo (ou inspiração) pode te levar a uma ideia ou conceito, mas o desenho só se concretiza quando você, de fato, o faz. Entretanto, um desenho também pode ser feito completamente sem inspiração da criatividade, como no caso de um desenho técnico ou algo do tipo. Da mesma forma, se pode jogar um jogo sem se estar exatamente lúdico e é possível sentir a inspiração ou o espírito lúdico e não os concretizar em nenhuma atividade. Existe uma relação entre estes dois conceitos e um é a origem da criação do outro, entretanto eles podem sim existir de forma independente.

Game Studies

A partir da década de 1990 é possível observar uma intensificação nos estudos acadêmicos de jogos, em particular com a inclusão de jogos digitais como um tópico prolífico de pesquisa. Este novo campo de pesquisa começou a ser chamado de Game Studies (não confundir com Game Theory que é uma linha de pesquisa da matemática). Como consequência do maior número de vozes simultâneas na área, no começo dos anos 2000 acontece uma grande cisão entre dois principais grupos: os ludologistas e os narratologistas. Os narratologistas partiram dos conhecimentos e teorias dos estudos literários para entender jogos, buscando entendê-los por meio da necessidade humana de compreender o mundo



Jesper Juul

por meio de narrativas. Os ludologistas viram nos jogos uma mídia completamente diferente de tudo que veio antes e impossível de ser entendida por meio das abordagens tradicionais tendo em vista sua natureza interativa e potencialmente não-linear.

Essa cisão foi reconciliada até o fim da década de 2000 como duas possíveis abordagens ao estudo de jogos que não são necessariamente antagônicos e que podem coexistir. Um destes ludologistas, **Jesper Juul** (2003), propôs uma definição de jogos que será a que usaremos nesta disciplina. Juul

comparou as mais proeminentes teorias de jogos antes dele, de Huizinga a pesquisadores contemporâneos a ele, e apontou que, para ser um jogo, uma atividade lúdica deve ter:

1. **Regras** – jogos devem ter regras que criem um sistema jogável.
2. **Variabilidade dos resultados** – as diferentes ações do jogador resultam de forma diferente.
3. **Valor dos resultados** – alguns desses resultados são mais desejáveis e outros, menos.
4. **Esforço do jogador** – os diferentes inputs do jogador resultam em diferentes resultados.
5. **Valorização dos resultados para o jogador** – o jogador pode ganhar ou perder e ter diferentes reações a isso.
6. **Consequências negociáveis** – o jogo pode ser atrelado (ou não) a consequências na vida real.

Vamos considerar essas características usando o exemplo do xadrez. Neste jogo, há regras que determinam o tipo e tamanho do tabuleiro, os tipos e números de peças e as movimentações possíveis para cada peça (1). Os resultados variáveis dizem respeito ao jogador escolher quais serão seus movimentos (2). Os movimentos resultam em situações que são eleitos pelas regras como melhores ou piores do que outros e essa mesma valorização (3). Por exemplo, o mate de um peão adversário é

um resultado positivo e que dará alguma vantagem ao jogador que o realizou, mas o mate de um bispo tende a ser considerado um melhor resultado que o do peão, pois dá uma vantagem maior ao privar o adversário de uma peça com mais poder, enquanto o mate do rei pode ser considerado melhor que ambos, já que dá a vitória ao jogador que o realiza. O esforço do jogador determina que esses resultados variáveis, que possuem diferentes valores, são consequência das escolhas do jogador que, no caso do xadrez, é majoritariamente intelectual (4). A valorização dos resultados pelo jogador e as consequências negociáveis dizem mais respeito a valores subjetivos do jogador e significam que o jogador pode vincular-se emocionalmente aos resultados do jogo (5) (como ficar feliz se ganhar e triste se perder), culminando com que os resultados do jogo podem desencadear consequências em sua vida a serem negociadas (6), como apostas ou prêmios pela vitória de torneios. É importante ressaltar que as duas últimas características são subjetivas sim mas também não são essenciais na definição de Juul. Elas existem apenas para apontar que são características que podem ser observadas ou não.

Algumas atividades que chamamos de jogos, entretanto, não possuem todas as características apontadas como necessárias para Juul, como é o caso da maioria dos jogos da série The Sims. Nesses jogos, apesar de haver uma enorme variedade de regras e resultados variáveis, não é definido pelas regras do jogo quais resultados são mais desejáveis que outros. O jogador pode, por si só, estabelecer quais resultados lhe são mais desejáveis (por exemplo, o resultado mais desejável pode ser montar uma família feliz ou uma família rica ou, ainda, atrair

todos os personagens para piscina e depois remover a escada², mas como esses resultados não são estabelecidos pelo jogo por si só e, portanto, não são amplamente aceitos pelos jogadores, não há em The Sims uma composição completa de um jogo. Isso não torna The Sims uma experiência menos rica e divertida ou ainda inferior para seus jogadores, apenas o classifica como uma composição diferente de sistema lúdico.

Vamos, a partir de agora, considerar atividades que possuem as quatro primeiras características da classificação de Juul como jogos e as que faltam algumas delas como brincadeiras. Essa diferenciação não deve ser entendida como algum tipo de hierarquia, como se jogos fossem melhores ou superiores de alguma forma. Jogos são atividades que seguem o modelo apontado e, por isso, são regidos pelos princípios que vamos estudar nesta matéria. Muitos destes princípios também podem ser aplicados à brincadeiras (e você é encorajado a testá-los), mas a própria natureza mais livre da brincadeira não torna esses princípios tão essenciais como em um jogo.

Parabéns, agora você sabe o que é um jogo! Saber o que constitui um jogo vai te ajudar enormemente em planejar o que é mais importante dentro do design de cada um dos seus jogos. Mas antes de desmontarmos o design de um jogo para entender como as peças funcionam, entender o porquê jogamos vai te ajudar a melhor elencar estas partes. É isso o que faremos no próximo capítulo!

² Essas novas regras e objetivos não definidos pelo jogo mas criados pelo jogador também são chamados de *gameplay emergente*. Vamos falar disso mais para frente.

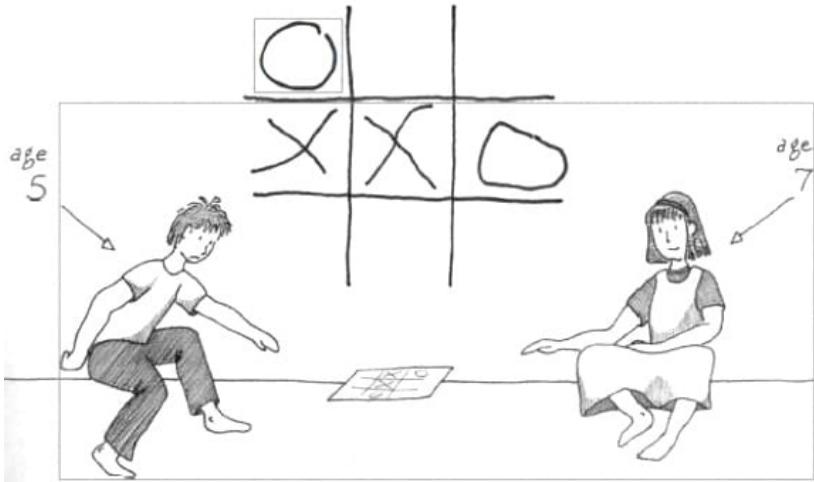
POR QUE JOGAMOS

No capítulo passado, falamos de como jogos são parte da nossa cultura e história humanos e sobre como começamos a tentar entender esses jogos. Nessa aula, vamos focar no porquê tomamos parte e criamos estas atividades que aprendemos a definir como jogos.

Passamos por várias definições de jogo, algumas até conflitantes entre si. Apesar desta variedade, um ponto era repetido por praticamente todos os teóricos: os humanos jogam principalmente para atingir um estado de espírito conhecido vulgarmente como **diversão**.

Diversão

O primeiro autor que vamos abordar é Koster (2005) pois ele trás a diversão diretamente em seu estudo sobre jogos, vinculando a existência da diversão ao desafio percebido pelo jogador. Para ele, o desafio é percebido pelos padrões inicialmente desconhecidos e que são decifráveis pelos indivíduos durante a atividade. Koster entende humanos como “máquinas” orgânicas decodificadoras de padrões que obtêm diversão pelo ato de decifrar mistérios e atividades muito difíceis ou muito fáceis serão consideradas tediosas, enquanto as que estão em uma dificuldade ótima poderão ser divertidas e estimulantes durante a experiência do jogador.



Página de *Theory of Fun* de Koster.

Assim, uma atividade pode começar como desafiadora e divertida, mas caso o nível de desafio aumente ou diminua (seja pela progressão do jogo ou pelo avanço de habilidades e conhecimentos do próprio jogador) para além do espectro considerado divertido para o sujeito, tal atividade não seria mais divertida. Neste ponto do argumento, Koster vincula atividades divertidas à idade do sujeito e usa como exemplo o jogo da velha, que segundo ele é divertido até a idade quando o jogador decodifica o procedimento padrão de vitória. Da mesma forma, dependendo das habilidades iniciais do sujeito, cada atividade tem seu ponto máximo de diversão em diferentes idades do indivíduo.

A teoria de Koster de que diversão se baseia apenas no desafio mental se enquadra muito bem para alguns jogos, como os mentais (puzzles) e de estratégia (RTS e TBS), mas não parece explicar completamente a diversão provinda, por exemplo, de jogos em que o reflexo é o principal requisito (jogos rápidos como FPS e alguns action platforms e beat'em ups) ou da repetição de jogos cujo padrão ou desafio o jogador já domina completamente e aparenta ainda se divertir ao jogar. Por exemplo, você já jogou seu jogo favorito inúmeras vezes, mas ainda torna a jogá-lo por vez ou outra. Esta teoria também não parece explicar completamente a diversão vinda de esportes ou videogames controlados por movimento (como jogos de dança, como Pump It Up e Dance Dance Revolution e o Wii, Kinect e PlayStation Move).

Koster tem algumas respostas, mas não todas, então temos que ir mais fundo. Uma das principais fontes de Koster é um teórico chamado Csíkszentmihályi, um psicólogo que buscava entender a origem da alegria e criatividade. Em seu estudo, o autor entrevistou indivíduos com os mais diferentes origens e ocupações na busca pelo motivo pelo qual algumas pessoas simples são muito boas no que fazem e se sentem bem enquanto o fazem. Surpreendentemente para ele, mesmo entrevistando pessoas completamente diferentes com funções de ordem transversalmente opostas (desde mães solteiras e montadores de linha de produção a cirurgiões e atletas olímpicos), ele encontrou descrições similares das sensações dos sujeitos desempenhando atividades tão radicalmente diferentes. O que era feito mudava completamente, mas a descrição de como os indivíduos sentiam-se realizando essas atividades era bem parecida.

Baseado nas entrevistas desta pesquisa, Csíkszentmihályi chegou a conclusão que os momentos mais prazerosos da vida da maioria das pessoas foram aqueles que, em vez de “não fazerem nada” ou apenas relaxarem, os indivíduos estavam envolvidos em atividades complexas e extremamente exigentes, mas apesar da grande demanda eles sentiam-se completamente capazes de realizá-las. Ou seja, as pessoas se sentem mais felizes quando estão fazendo algo muito difícil enquanto se sentem totalmente no controle desta atividade, como por exemplo quando você está enfrentando um chefe do seu jogo favorito e está quase vencendo depois de algumas tentativas fracassadas.

Podemos perceber aqui a inspiração de Koster de que o desafio é o que mantém o jogador entretido, mas Csíkszentmihályi não associa esse desafio à solução de problemas apenas, como Koster fez, e sim a percepção subjetiva do jogador em relação à dificuldade. Ou seja, não é o desafio objetivamente que torna a situação desafiante, mas sua própria percepção do desafio. Esta “sensação”, para Csíkszentmihályi, se diferencia da simples diversão por permitir realização pessoal e o orgulho de ter alcançado ou ter participado da ação.

Csíkszentmihályi aponta determinadas características desse estado, que ele chama de flow, sendo elas:

- **Percepção de dificuldade/desafio apropriados**, tendo em vista que “em todas as atividades que pessoas de nosso estudo reportaram tomar parte, satisfação vinha em um

ponto específico: sempre que as oportunidades de ação percebidas eram iguais à capacidade do sujeito”.

- Todos os **sentidos do indivíduo estão concentrados** na atividade em si e nos estímulos proporcionados para o máximo desempenho.
- **Metas** que guiam as ações presentes e futuras em direção a um objetivo razoavelmente claro.
- **Feedback imediato** (ou percepção dele) de como as ações do indivíduo estão resultando na atividade e como essas ações o posicionam em relação às metas.
- Ao se perceber todos os fatores antes citados, o autor indica uma característica resultante que é a **sensação de controle sobre a atividade realizada**, mas é importante ressaltar que o relevante aqui “não é a sensação de se estar no controle, mas sim a de exercitar o controle em situações difíceis”.

Podemos ver como a teoria do flow é interessante para jogos tendo em vista que descreve um estado mental (total foco provindo da dificuldade percebida) que todos nós conseguimos lembrar de ter experienciado enquanto jogávamos ao passo que aponta alguns elementos essenciais para o design de qualquer jogo (metas, dificuldade equivalente às habilidades, feedback às ações desempenhadas no jogo). Mas não podemos deixar essa relação clara nos fazer esquecer de um dos pontos principais da teoria de Csíkszentmihályi, de que

tanto fatores internos e externos ao indivíduo influenciam o flow, ou seja, a teoria não se aplica apenas ao jogo, mas ao jogador também. Inclusive, segundo o autor, embora os fatores externos tornem a obtenção do estado de flow mais ou menos favoráveis, o indivíduo é a condição essencial e insubstituível para a sua teoria. Mesmo que um jogo tenha sua dificuldade perfeitamente balanceada, deixe claro os objetivos do jogador e mostre claramente o resultado das ações do mesmo, é o jogador que irá perceber cada um destes elementos e utilizá-los na criação de sua experiência. Em resumo, o flow está na percepção subjetiva do indivíduo em relação à atividade e por mais que possamos usá-lo para entender nosso jogador enquanto ele joga, o flow não é uma ferramenta feita para analisar a atividade, no caso o jogo.

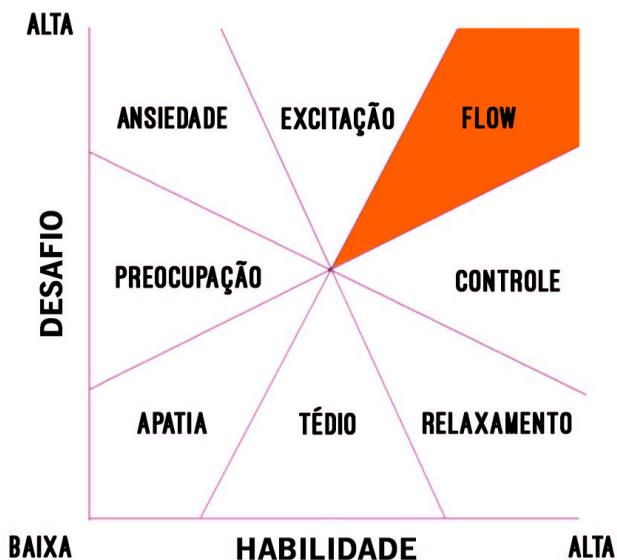


Gráfico do *flow* segundo Csikszentmihályi.

Como o flow depende do indivíduo, as habilidades mentais e físicas que esse já possui são um grande determinante em como o mesmo experiencia atividades. Por exemplo, uma pessoa com uma sensibilidade maior a determinados sons pode achar mais ou menos frustrante interagir com esses sons do que uma pessoa sem esta característica. Da mesma forma, Csíkszentmihályi acredita que algumas pessoas são mais propícias em encontrar o flow nas atividades que fazem no dia a dia devido a um conjunto de habilidades ou ações que estas desempenham com mais frequência. A essas pessoas ele dá o nome de autotéticas, afirmando que possuem esta característica por ter a tendência de seguir uma série de ações que as conduz de forma mais frequente ao flow.

Como essas características são parte do indivíduo, elas não parecem algo que você pode aplicar diretamente como um designer a um jogo. Ainda sim, I rei citar as ações que Csíkszentmihályi atribui a pessoas autotéticas não só porque acredito que vai ser útil a você, estudante, tentar incorporar estas ações no seu dia a dia, mas também por que elas podem ser levadas em consideração na hora de fazermos um jogo de forma indireta. Lembre-se sempre que não há uma aplicação 1:1 já que o foco da teoria de Csíkszentmihályi é o indivíduo e não a atividade, mas mesmo assim confio na sua habilidade de abstrair e reaplicar conhecimentos. Estas ações são:

- Apontar metas, pois “assim que as metas e desafios são definidos em um sistema de ação, elas então sugerem as habilidades necessárias para operar”.

- Tornar-se imerso na atividade, envolvendo-se completamente nas estratégias envolvidas em seu cumprimento.
- Prestar atenção ao que está acontecendo e ignorar as informações desnecessárias, de forma que a pessoa “não mais se sente mais como um indivíduo separado, de forma que seu ‘eu’ torna-se mais forte. [...] Devido a esta união da pessoa e do sistema, o indivíduo emerge em um nível superior de complexidade”.
- Aprender a apreciar a experiência imediata, mas não apenas na limitação do momento em questão e sim tendo em contexto, ao mesmo tempo, as “metas para os eventos cotidianos a fim de criar sentido”

É clara a influência da experiência definida por Csíkszentmihályi como flow com o centro da origem da diversão de Koster, mas há algumas diferenças essenciais. Koster situa o cerne de sua teoria na decodificação de padrões e, assim, vincula seu conceito de diversão à idade, afirmando que, com o passar do tempo, a maioria das atividades que consideramos divertidas passa a não mais o ser, enquanto Csíkszentmihályi não aponta esse padrão e, ainda, exemplifica casos de pessoas que sentiam o flow nas mesmas atividades por anos seguidos, por mais repetitivas que fossem. Koster também afirma que padrões que o indivíduo já tenha decodificados passam a não ser mais divertidos, enquanto Csíkszentmihályi não só não observa este comportamento como afirma que o flow existe em especial neste ponto, quando

o indivíduo busca o constante aprimoramento de suas habilidades. Mas talvez a principal diferença venha de que Koster identifica a diversão como uma função meramente cerebral, enquanto Csíkszentmihályi aponta o flow como um fenômeno intelectual com fortes ramificações físicas.

Tanto Koster quanto Csíkszentmihályi apontam que há claras diferenciações nos tipos de atividades consideradas divertidas por diferentes indivíduos. Os demais autores que vamos discutir partem desse ponto e constroem suas teorias na premissa de que não há um tipo único de diversão, muito menos um tipo de jogador. Essas teorias identificam perfis de jogador que devido às suas características intrínsecas preferem certos tipos de diversão a outras. A partir deles, também podemos observar uma preocupação maior em **motivações** do jogar.

Motivação

Bartle (1990) foi um dos primeiros a documentar esse tipo de teoria. Ele não era na época em que escreveu seu artigo um pesquisador, mas sim um jogador de RPGs jogados em salas de bate papo online chamados MUDs. Ao analisar seus colegas jogadores nas salas de MUD, Bartle foi capaz de apontar quatro tipos principais de jogadores e, conseqüentemente, quatro motivações principais:

- **Killers** – Jogadores que se divertem ou em “quebrar” o jogo, trazendo à tona problemas de programação ou de

game design, ou, ainda, prejudicar a experiência dos outros jogadores.

- **Achievers** – Jogadores que se focam em obter êxito quantificável dentro do mundo do jogo, seja por meio de recompensas, de quebra de recordes ou atingindo alto grau de expertise.
- **Socializers** – Jogadores que têm seu principal foco em se socializar com os demais jogadores, procurando conhecê-los dentro e fora do mundo do jogo.
- **Explorers** – Jogadores que buscam conhecer o máximo possível do jogo ou de alguma de suas peculiaridades (terreno, criaturas entre outros aspectos do jogo).

Bartle afirma que nem todos os jogadores apresentam um padrão de comportamento puro, mas que em geral um dos perfis se sobressai. Também aponta que, a fim de entreter um grupo com êxito, em geral, é necessária uma composição mista de tipos de jogadores, ou seja, um grupo vitorioso e completo é comumente aquele em que os jogadores buscam coisas diferentes dentro do jogo, mas, ainda assim, são capazes de conciliar seus objetivos pessoais com os gerais do grupo.

Hunicke et al. (2004) propõem uma abordagem mais direcionada para o desenvolvimento de jogos em si e não tanto ao indivíduo. Desta forma, encontraram oito diferentes tipos de diversão, os quais são bem mais práticos e aplicáveis diretamente ao design de jogos do que a observação de Koster

baseada na teoria de Gardner que tem foco psicológico e educacional. Tais elementos são:

- **Sensações** – o jogo apresenta sensações prazerosas.
- **Fantasia** – o jogo é um conto, uma história escapista da realidade.
- **Narrativa** – o jogo diverte por seu drama.
- **Desafio** – o jogo é um desafio a ser superado.
- **Socialização** – o jogo como uma estrutura social.
- **Descoberta** – o jogo como um território a ser explorado.
- **Expressão** – o jogo como terreno para a autodescoberta.
- **Submissão** – o jogo como um entretenimento desprovido de desafio mental.

Os autores acreditam que, embora a maioria dos jogos representem majoritariamente um ou dois tipos de diversão, muitos outros podem estar presentes de forma mais sutil. Por exemplo, eles atribuem à série de videogames de RPG Final Fantasy os elementos de Fantasia, Narrativa, Expressão, Descoberta, Desafio e Submissão, nesta ordem, enquanto, para série Quake, são atribuídos os elementos Desafio, Sensações, Competição e Fantasia, também nesta ordem.

Estes elementos devem ser pensados, segundo os autores, antes e durante a elaboração do jogo e, quanto mais tipos o jogo oferecer, desde que bem estruturados na ideia geral do jogo, mais chances ele terá de oferecer diversão a diferentes tipos de pessoas.

Se irmos ainda mais a fundo na compreensão da motivação de jogar, encontraremos a pesquisa de Nick Yee (2007), que aborda a questão de forma mais analítica ao examinar os resultados de seu questionário de 40 perguntas a mais de 3.000 jogadores. Yee foca em jogadores de videogames do tipo MMO (jogos que mantêm milhões de jogadores online simultaneamente em um mesmo “mundo”) a fim de compreender o que motiva esses jogadores a se manterem jogando por mais de 20 horas semanais, como uma pesquisa anterior de sua autoria havia indicado. Yee então separou as respostas e chegou a uma separação em três principais categorias, cada uma com subdivisões:

| Conquista | Social | Imersão |
|--|--|--|
| Avanço Progresso, poder, acúmulo, status | Socializar Conversas casuais, ajudar os outros, fazer amigos | Descoberta Exploração, tradições, descobrir coisas escondidas |
| Mecânica Números, otimização, análise | Relações Pessoal, autoconhecimento, encontrar e dar apoio | Role Playing História do mundo e do personagem, interpretação e fantasia |

| | | |
|--|---|---|
| Competição Provocar os outros, provação, dominação | Trabalho de equipe Colaboração, grupos, conquistas do grupo | Customização Aparência, acessórios, estilos, cores |
| | | Escapismo Relaxamento, escapar e/ou evitar problemas da vida real |

Fonte: Yee (2007).

Temos que levar em consideração que Yee encontrou estes resultados pesquisando um tipo específico de jogador, os de MMO, o que já torna o grupo de jogadores disponível mais direcionado à certos elementos, como os elementos sociais considerando que MMO são jogos multiplayer massivos. Ainda sim, é impressionante a quantidade de jogadores que participaram da obtenção destes resultados.

Conciliação

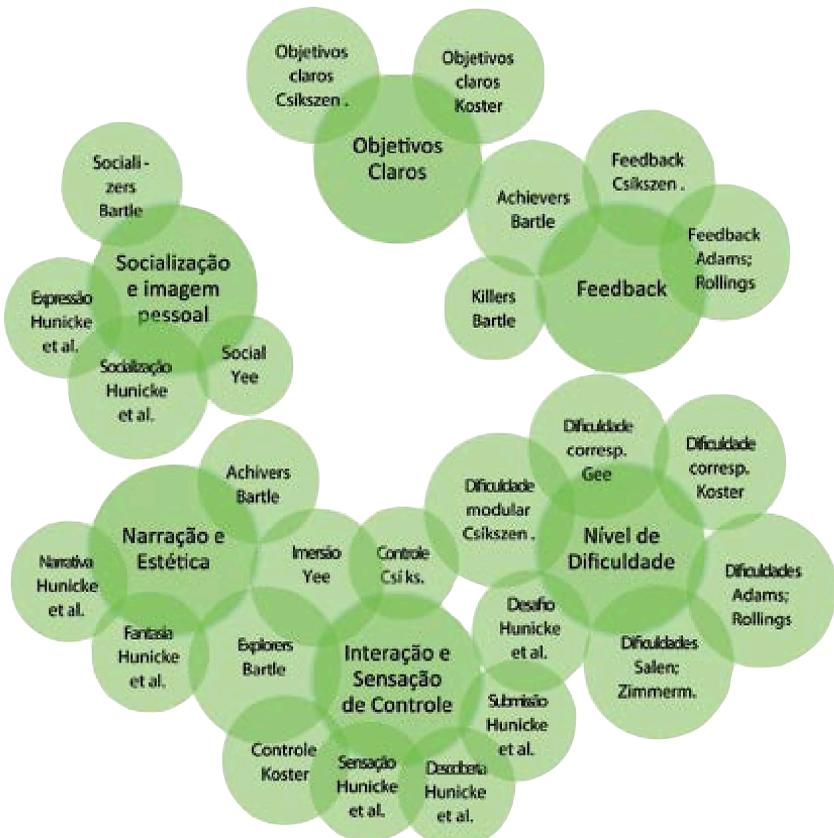
A divisão de Yee em muito lembra as teorias de seus antecessores, como Bartle e Hunicke, mas há peculiaridades nas divisões. Por exemplo, killers e achievers de Bartle caíam na motivação de conquista de Yee pois ambos buscam desempenho, mesmo que este desempenho seja contrário um ao outro (se as metas do jogo ou ir contra elas) e depende da orientação de cada indivíduo. Os socializers e explorers de

Bartle seriam a explanação geral das motivações Sociais e de Imersão em Yee, entretanto Yee aponta proveniências mais diversificadas e com diferentes agrupamentos para ambos os grupos.

Já os tipos de diversão de Hunicke et al. encaixam-se nas categorias de Yee, com maior ou menor precisão. Por exemplo, fantasia é perfeitamente correspondente a role playing, mas narrativa pode estar presente em toda categoria imersão, enquanto sensações é tão ampla que pode facilmente estar diluída entre todas as divisões de Yee já que todas foram apontadas justamente pelas sensações agradáveis identificadas pelos sujeitos em diferentes ações.

Ao observarmos com mais atenção as motivações encontradas na pesquisa de Yee, perceberemos que os dois pilares da motivação por trás do jogo, previamente estudados pela a teoria da diversão de Koster quando do flow de Csíkszentmihályi – diversão e desafio – parecem dizer respeito apenas à primeira categoria de motivações de Yee (conquista). Isto porque os sistemas recorrentes na teoria da diversão e no flow estão claramente presentes nas mecânicas de jogo, as quais estão relacionadas às motivações da categoria conquista, enquanto as motivações sociais e de imersão de Yee não são previstas por Koster e Csíkszentmihályi. Desta forma, podemos observar que, por melhor que Koster e Csíkszentmihályi compreendam a diversão de certas partes do jogo, utilizando apenas a sua base teórica, não conseguiríamos abranger todos os aspectos que são considerados divertidos e interessantes em um jogo.

No meu mestrado, fiz um gráfico tentando conciliar todas essas teorias, que você pode conferir abaixo. Hoje, eu acho que é mais útil vocês terem uma noção geral de cada um destes autores e usá-los para melhorar sua própria capacidade analítica. Cada autor trás consigo uma visão única e bem estruturada sobre porque jogamos, e cada um deles te permite ver coisas únicas sobre jogos. Essa mesma força, entretanto, trás consigo uma fraqueza que cada teoria tem em si mesma. Imagine assim: jogos são uma maçã em cima de uma mesa e cada teoria é uma pessoa olhando para essa maçã de um ponto de vista diferente. Cada um destes pontos de vista permite uma visão clara e única do que está a sua frente, entretanto a maçã é um objeto tridimensional e, portanto, possui um lado oculto para cada um destes observadores. Conhecer todas essas teorias te permite um entendimento mais completo da mesma forma que pegar a maçã e rodá-la, vendo assim todos os lados e sentindo sua pele contra seus dedos, te permite decidir se ela está boa ou não para seu consumo. Evite comprar maçãs baseados em uma única foto ou imagem de relance, sempre procure mais informações antes de dar sua análise como terminada.



Weiller (2015)

GAME DESIGN

Ok, já falamos sobre o que é ou não um jogo e o por que jogamos. Agora, finalmente, podemos falar sobre esse tal de game design. Chamamos de game design todo o planejamento e execução a respeito do *funcionamento* do jogo *ao ser jogado*. Não estamos falando do funcionamento do jogo como software, se ele começa ou se apresenta bugs ao ser iniciado, se foi feito na Unreal ou na Unity; esse tipo de funcionamento impacta no game design, mas não é uma questão do design.

O funcionamento que me refiro aqui é como o jogo se comporta enquanto sob domínio do jogador: o que acontece quando esses botões são apertados? E quando os soltamos? Qual a altura do pulo ou a intensidade e alcance do golpe? Onde se passa essa fase e o que faço com esse item? Tudo que diz respeito ao que acontece *dentro* do jogo e como acontece diz respeito ao design. Com essa definição, parece que tudo que faz de um jogo *um jogo* passa pelo design e você não estaria errado aí. Nos definimos jogos usando a interação (e mais uma ou outra regra) como parte central e o design é a forma que usamos para projetar essa interação.

As partes do Design

Uma área tão grande e importante dos jogos com certeza deve ter algumas divisões e se você pegar um livro sobre o assunto irá encontrá-las e tudo ficará bem, certo? Certo, desde que

you don't decide to read *um segundo livro sobre*. Se você resolver que quer saber mais e partir para o próximo livro, ficará desapontado ao ver que o próximo autor não usa as mesmas divisões que o primeiro. E se decidir ler mais do que dois livros (que escândalo!) no assunto, para finalmente decidir essa questão de uma vez por todas, vai ver que o terceiro escolhe uma outra divisão ainda e que **cada autor define e divide game design de formas completamente diferentes**. Alguns autores associam o design mais às mecânica, design de levels ou aos balanceamento de jogo. Outros, podem associá-lo a contar histórias em jogos e designs de personagens. Também existem autores, e esses parecem ser a maioria, que preferem analisar o design de jogos atrelado aos diferentes gêneros de jogos disponíveis, como FPS, estratégia, plataforma, etc. sem dar uma divisão clara ao design em si.

Não há nada de errado nisso, é até parte do charme da área em que tudo é tão novo que ainda não há consensos nem sobre como estudar ela mesma, mas isso cria uma dificuldade quando precisamos *conversar* sobre design entre nós, designers de jogos. Essa dificuldade é ainda maior quando precisamos ensinar e aprender game design: como discutir elementos do game design se nós não conseguimos sequer concordar com quais elementos são esses?

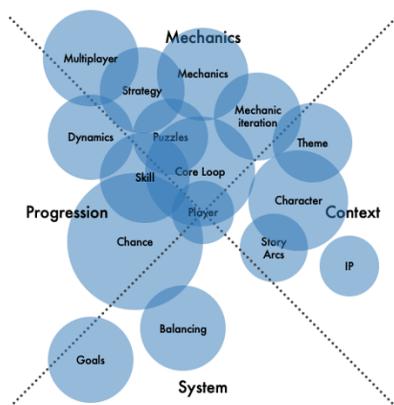
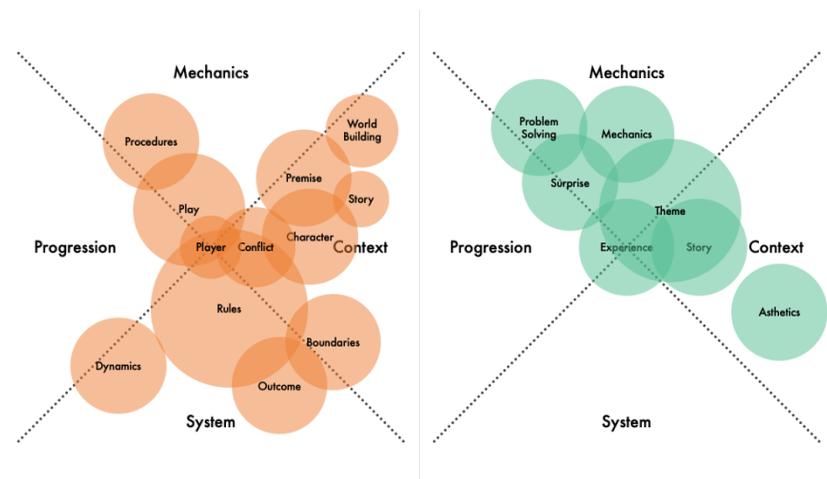
Com a falta de consenso na academia, eu achei melhor partir para indústria. Afinal, empresas precisam conseguir comunicar claramente o que eles esperam de um profissional para conseguir contratá-lo. Na área de game design, empregos geralmente caem em quatro principais áreas: (1) designer generalista ou designer de mecânicas e de gameplay, (2) level

designer, (3) designer de narrativa, designer de mundo e de história e (4) designer de sistemas.

Os designers de (1) mecânica/gameplay focam em como o jogo funciona. Quando o jogador pega o controle, o que ele pode fazer? E como? Quais são as consequências? O que eles não podem fazer? As principais regras e dinâmicas são estabelecidas por esses profissionais. O designer de level, área que chamaremos de (2) progressão, é o curador das interações dos jogadores com o trabalho de todos os outros. Este designer prevê como o jogador progride no jogo e quando e onde novas mecânicas e funções podem ser introduzidas. Isso também envolve guiar os jogadores ao longo de certos caminhos das seções do jogo e decidir o fluxo e refluxo da jogabilidade e dos sistemas no jogo. Um designer de narrativa e de mundo ou de (3) contexto é aquele que aplica significado a todas as coisas no jogo. O porquê do que o protagonista está fazendo o que está fazendo ou quais são os temas e sentimentos deste jogo são sua principal responsabilidade. Os designers de (4) sistema são aqueles que atribuem valor a essas ações ou itens ou peças no jogo e os equilibram de uma maneira que faça sentido com o que se espera que seja a experiência do jogo.

Eu analisei alguns livros de game design para levantar algumas das divisões utilizadas³ e comparei esses resultados com os títulos de vagas de emprego na área de game design mais comuns e o resultado são estes gráficos:

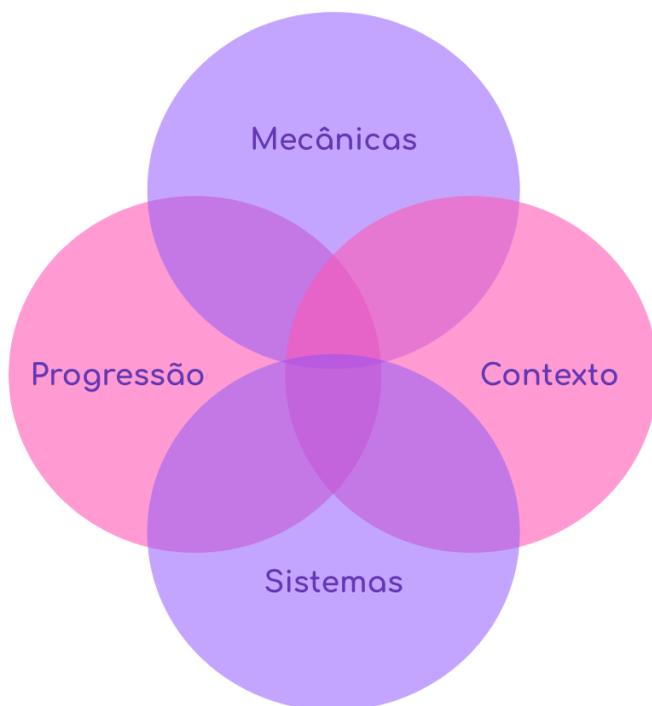
³ Esse processo está mais detalhado em artigo que publiquei sobre a forma que explico game design das minhas aulas (Weiller, 2021). Se quiser mais informações sobre isso, [veja o artigo inteiro!](#)



Cada esfera é uma das divisões de game design dos autores (em ordem, Fullerton, 2008, Brathwaite et al., 2009, e Schell, 2008) e relação aos quatro eixos de empregos mais comumente encontrados na esfera de game design.

Para facilitar nossa conversa, eu irei usar a divisão baseada na divisão de vagas de emprego em game design (mecânicas,

progressão, contexto e sistemas) para falarmos de game design. Cada capítulo daqui em diante irá tratar de um destes elementos em mais profundidade. É importante sabermos que esta divisão escolhida, assim como todas as demais divisões que apontamos, é uma divisão arbitrária criada por um objetivo que neste caso é facilitar a conversação e não uma classificação de cada parte do design do jogo. O mérito desta divisão em relação às que outros autores usam não se deve a sua novidade, muito pelo contrário, mas sim pela forma que já é informalmente usada pelos próprios profissionais da área há pelo menos duas décadas, o que prova sua aplicabilidade e versatilidade para área e aumenta os espaços (profissional e acadêmico) em que é entendida.



Outro ponto relevante importante que devemos lembrar diz respeito a aplicabilidade desta divisão quando analisamos componentes isolados de um jogo. Cada parte do design do jogo não pode ser totalmente separado do resto do jogo e encaixado em algum destes elementos que apontamos (mecânicas, progressão, contexto e sistemas), muito pelo contrário: cada parte do jogo faz parte de todo o resto e tem repercussões em cada uma das outras áreas. Mas isso não impede que cada uma dessas partes esteja mais afinada com um dos elementos.

Por exemplo, se falarmos de um personagem de um jogo. Claramente, seu funcionamento diz respeito mais ao elemento do *contexto*, afinal ele é um agente narrativo da história do jogo. Entretanto, esse personagem também pode ter uma forma única de combate ou é parte de uma *party* com o jogador (*mecânicas*), ele pode evoluir e ganhar novas habilidades junto ao jogador (*progressão*) e todas essas informações precisam ser guardadas e balanceadas de alguma forma no jogo (*sistemas*). É por isso que em cada capítulo vamos falar das principais competências deste elemento dentro do game design, alguns de seus principais conceitos e quais suas necessidades.

MECÂNICAS

Neste capítulo, iremos nos focar no entendimento das mecânicas, ou seja, o funcionamento principal dos jogos.

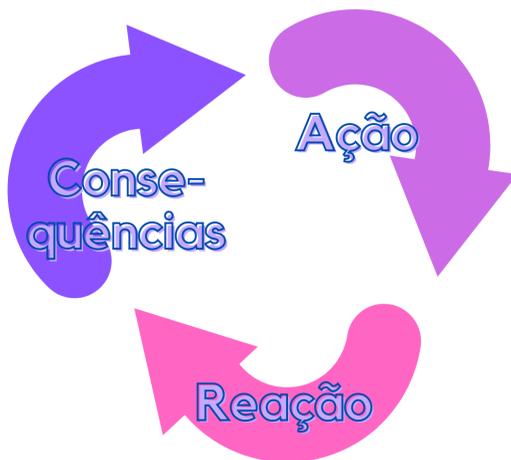
Se você ainda possui alguma dúvida sobre como entender e como ver as mecânicas de jogos, pense no que o jogador pode fazer. Imagine o seu jogador com o controle em suas mãos, cada botão que ele aperta o que acontece na tela? As mecânicas são a forma que o jogador interage com o mundo do jogo e como o mundo interage de volta baseado nessas ações que o jogador fez. É como se a mecânica fosse o coração do jogo, permitindo que todo resto funcione, ao mesmo tempo que determina as regras de funcionamento dos demais elementos.

Mecânicas são tão importantes que nós definimos gêneros de jogos baseado nos tipos principais de mecânicas. Quando falamos de um jogo do tipo FPS nós estamos nos referindo a forma como jogador se desloca no mundo 3D com a câmera em primeira pessoa e seu objetivo geralmente é atirar nos demais jogadores ou NPC. Para definir os gêneros dos jogos, o próximo conceito será extremamente importante.

Core Loop

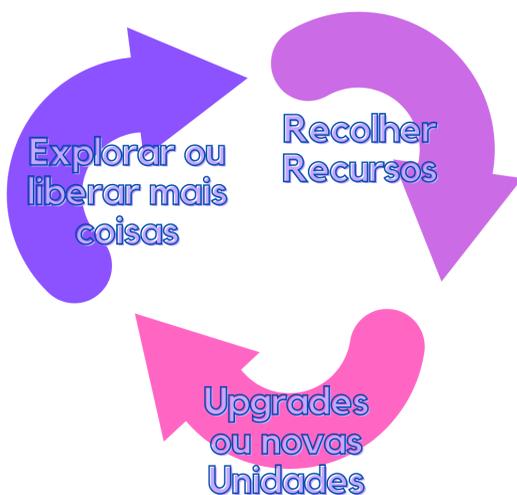
Se as mecânicas são o coração, o core loop é a batida dele, é o movimento repetitivo que mantém o jogo vivo. O core loop é o conjunto de ações principais que o jogador está

constantemente fazendo em qualquer momento do jogo. Preferencialmente, cada uma das ações do core loop se complementa interferem na ação seguinte em forma circular. Desta forma, cada vez que o jogador toma uma decisão e faz algo (1), essa decisão tem uma reação no mundo do jogo (2) que pode ser positiva, negativa ou neutra e muda a situação do jogo (3) de forma que irá influenciar a próxima decisão do jogador.



Vamos imaginar um jogo do gênero RTS por exemplo. Por mais complicado que o RTS seja, a primeira coisa que o jogador geralmente faz é explorar e recolher recursos. É por isso que na maioria dos mapas na maioria dos títulos de RTS, o jogador começa tendo uma ou mais unidades que recolhem recursos e nem sempre com as unidades responsáveis por ataque e

defesa, as quais ele geralmente pode construir depois, com os recursos que coletou. Tendo estes recursos, o jogador pode escolher como e onde gastá-los, seja com mais unidade coletoras ou em novas de ataque, seja em construções ou evoluções de tecnologia. No caso do RTS, o jogo não dá um feedback direto logo aqui se a decisão foi positiva ou negativa pois RTS são jogos em que a estratégia ao longo da partida toda influenciam no resultado final, mas o jogador tem um feedback direto sobre quando as unidades e upgrades ficarão prontos de quando poderá usá-los, caso não seja imediato. Assim que toma a decisão, o jogador se depara com o mapa do jogo, agora povoado as consequências dela (novas unidades, pesquisas sendo conduzidas, construções progredindo, por exemplo) e deve tomar novas decisões levando-a em conta, como dar ordens para as novas unidades ou ver se tem recursos o suficiente para construir outra obra.



Repare que o core loop não define todas as ações nem a complexidade permitida pelo jogo ao seu jogador em qualquer momento do jogo. Por exemplo, considerando um jogo do gênero RTS como Age of Empires, cada uma das decisões se baseia em muitos outros elementos como o posicionamento das demais civilizações e as micro-ações de suas unidades, mas, essencialmente, a todo momento o jogador se vê voltando às ações do core loop.

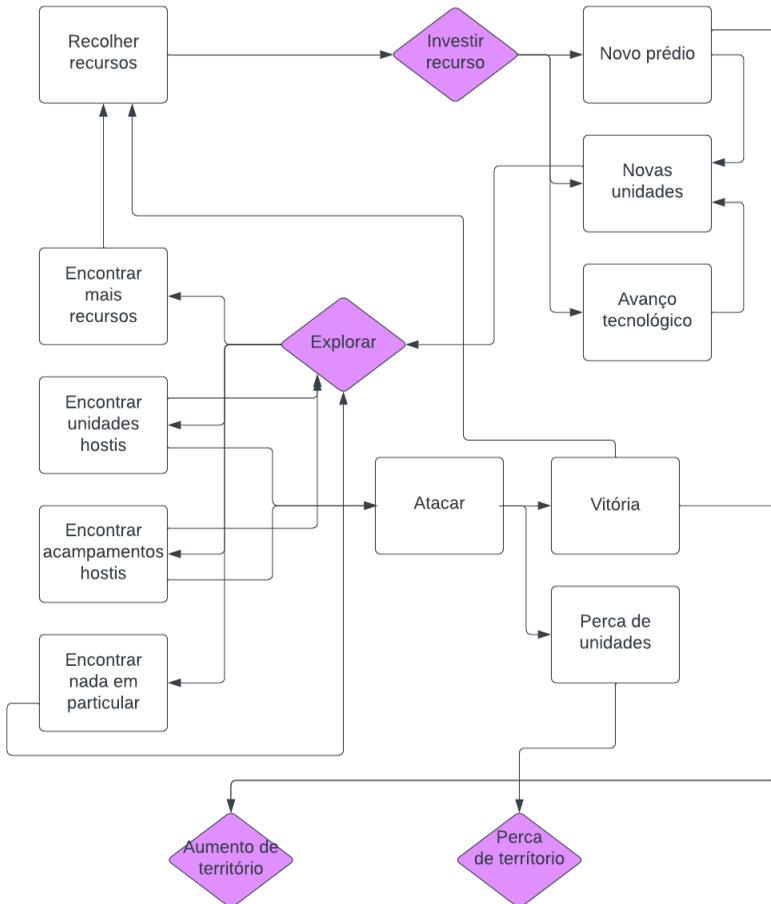
O conceito de core loop pode parecer nebuloso em jogos com muitos momentos com mecânicas diferentes, como é o caso de muitos jogos AAA, como por exemplo jogos de mundo aberto. De fato, em jogos assim, o jogador pode escolher qualquer área do jogo para ser sua parte favorita e a parte principal do jogo, da mesma forma que esse mesmo jogador pode passar horas sem fazer o que é considerado a principal mecânica. Pessoalmente, em jogos assim, eu escolho a mecânica que os desenvolvedores parecem ter focado mais tempo desenvolvendo (que, geralmente, também é o que está no gênero do jogo) ou a principal coisa que você tem que fazer para continuar progredindo no jogo. Desta forma, eu consideraria Mass Effect e GTA como shooters (mesmo com a grande parte de RPG de cada um, o foco do jogo está no combate e o combate destes jogos é tirinho) e analisaria essa parte para fazer o core loop, enquanto em Minecraft eu possivelmente faria um loop misto entre minerar e lutar (é praticamente impossível chegar no End sem ter matado alguns Enderman).

Loops Auxiliares

São todos os outros loops que trazem mecânicas que complementam o core mas que não são parte da mecânica ou funcionamento essencial do jogo. Por essa descrição, pode parecer que são menos importantes, mas se você para para pensar que *absolutamente todo o resto das coisas que você faz em um jogo que não faz parte do core loop* faz parte dos loops auxiliares. A diferenciação entre jogos do mesmo gênero, por exemplo, ocorre principalmente entre os diferentes loops auxiliares presentes nestes jogos o que dá a cada jogo suas características únicas. Por isso, não dê menos atenção a cada uma dessas outras mecânicas, já que são elas que vão dar ao seu jogo a cara dele, o motivo dele existir⁴. Criei uma simplificação de um RTS genérico na imagem a baixo.

Os loops auxiliares também podem ter durações bem diferentes do que o core loop. Enquanto o core loop é o conjunto de ações que o jogador está constantemente fazendo (ao menos, durante o momento do gameplay principal), os demais loops podem tanto dizerem respeito a mecânicas complementares a esse loop e que ocorrem ao mesmo tempo quanto loops mais longos que envolvem várias permutações do core loop.

⁴ Para mais sobre motivo de existir ou *leitmotif*, veja Weiller (2017).



No exemplo do RTS exemplificado no gráfico, o core loop seria encontrar, coletar e gastar os recursos. Essa simplificação do core loop infere o gasto dos recursos em unidades de baixo custo, que auxiliam na obtenção de mais recursos, como peões. Entretanto, essa não é a única forma de se gastar recursos. Pode-se, também, comprar diferentes unidades,

como unidades ofensivas que servem para defesa, ataque e exploração do mapa, ou construção de prédios e pesquisas tecnológicas que desbloqueiam diferentes unidades e opções. Dependendo do custo destas opções, o jogador deve esperar desde alguns segundos até metade da partida ou mais para poder toma-la.

Mecânicas Concorrentes

Quando duas mecânicas diferentes, que podem estar no core loop ou nos loops auxiliares, consomem o mesmo recurso dentro do jogo podemos chamá-las de mecânicas concorrentes. O desafio vem de balancear entre as duas ações desejadas dentro deste mesmo recurso para o jogador. Um exemplo disso se vê no jogo Papers, Please! em que o jogador possui o objetivo de resistir a cada dia dentro da alfândega de seu país aceitando pessoas que possuem os papéis e documentos necessários para entrada dependendo das exigências que mudam diariamente. O jogador recebe por cada pessoa processada neste ponto de entrada e deve receber diariamente o necessário para pagar as contas de sua família e mantê-los vivos e saudáveis. A cada dia, aparece um resumo do quanto jogador recebeu pelo dia de trabalho e quais são as despesas de seus familiares. Caso o dinheiro do dia não seja suficiente alguns dos serviços como comida e aquecimento devem ser cortados, o que pode levar a doença até a morte de seus familiares. A mecânica concorrente vem do fato de que às vezes, fazer o que é correto dentro da burocracia exigida da

posição do jogador não é o moralmente correto entre os personagens que se apresentam perante a ele.

Mecânicas concorrentes são essenciais para os loops de um jogo pois obrigam o jogador a tomar *decisões* sobre o que é mais importante para ele naquele momento ou qual estratégia a longo prazo ele deve tomar.

Loops de Feedback

Cuidado! Esse não é o mesmo feedback que vimos no capítulo de por que jogamos, que são sinais que nós percebemos e entendemos como sinais de como nosso desempenho, seja num jogo ou entre outras atividades.

Neste caso, o feedback não se refere a algo comunicado ao jogador mas sim a algo que influencia o gameplay baseado no desempenho do jogador. Existem dois tipos de loops de feedback, o positivo e o negativo: o positivo é quando há cada vez mais incentivos para algo que está acontecendo continue acontecendo, enquanto o negativo é quando existem empecilhos para que o ciclo se repita.

Parece difícil, mas não é se formos ver um de cada vez. O exemplo mais clássico de loop de feedback positivo é o do jogo de tabuleiro Banco Imobiliário, no qual quanto mais propriedades e companhias um jogador tem, maior a chance dos demais jogadores pararem nas mesmas e terem que pagar o tal jogador, aumentando os fundos deste jogador para

adquirir ainda mais propriedades. Em um jogo com mais de um jogador, um loop de feedback positivo cria um desequilíbrio que torna cada vez mais difícil que jogadores que tiveram azar no começo ganhem.

Banco Imobiliário, originalmente chamado de The Landlord's Game (ou, o Jogo do proprietário), foi feito por Lizzie Magie desbalanceado de forma proposital pois era uma crítica ao sistema capitalista. Ela queria mostrar de maneira simples e didática as consequências da lei de Ricardo e do livre mercado desregulado, e acredito que foi tão efetiva nisso que teve seu design roubado por uma empresa maior num clássico exemplo ainda maior da situação que queria trazer luz à.

Deixando a história do jogo de lado, que é muito interessante mas não vem ao caso, mesmo que o loop de feedback positivo de Banco Imobiliário seja intencional, isso não torna seu uso menos problemático. O loop de feedback positivo é uma ferramenta extremamente útil para o game designer, mas seu uso em jogos multiplayer deve ser aproximado com muito cuidado: ao mesmo tempo que dá uma vantagem e um possível sentimento de satisfação para o jogador que está na frente, diminui as chances dos demais jogadores de alcançá-lo e torna a competitividade nula. Não há entretenimento em jogar um jogo cujo o resultado já está definido.

O loop de feedback negativo é o oposto do positivo: quanto mais o jogador está próximo de algo, mais o jogo torna essa posição improvável de se manter. Um excelente exemplo dele está em Mario Kart, em que quanto melhor colocado um player

é, menores são as chances de ele receber benefícios do jogo na forma de power-ups mais poderosos. Os jogadores em último lugar, entretanto, têm a chance de receberem os melhores power-ups do jogo, o que lhes permite virar o jogo a qualquer momento e manter o jogo competitivo.

Riscos e Informações Assimétricos

Simetria se refere a o quão semelhantes duas ou mais partes são. Uma reflexão em um espelho plano e sem distorções é perfeitamente simétrica a imagem ou objeto que está refletindo, mesmo que seja oposta. Balancear jogos é algo complexo, jogos multiplayer então nem se fala, mas torna-se mais simples se há simetria envolvida. Quando os jogadores começam com as mesmas habilidades, recursos e localização, temos jogos simétricos.

Xadrez e damas, por exemplo, são jogos praticamente simétricos com ambos os jogadores começando com a mesma quantidade de peças, igualmente posicionadas e com as mesmas habilidades. A única assimetria em ambos os jogos é que, por serem jogos baseados em turnos, um jogador joga antes do outro, o que lhe dá uma pequena vantagem. Em jogos digitais é mais comum termos perfeita simetria de design pois só temos que copiar assets e balanceamentos para ambos os lados em um jogo que não seja em turnos. Um jogo FPS, por exemplo, desde que o mapa seja espalhado pode ser perfeitamente simétrico ou nos jogos da série Warcraft em que

as facções humanas e orcs são, em geral, equivalentes uma à outra com unidades com papéis e atributos semelhantes.

Jogos assimétricos são, por consequência, jogos em que os jogadores têm habilidades, unidades e/ou posicionamentos diferentes. Um exemplo é outro jogo RTS também da Blizzard, StarCraft. Em StarCraft há três principais facções, cada uma com habilidades e estilos de jogo únicos. Os Zergs são aliens que se reproduzem rápido, fazem muitas unidades em pouco tempo embora todas sejam “fracas” eles podem dominar o terreno com sua velocidade. Os Protoss, por outro lado, são um grupo de aliens que demora bastante para fazer ou construir qualquer coisa e cada unidade consome bastante recurso. Por fim, os Terranos, que são os humanos do jogo, tem atributos mais equilibrados e representam um meio termo entre Zergs e Protoss.

Devido a quantidade de componentes que podem existir de cada lado, jogos assimétricos são muito mais difíceis de se balancear que os simétricos. Se uma das unidades em um jogo simétrico está desbancada, ao menos ambos jogadores terão acesso a ela e isso acaba por balancear a partida por mais que o jogo em si não esteja totalmente balanceado. O mesmo não pode ser dito dos assimétricos, pois cada jogador tem opções de unidades completamente diferentes entre si. Da mesma forma, jogos assimétricos apresentam um fator de replay maior, pois jogar com cada uma das opções diferentes apresenta uma experiência de jogo completamente diferente.

Gameplay Emergente

Criamos os sistemas e mecânicas que regem nosso mundo fictício do jogo pensando na forma que eles vão servir o objetivo do protagonista e do jogador. Entretanto, isso não significa que essa será a única forma que eles podem ser usados. Seus jogadores podem descobrir novos usos que podem tornar o jogo mais interessante para eles, mas que podem tanto ser benéficos e interessantes para o jogo como um todo quanto podem destruir a experiência ou a mensagem esperada.

A forma mais comum de gameplay emergente é quando jogadores descobrem algum tipo de exploit ou benefício inesperado usando um ciclo de ações dentro do jogo, mecânicas que se complementam em um loop de feedback de forma que os desenvolvedores não esperavam. Toda vez que um novo campeão é adicionado a um MOBA, as chances disso acontecer aumentam na magnitude da quantidade de quantos outros campeões estão disponíveis. Por isso, não é de se estranhar que às vezes no lançamento de um destes personagens as vezes ainda há alguns bugs e exploits quando o campeão interage com um determinado outro campeão.

Um outro exemplo de mecânica emergente é o que se encontra no jogo GTA quando jogadores começaram a combinar as mecânicas de contratar serviços de uma profissional do sexo para aumentar seu hp e a de atropelar pedestres para conseguir o dinheiro deles. Fazendo uma destas ações logo em seguida à outra levou ao gameplay emergente de usar o

serviços de uma profissional do sexo e atropela-la logo em seguida para se recuperar o hp sem custo monetário. A problemática moral destes ciclo de ações que parece se complementar é evidente, esses profissionais estão entre os grupos mais vulneráveis e vitimizados pela violência, mas durante a época de design e de teste a associação destas duas mecânicas passou batida pelos desenvolvedores. Após sua exposição pela mídia, os mesmo tiveram que retirar algumas destas mecânicas para interromper este loop (Rodenberg, 2013).

Resumindo

Estes são alguns exemplos de termos e conceitos que vão ser essenciais para discutirmos e desenvolvermos jogos. E não se preocupe se estes conceitos ainda não vêm muito facilmente para você: é muita informação de uma vez e é normal esquecer ou misturar algumas coisas. Sempre que precisar, volte a ler esse texto e leia a explicação do termo que está precisando.

PROGRESSÃO

Já sabemos o que são jogos, o que os classificam e do que são compostos. No último capítulo, discutimos o funcionamento de jogos sob a forma de mecânicas. Hoje, vamos falar sobre como planejar e estruturar essas mecânicas ao redor da experiência do jogador.

A progressão leva em consideração tudo que você, como desenvolvedor, tem de pensar e organizar para que seu jogador tenha a melhor experiência possível durante sua experiência de jogo. Para tanto, você deve planejar: a forma que ele encontra novos desafios, com qual frequência eles são encontrados, posicionamento de determinados inimigos, uso de itens e com qual seu jogador é recompensado, dentre outros elementos dessa experiência.

A forma mais comum de progressão é a level design, que basicamente é a progressão representada em um ambiente 2D ou 3D do seu jogo, mas jogos que não possuem o que tradicionalmente entendemos como level também tem progressão. Nesse capítulo vamos falar de alguns conceitos essenciais para se planejar a progressão, como o aprendizado do jogador e o ritmo do jogo, e também algumas ferramentas para planejar a boa distribuição de tudo.

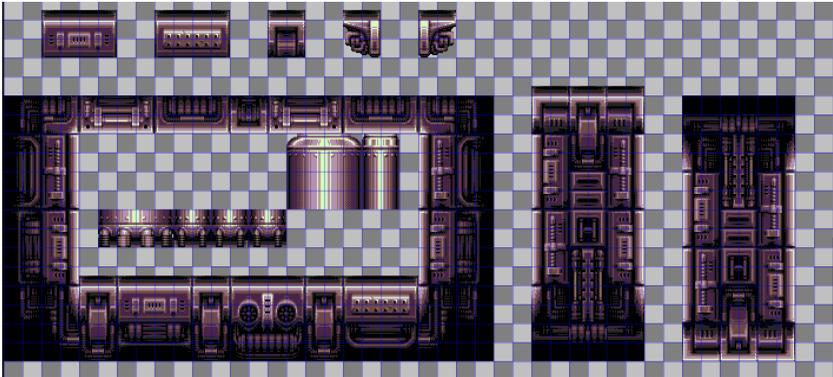
Técnicas de construção modular

Vamos imaginar que você precisa construir uma fase. O que você precisa? Possivelmente, você pensou em elementos de chão, parede, objetos... E de onde vem estes elementos? Oras, da equipe de arte, com certeza. E qual padrão de tamanho e escala cada um desses elementos vai ter?

Nesse momento, precisamos decidir se nosso jogo vai ter fases com arte única e não padronizada (o que pode até ficar mais elegante, mas dificulta muito o processo de fazer fases) ou se vamos usar a construção modular (mais difícil de decidir as regras no começo, mas depois que os assets estão prontos é muito fácil fazer muitos níveis).

Para começar a fazer design modular, imagine qual a unidade mínima necessária para construir. Qual a menor coisa que você vai precisar colocar no mapa que é interagível e comece a partir dela. Todas as outras coisas tem, obrigatoriamente, que encaixar no mesmo grid desta coisa.

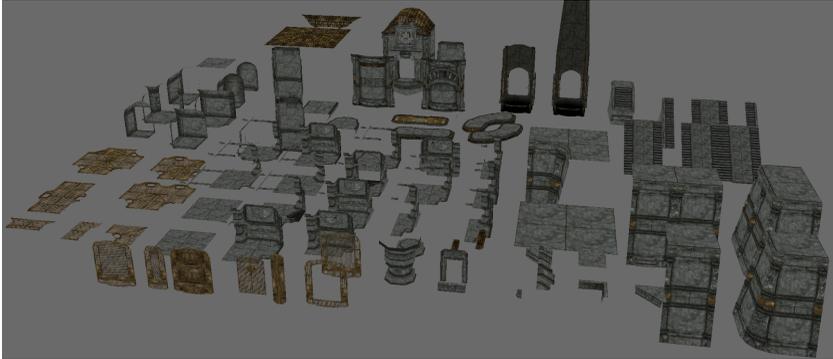
Isso é bem mais fácil de entender considerando-se um jogo 2D, como por exemplo Vengentful Guardian MoonRider. Abaixo, você pode ver alguns dos *tiles* (quando usamos o sistema modular, chamamos cada peça de mesmo tamanho em um jogo 2D de *tile* que quer dizer azulejo) de cenários da primeira área do jogo e consegue ver claramente o grid que limita sua construção. Quando todas as peças se encaixam utilizando o mesmo grid, fica muito mais fácil montar níveis.



Exemplo de tiles com linhas do grid aparecendo.

Como você pode ver, alguns desses sprites são menores do que o grid, mas eles encaixam no grid utilizando-se transparência. Alguns outros, são maiores do que um quadrado do grid e se resolve isso utilizando mais de um quadrado para criar o desenho. A beleza de se utilizar o design modular é por ser muito mais simples e prático pegar cada uma dessas peças e montar a fase do que montar a fase e pintar a arte em cima, que não poderá ser reutilizada em outras áreas

Mas só porque é mais fácil imaginar construção modular em 2D, não significa que ela não seja usada em 3D. Imagine modelar cada árvore e cada grama, pintar cada textura de pedra ou de solo. Muito pelo contrário, a maioria dos jogos 3D usam esta técnica para viabilizar a construção rápida de muitos levels. Se você já mexeu com o editor de levels de qualquer jogo 3D sabe exatamente do que estou falando. Abaixo, você pode ver alguns dos módulos utilizados na construção do level design de Skyrim.



Módulos facilitam o trabalho do departamento de arte e o seu como designer, pois depois de algum tempo construindo levels utilizando a limitação dos módulos você começa a pensar em módulos. Se é preciso fazer um corredor curto, imediatamente você já sabe que vai precisar três módulos de corredor, se é preciso uma sala com três saídas você já sabe qual modelo e quais paredes utilizar para dar o fluxo correto. No começo, parece uma limitação chata, mas com o tempo você aprende a utilizá-lo como sua melhor ferramenta de trabalho.

Utilizar módulos é a forma mais comum de se fazer fases, mas não é a única. É possível fazer artes únicas para áreas únicas, o que aumenta consideravelmente o trabalho, mas dependendo do jogo é algo que vale a pena. Por exemplo, jogos do estilo point-and-click em que o jogador passa muito tempo em cada fase, interagindo e mexendo nos objetos são candidatas perfeitas para esse tipo de design.

Mesmo em outros jogos, algumas áreas em jogos que usam principalmente level design modular há alguns pedaços ou objetos que não são modulares e servem para ajudar o jogador

a se localizar ou para contar um pouco da história. Prédios, estátuas, montanhas, árvores ou qualquer outro modelo que seja memorável e será utilizado uma ou poucas vezes para mostrar para o jogador onde ele está e por que essa área é especial. E é aqui que sua prática em utilizar os módulos para level realmente vai brilhar: depois de se acostumar com todos os módulos no seu arsenal como se fossem um pincel de artista, você sabe exatamente quais áreas estão precisando de um asset único, feito especialmente para elas.

Aprendizado

Como vimos anteriormente, jogamos por vários motivos diferentes, mas um dos principais é que jogos são caixinhas de aprendizado planejadas para nos entreter. A diferença entre este aprendizado dos jogos com o aprendizado de outras coisas que não gostamos tanto é a forma que este aprendizado ocorre. Em jogos, nós designers planejamos a quantidade de novas informações apresentadas ao jogador de forma que não o sobrecarregue, e só deixamos o jogador continuar quando ele entender e praticar bem. Fazemos isso planejando as mudanças e upgrades e também fazendo testes com jogadores, e essa frequência de novidades é a principal base da dificuldade do jogo.

É importante, também, deixar claro ao jogador quando suas ações estão resultando em comportamentos desejados e quando não, e recompensá-lo pelo sucesso.

Para entender melhor como organizar tudo isso, leia os capítulos 8 e 9 do Game Start (Weiller, 2015).

Ritmo

Um jogo em que o desenvolvedor não considera a progressão do jogador é como uma música com uma nota só, repetida incessantemente até o fim. Igualmente, um jogo tornando-se progressivamente mais difícil e só isso é como uma música em que as notas vão progressivamente subindo na escala musical sem nenhuma surpresa ou imprevisibilidade. Desta forma, nosso objetivo como level designers não é fazer uma progressão meramente homogênea, mas sim, como uma boa canção, uma progressão com ritmo.

O ritmo do jogo é intimamente relacionado com o contexto do mesmo, que veremos no próximo capítulo, mas por enquanto tente apenas se tentar se lembrar de como foi sua última jogatina. Você manteve a mesma sensação durante todo o momento? Ou houveram momentos que você estava tranquila, em outros surpresa e em outros ainda na borda da sua cadeira de tão tensa que estava a situação? A essas mudanças entre o nível de tensão e as sensações gerais trazidas pelo jogo chamamos de ritmo e como um desenvolvedor você tem que ficar atento a elas.

Em algum momento no começo do desenvolvimento, você vai fazer uma tabelinha planejando o jogo todo. Nas linhas, você coloca os momentos do jogo, e nas colunas coloque os

elementos que você acha importante para planejar sobre. Os elementos dependem de jogo a jogo, mas por exemplo podemos colocar uma para história, uma para novas mecânicas, uma para itens disponíveis... Caso seu jogo seja dividido em fases, as linhas podem ser as fases.

| Level | Nova mecânica | História |
|-------|---------------------|--------------------|
| 01 | Combate Melee | Chamada à aventura |
| 02 | Combate Ranged | Primeiro cerco |
| 03 | Múltiplos inimigos | Primeira batalha |
| 04 | Inimigos com escudo | Chegada na capital |
| 05 | Chefão | Invasão do castelo |

Exemplo de tabela de planejamento do jogo. É algo simples que te ajuda a ter uma visão geral do jogo todo e conseguir visualizar áreas que estão sobrecarregadas de conteúdo ou de novos aprendizados antes de você começar a produzi-las.

Linearidade e sensação de interação

Uma das maiores diferenças entre jogos e outras mídias é a capacidade do jogador interagir com o jogo e até encontrar momentos, caminhos e histórias diferentes dependendo das escolhas que ela tomou anteriormente. Entretanto, cada novo galho de narrativa ou de caminho, é uma nova possibilidade para o jogador explorar que poderá ou não ser encontrada, mas que com certeza terá que ser feita pelos desenvolvedores e testada pelo QA. Imagine a tabelinha que aprendemos ali em cima e imagine colocar novas informações a cada opção que o jogador pode fazer, seja de novas fases ou mecânicas ou diálogos.

Resumindo

A progressão é a sua principal interface com o jogador, é por meio dela que você comunica cada um dos outros elementos na ordem e intensidade esperada para causar a sensação que você deseja. Justamente por isso, também é o elemento do jogo que mais precisa ser testado e, preferencialmente, pelo máximo de jogadores que ainda não conhecem seu jogo: nada como um par de olhos novos para encontrar uma inconsistência.

CONTEXTO

Até o momento, entendemos como os jogos funcionam e como planejar esse funcionamento de forma que o jogador se mantenha engajado com o jogo, nos referindo à funcionalidade do jogo. Entretanto, funcionar não é o único atributo de jogos e não é o único motivo pelo qual jogamos. Neste capítulo, falaremos do contexto que se refere a tudo dentro do jogo que dá sentido à jornada do jogador, seja sonoro, visual ou narrativo.

O essencial para se entender sobre o contexto do seu jogo é qual a sensação que você quer passar para o seu jogador. Claro, idealmente um jogo não trás ao jogador uma única sensação e sim uma grande gama delas. Conversamos superficialmente sobre isto no último capítulo quando falamos sobre ritmo e como controlá-lo de forma a tornar momentos no jogo mais ou menos tensos e mais ou menos significativos para a jornada do jogador, mas naquele momento discutimos como fazer isso utilizando apenas mecânicas. Seja pela introdução de novos desafios ou pelo escalamento das dificuldades, as mecânicas são sem dúvidas sua principal ferramenta como game designer para definir o tom do momento do jogo. Entretanto, elas não são as únicas.

Leitmotiv

Seu *leitmotiv* é a principal mensagem do seu jogo. O porquê ele existe. E se você está em um equipe, quanto antes todo mundo concordar com essa mensagem, mais rápido vocês vão começar a realmente trabalhar nesse projeto. Eu gosto de pensar que todo jogo pode ser resumido em uma palavra, mas você não precisa ser tão reducionista. Uma frase é mais que o suficiente para entender qual o motivo guia do seu jogo.

Super Mario tematicamente é sobre um encanador se divertindo em uma terra mágica. Todo o clima do jogo é leve e divertido, o sequestro de Peach é só um *plot device* como qualquer outro e em nenhum momento impacta de forma séria ou negativa no jogo. Mas um jogo não é apenas temático, é também jogável. Mecanicamente, Super Mário é um jogo sobre pulos. Pare para pensar: tudo que você consegue fazer em Super Mário é, de alguma forma, relacionado a pulos e sua precisão em executá-los. E isso fica claro logo na primeira fase. Dessa forma, toda a estrutura de levels de Super Mário foi pensada para explorar ao máximo a mecânica do pulo, enquanto que toda a sua arte (visual e sonora) ressalta a leveza e diversão desse mundo. Um belo sanduíche de *leitmotiv*.

Alien Isolation tematicamente é sobre sobrevivência em um ambiente hostil e futurista, e essa filosofia está tanto na temática como no gameplay. Tudo no gameplay no jogo reverte para essa tensão, para a falta de segurança e poder dentro desse mundo que te quer morto. O jogo nem permite ao

jogador usar o craft dentro de esconderijos: o jogador deve se sentir vulnerável enquanto faz algo de útil e quando se esconde é obrigado a ficar sempre prestando atenção no exterior. Um sanduíche de ansiedade, mas um sanduíche intencionalmente servido assim.

Em Odallus queríamos fazer um jogo de exploração plataforma clássico, mas que também fosse simples o suficiente para não precisar de mapas para os levels. Essa simplicidade nos levou ao sistema de fases. A "classicidade" do gameplay nos levou a um tipo de história também clássica, sobre busca por poder e vingança, mas também nos levou a estrutura comum em lendas que se baseia na circularidade da história. Todos os personagens do jogo são egoístas e só se importam com seu próprio benefício, o gameplay permite que o jogador acumule poder, upgrade por upgrade. O fim de Odallus é muito próximo do começo de Odallus, trazendo a circularidade de volta.

Em Rainy Day, o *leitmotiv* era só um: passar ao jogador a sensação de viver com ansiedade e depressão. Isso guiou todas as escolhas, de tema a gameplay e arte. Eu explico isso mais a fundo [aqui](#) (em inglês).

Se você definir exatamente qual o principal objetivo ou mensagem do jogo antes de qualquer outra coisa, o porquê ele existe e quais as sensações que você quer passar, você terá a semente para o seu projeto. Não desmereça essa tarefa ou ache que está levando tempo demais: é melhor gastar tempo definindo os limites no começo, quando ainda se pode tudo, no que no meio, quando redefinir coisas significa jogar fora muito

trabalho que já foi feito. Ter as bases definidas no começo não só é uma mão na roda para evitar que o projeto fique grande demais e perca o escopo, mas também resulta em um jogo com uma mensagem bem mais clara para o jogador.

Isso não quer dizer que seu jogo não pode ter mais de uma mensagem. Dependendo a extensão dele, é até aconselhável que ele tenha várias! Mas, preferencialmente, uma tem que ser a principal para ser o seu foco no desenvolvimento e para ajudar a decidir momentos em que há dúvidas sobre onde gastar mais tempo desenvolvendo coisas. Por exemplo, no meu jogo sobre contemplação, é mais legal eu gastar mais tempo desenvolvendo um sistema de crafting ou um de combate? A escolha da mensagem principal também é um plano de contingência para caso tudo der errado e as mensagens não ficarem muito claras, pelo menos a principal, que você investiu mais tempo e cuidado, vai estar mais clara que as demais. Como desenvolvedor de jogos, metade do seu trabalho é pensar em planos de contingência para caso coisas deem errado. Por que elas vão dar errado e seu sucesso vai ser medido no índice de acerto das suas previsões.

E não se esqueça que da mesma forma que cada jogo que tem *leitmotif*, cada parte do jogo também pode ter. Pense na tabelinha que você fez no capítulo anterior e como cada unidade do jogo (levels ou outra forma de divisão) tem um motivo de existir dentro da jornada total do jogador. Adicionar uma coluna para os principais temas de cada área é uma boa ideia para ajudar todo mundo a estar na mesma página durante o desenvolvimento.

Narrativa

Cada jogo tem necessidades narrativas diferentes, um RPG precisa de elementos diferentes que um puzzle. Uma história é a combinação bem sucedida de personagens e trama e, para ser bem sucedida, os personagens devem ser cativantes e a trama deve ser coerente.

Personagens são todos os participantes do seu jogo, desde os mais importantes e com mais tempo junto ao jogador até os que só estão no fundo da tela. Um universo rico e interessante em um jogo é um universo com personagens interessante e que façam sentido entre si. Tradicionalmente em estudos literários, chama-se o personagem principal ou os personagens principais de protagonistas, mas em jogos reservamos esse nome para personagens jogáveis.

Para fazer bons personagens considere sempre suas forças, mas especialmente considere suas fraquezas. Um personagem sem fraquezas é um personagem que não é muito relacionável e carismático para o público, nós somos a soma de nossas características e algumas delas não são as melhores. Além disso, todos personagens devem ter suas próprias ambições e objetivos, e isso explica o motivo pelo qual ele se envolve com a trama e toma as decisões que toma. Esses elementos são importantes para todos os personagens, mas em especial para o ou os protagonistas, já que são os personagens com quem o jogador mais terá contato. Como um jogo é uma peça interativa, o máximo de decisões de escolhas do protagonista que você consegue colocar na mão do jogador e de fato

impactar na trama, melhor, mas ao mesmo tempo mais trabalho como desenvolvedor você cria.

A trama é o conjunto de situações e temas que se desencadeiam no decorrer da peça, que pode ser um livro, um filme ou um jogo. Uma boa trama possui começo, meio e fim claros, altos e baixos entre estes momentos que trazem conflitos a serem resolvidos pelos personagens e, enfim, que concluem com a transformação dos personagens. Essa transformação, aliás, é o que chamamos de arco do personagem e geralmente é relacionado com quem o personagem é e qual seu objetivo.

Mas além dos personagens, que são os agentes da história, e da trama, que é a sucessão de eventos que transcorrem, um bom jogo também tem um universo. Muitos jogos se passam no nosso mundo, então como desenvolvedor você não precisa gastar muito tempo estabelecendo como as coisas são, mas a maioria dos jogos diferem do nosso mundo em várias coisas. Essas variações podem ser coisas pequenas, como a existência de certas organizações ou pessoas, ou enormes, como se passar em outros planetas, a existência de outras espécies, mudanças em eventos históricos e até se passar em momentos históricos diferentes do nosso. Todas estas diferenças precisam ser apresentadas para o jogador de alguma forma, para que ele seja introduzido a esse mundo e para que os personagens e a trama façam sentido nele. Como desenvolvedores, você não precisa explicar absolutamente tudo logo no começo, isso até resultaria em uma introdução meio chata, mas o ideal é planejar quais informações são importantes serem apresentadas em quais momentos, de

forma semelhante ao planejamento de dificuldade que vimos na progressão. Prey (2017) faz isso de forma interessante. O jogo se passa em um futuro próximo (2035) mas com uma historia alternativa em que John Kennedy não morreu e por isso a exploração espacial continuou sendo uma prioridade mundial. O jogo não te fala isso de cara, mas conforme você vai explorando a estação em que o jogo se passa vai encontrando informações históricas que te introduzem a isso. Há até quadros de bancos em parques dedicados a Kennedy, nos quais consta a data de nascimento e morte do politico norte americano, indicando essa realidade paralela.

Parece muita coisa? É por que é sim, contar histórias é uma habilidade que leva muitos anos para se dominar, mas não para se praticar. Nós já nascemos contando histórias e muitas delas se assemelham a outras histórias que já existem. Muitas dessas histórias que já existem são, na verdade, arquétipos de narrativas, como a própria jornada do herói. Se você é um contador de histórias iniciante, tente se manter próximo de um destes arquétipos e não tem erro. E lembre-se: a principal diferença entre um contador de historias iniciante e um experiente é praticar contando historias. Se você não começar, nunca vai chegar em lugar algum. Se você começar, não dá para saber onde vai parar!

Estética

A arte e sons do seu jogo devem reforçar as idéias e ajudar a passar ao jogador as sensações do *leitmotiv*. Nosso papel

como game designers não nos permite lidar diretamente com a arte, visual ou sonora, pois isso é atribuição de um grupo completamente diferente de profissionais, mas saiba que você precisa ter uma comunicação direta e próxima com os mesmos. Eu sugiro que, inclusive, eles participem da elaboração da elaboração da tabela que decide as mecânicas e demais elementos de cada level, e que a estética seja também uma das colunas dessa tabela. Dessa forma, a estética ajudam a reforçar e complementar os temas de cada fase.

Resumindo

Cada jogo tem um motivo de existência (*leitmotif*) e ele tem mais chances de ser um bom jogo se todos seus elementos reforçam essa essência. Os elementos relacionados com o contexto são os que mais fortemente conseguem comunicar ao jogador essa essência, seja por meio da estética ou da narrativa e dos personagens com quem o jogador interage.

SISTEMAS

Até o momento, falamos sobre como os jogos funcionam e como organizar o fluxo de informações para o jogador ter a melhor experiência possível. Mas não falamos exatamente como as relações entre estas informações ocorrem e como são balanceadas. Todas as relações que ocorrem num videogames são registradas de forma numérica e, ao manipularmos estes números, estamos manipulando a própria experiência do jogo. Ao conjunto destes números e suas relações damos o nome de sistema. Se você já jogou algum RPG de mesa, essa diferenciação deve estar clara, mas se esse não é seu caso, vem comigo (e tente jogar um RPG de mesa o mais breve possível).

Vamos pensar no sistema de combate de um jogo qualquer. Se o ataque “soco” tira 4 pontos de hp, essa informação numérica está guardada em algum local do sistema do jogo. E se o inimigo estiver usando uma armadura ou algo que diminua o dano de ataques físicos? Por exemplo, ele pode estar utilizando um escudo que reduz 50% do dano até 10 hp. Essa equação que envolve uma validação (se o dano for inferior a 10) e uma porcentagem (50%) está guardada em outra área do sistema de combate.

Assim, podemos dizer que um sistema se refere a todas as regras, números e informações relacionados a uma área do jogo. Os sistemas de um jogo se relacionam entre si para fazendo o jogo funcionar, mesmo porque em um dado jogo

geralmente não tem apenas um sistema funcionando. No nosso exemplo do "sistema" de combate na verdade diz respeito a compilação de vários outros sistemas menores, como o próprio sistema de itens que continha as informações do escudo, os sistemas de dano de hp, os sistemas de golpes e inimigos, mas também podemos ter um sistema de quests ou um sistema de relação com os personagens que impactarão nesse sistema também.

Essa idéia de elementos textuais, como *quests* ou diálogos, fazer parte dos sistemas pode parecer contra-intuitivo a princípio, mas lembre-se que toda informação textual tem que ser guardada em algum lugar, desde o nome de assets do jogo a nomes de itens e diálogos inteiros. Praticamente tudo que você vê e interage com em um jogo está guardado em algum sistema dele!

Sistemas determinísticos e não-determinísticos

Chamamos de determinístico um sistema em que uma determinada entrada do jogador causará sempre a mesma saída ou reação. Por exemplo, em um jogo de xadrez, se o jogador decide mover o peão, o resultado desta ação é o peão sendo movido para exata casa que o jogador escolheu. Não existe carta-armadilha, contra-ataque surpresa no tabuleiro, resultado de dado ou contador de especial que fará o peão ser imediatamente movido para outro lugar diferente do escolhido ou “comido” pelo oponente. Claro, imediatamente após o fim do

turno deste jogador, seu oponente pode fazer um movimento que resulta na morte deste peão, mas isso ocorreria dentro do turno do outro jogador, seguindo as mesmas regras. Outro exemplo é *Space Invaders* (1978), em que toda vez que um tiro do jogador atinge um dos aliens (os *Invaders*), esse alien morre e deixa de existir. Cada alien tem um tiro do jogador de vida.

Diferentemente, um sistema não-determinístico é um sistema em que a ação do jogador pode saídas ou reações diferentes. Por exemplo em um jogo de RPG, quando o jogador decide atacar, o resultado não é um simples ataque. Em vez disso, o jogador precisa lançar um dado para saber o quão efetivo seu ataque foi e, dependendo do número tirado, pode até falhar em sua ação. Ainda usando *Space Invaders*, os Aliens atacam o jogador com mísseis que, quando alvejados pelo jogador, sempre tem uma chance de não serem destruídos pelo tiro do jogador. Então, um tiro certo em um míssil nem sempre significa a destruição do mesmo.

De forma direta, sistemas determinísticos são os que todas as ações tem resultados pré-determinados e esperados, enquanto nos não-determinísticos os resultados das ações sofrem influências externas, como sorte ou outros elementos que não são de controle do jogador.

Essa distinção não implica em uma hierarquia entre estas categorias, nenhuma é mais ou menos desejável ou complexa que a outra. Na maior parte dos jogos digitais, nós utilizamos uma mistura entre sistemas determinísticos e não-determinísticos para criaram experiência do jogador. Em jogos

em que todos os elementos são perfeitamente previsíveis, corremos o risco de criar uma experiência muito repetitiva ou que depende demais na habilidade dos jogadores, o que pode tornar tudo muito chato. Por outro lado, em jogos em que todos as ações passam por um filtro que foge ao controle do jogador, corremos o risco contrário de ter uma experiência muito imprevisível ou que o jogador sinta que nenhuma das suas decisões realmente importa no fim das contas.

Saber onde deixar os elementos totalmente sob controle do jogador e quando adicionar um pouco de sorte ou descontrole vindo do jogo é uma das maiores e mais difíceis responsabilidades do game design.

Grandezas

A parte de sistemas do game design pode até parecer uma parte mais técnica e exata por conta dos números, mas na verdade envolve muita sensibilidade humana. Cada número que colocamos em algo não é um número absoluto, é um valor que estabelece uma relação com os demais números do jogo. O principal objetivo destes números é fazer o jogador ter a experiência que esperamos para ele.

Por exemplo, a gravidade no nosso planeta terra é praticamente uma constante de 10m/s^2 , com variações desprezíveis dependendo de várias condições, na altura do chão. Em jogos, entretanto, esse dificilmente é o número utilizado no cálculo de queda de itens e personagem; a

gravidade em jogos é decidida baseado no que o jogo precisa, muitas vezes nem contendo aceleração real. O que realmente importa é o quanto os números reforçam a realidade do jogo e mantém o que o jogador deveria estar sentindo.

Outro ponto a se considerar sempre é o quão grande os números serão dentro de um sistema e qual a sua relação com os demais. Um item que custa 1.000 dinheiros é caro? Bom, não tem como se saber apenas com essa informação! Se o jogador ganha em média 10 dinheiros a cada 10 minutos de jogo, esse item possivelmente será caro sim. Se o jogador ganha 10.000 por cada monstro que ele mata e mata um a cada cinco minutos, daí esse item é muito barato, não é mesmo?

Escolher esses números pode pareceres-te confuso e arbitrário a princípio, e, honestamente: é mesmo. Pelo menos no começo, você vai escolher alguns valores de forma arbitrária para servirem de base para entender todo o resto do sistema. Eu recomendo aos meus alunos que sempre comecem por um dos sistemas que está no core loop, para garantir que você vai começar o balanceamento pelo que há de mais essencial no jogo. Se combate é o core, comece entendendo as funções de ataque e dano (o que vai influenciar no dano? Vai ter arma? Vai ter armadura? Quais são as relações?) e alguns dos primeiros números, como dano com ataque sem arma e com algumas das primeiras armas e o HP de alguns personagens.

Estabelecendo esse começo, já começa a ficar claro para você se você ficaria mais confortável manipulando esse sistema

com dezenas, centenas ou até decimais. Mas lembre-se que se esses números são visíveis para o jogador, evite decimais! Números com vírgulas demoram um pouco mais para ser entendidos e tendem a alienar a maioria dos jogadores, então mude nem que seja apenas a apresentação para o jogador de 1,5 para 15 de bônus de dano! Problema resolvido!

Tabelas

A forma mais simples e visual de lidar com sistemas é utilizando tabelas, que também são a forma mais prática e organizada de se guardar informações. Se você estiver utilizando Microsoft Excel ou Google Docs, é possível ter um arquivo só para o jogo todo, mas cada sistema precisa ter sua aba única. Comece pensando qual unidade de coisas é mais utilizada no seu jogo; são armas, itens, frutas? Vamos listá-las. Agora, vamos pensar quais os atributos de cada uma destas que serão importantes para seu jogo. No caso de armas em um jogo de tiro, você possivelmente vai precisar saber, além dos nomes das armas, qual o tipo de munição que ela usa, qual a velocidade em que ela atira, quantos tipos de cada vez, o dano de cada tiro e a precisão da trajetória dos mesmos. Dependendo de quais mecânicas serão implementadas, talvez o peso de cada arma seja importante para limitar a quantidade de itens e o jogador pode levar, talvez não. Se o seu jogador tiver um sistema de level, talvez seja importante colocar em qual level a arma se torna disponível para uso ou compra, talvez não. Se há um sistema econômico e as armas podem ser compradas, é interessante colocar o valor de cada uma e

em qual mercado ela estará disponível. Como você pode ver, não existe uma receita de bolo sobre como organizar seus sistemas ou quais atributos devem constar; cada jogo é um universo único e tem informações relevantes diferentes de outros jogos.

Sistemas em movimento

Uma boa forma de se testar sistemas e suas conexões é fazendo protótipos, em particular protótipos de destas conexões em particular. Um excelente programa para se testar estas dinâmicas é o Machinations, que te permite criar e testar sistemas com valores e ver como os recursos se movimentam demoram automática ou interativa. Eu poderia escrever um livro inteiro sobre o Machinations mas não vou nesse momento! Você pode aprender mais sobre ele no [tutorial oficial deles](#).

Balanceamento

Essa é a palavrinha mágica que estamos constantemente correndo atrás como game designers, mas *balanceado* é uma qualidade elusiva. Não existe um balanceamento perfeito, muito menos um modelo universal de um bom balanceamento. Em jogos com gameplay mais técnico, como RTS ou jogos com múltiplas unidades podemos até jogar inúmeras vezes e criar modelos e estatísticas (a ferramenta de gráfico no Machinations é bem útil aqui) para ver se há algum buraco

entre as unidades planejadas ou se algumas delas está com um custo/benefício fora do padrão, mas na maioria dos jogos o bom balanceamento ;e algo que está mais no *feel* do que em qualquer outra coisa. É só jogando inúmeras vezes e vendo novos jogadores aprendendo o mundo que você criou que conseguimos vislumbrar se os sistemas estão se movimentando conforme o planejado e se a experiência está sendo o que gostaríamos que fosse.

Resumindo

Os sistemas de um jogo são onde o difícil equilíbrio entre todos os elementos deve ser alcançado. Não é um trabalho fácil, mas com essas informações e testando algumas vezes você deve conseguir!

DOCUMENTAÇÃO DE GAME DESIGN

Agora que aprendemos o essencial a respeito de game design e seus elementos já temos tudo que precisamos para analisar e criar jogos. Mas saber como criar um jogo não nos garante conseguir conduzir o projeto de desenvolvimento até o final e terminarmos com um bom jogo. Nestas duas últimas unidades falar de assunto relacionados com o desenvolvimento do jogo em si. Na unidade de hoje, vamos falar sobre a documentação do design de um jogo, também conhecido como GDD.

GDD

Para que serve um GDD? A única e grande função dele é servir de guia de desenvolvimento do jogo para que todo mundo esteja na mesma página durante o desenvolvimento. Imagine um jogo como uma estante da Tok & Stock, o GDD é o manual que vem junto para te ensinar a montar. O GDD não é um livro sobre estantes, não é um compilado das melhores estantes do mundo, não é um tratado acadêmico da utilidade de estantes nem um livro narrativo sobre a vida de uma estante. Ele é o manual de montagem e só.

Como o manual de montagem, toda informação contida nele tem que ter uma aplicação direta e prática no jogo. Rever jogos

do mesmo gênero que o jogo a ser feito só é relevante se essa revisão impacta diretamente nas decisões do jogo. Contar a backstory do personagem secundário só pertence ao GDD se em algum momento do jogo ela é revelada ou relevante, e ainda assim ela deve estar presente da exata forma que vai aparecer no jogo (há exceções, trataremos caso a caso).

Isso não significa que esses conteúdos adicionais não devam existir ou ser organizados em algum documento, se eles são importantes para o jogo e para a equipe eles podem e devem ser feitos. O ponto é que mesmo existindo, o lugar deles não é no GDD.

Quando não documentar

Nem todo jogo a ser desenvolvido precisa de um GDD. Quando o tempo de desenvolvimento é curto e a equipe é pequena, como por exemplo em uma game jam, cada minuto que você gasta documentando é um minuto que perde desenvolvendo. Se você está trabalhando sozinho ou com poucas pessoas em um jogo que é um clone ou um experimento isolado de uma única mecânica, documentar o que deve ser feito parece e é redundante. Se o seu objetivo é fazer mais um clone de Flappy Bird só para testar seus conhecimentos em uma nova engine, para que se dar ao trabalho de documentar? A duração do projeto também é essencial nessa escolha. Se é um projeto de um ou dois dias que será realizado sem interrupções, talvez documentar realmente não seja uma necessidade.

Mas se o projeto envolver modificações ou outras atividades ficarão no meio do desenvolvimento, cuidado para não superestimar sua memória! Realmente não vale a pena documentar ao menos o funcionamento geral e principais features?

Documentar muito cedo coisas que ainda não estão completamente definidas também cria a tendência de solidificar um conceito ou feature antes da maturidade. Se alguém já escreveu 3 páginas sobre o sistema de level, por exemplo, todas as outras pessoas vão pensar duas vezes antes de propor um sistema novo, mesmo que melhor, já que a mudança envolveria jogar todo aquele trabalho no lixo. Faz parte da natureza humana, por mais que tentamos manter a mente aberta a mudanças, ninguém gosta de ver trabalho sendo jogado fora. Também faz parte da natureza humana ir aprimorando suas idéias com o tempo, aperfeiçoando a forma como as coisas são feitas o que em geral leva a um jogo melhor. Documentar cedo demais, coisas demais, barra ou atrapalha esse desenvolvimento natural.

Não documentar, por outro lado, tem consequências ainda piores, como informações importantes serem perdidas, esquecidas ou mal interpretadas, a mudança de escopo de projeto sem os próprios desenvolvedores perceberem ou ainda discussões desnecessárias aparecerem. O tempo passa, é normal resultados de discussões do começo do projeto não estarem mais tão frescas na cabeça e algumas informações já encontram-se misturadas com outras.

Logo, existe aí um difícil equilíbrio entre a quantidade de documentação necessária para cada projeto que não consuma mais tempo que o necessário e também permita que os membros da equipe trabalhem da melhor forma possível (mesmo que seja uma equipe de uma pessoa).

Tipos de GDD

Documentação de software é uma disciplina antiga e amplamente documentada (desculpem o trocadinho). Como o desenvolvimento de jogos empresta muitos dos processos do de software, a exemplo das metodologias de gerenciamento, a documentação foi uma das ferramentas que mais cedo foi adotada.

Quanto a extensão do documento, você e sua equipe tem algumas opções. A abordagem da "uma página" é ótima para jogos curtos com equipes pequenas e até em jams, mas não costuma funcionar tão bem em projetos um pouco maiores. Voltando à metáfora do manual de montagem, imagine uma estante retrátil, que vira uma mesa com bancos. O manual vai precisar de uma seção para cada uma das duas diferentes fases de montagem, uma seção para o uso cotidiano e manipulação da estante e talvez até uma seção para limpeza.

Em casos de equipes com mais de duas pessoas em projetos que vão durar mais de uma semana, é importante ter algo que contenha mais informações. E para ter mais informações, você

vai precisar de um pouco mais de estrutura, como a divisão por seções.

Não, isso ainda não quer dizer que toda e qualquer parte relacionada ao jogo deve ser escrita, mas sim que todas as partes relevantes precisam estar documentadas e organizadas por tema de forma que seja fácil para qualquer pessoa encontrar mesmo que nunca tenha mexido no GDD antes.

Gameplay, Backstory, Sistemas, Audio, UI Flow são algumas das possíveis seções que o documento irá se estruturar ao redor. Ou não. Cada jogo é único e, enquanto o jogo em si não está pronto para provar isso, as categorias do seu documento devem ser a maior expressão da sua unicidade. Pense o que é mais importante para seu jogo e comece listando essas áreas. Essas áreas vão ser seus capítulos, cada capítulo pode ter sub capítulos.

Por exemplo, um documento de Super Mario Bros. possivelmente teria um capítulo sobre 1) Movimentação, 2) Inimigos, 3) Elementos do level, 4) Levels, 5) Limitações de gameplay e 6) Feedback sonoro e visual. No primeiro capítulo, seria descrito a movimentação do Mario, no que ela era semelhante a outros jogos plataforma da sua época e no que ela era diferente. No segundo, cada um dos inimigos seria descrito e como eles funcionavam. No terceiro, o funcionamento das estruturas do mundo como os buracos, os tubos, as plataformas que podem ser atravessadas ou não, destruídas ou não e a bandeira de fim de fase. O capítulo de levels daria uma intro de quantos mundos e quantos levels por

mundo existem, mencionando as mudanças estéticas e de level design em cada um deles. O quinto capítulo falaria do sistema de vidas, de pontos e de tempo, ou seja, tudo que limita o jogador e adiciona dificuldade ao jogo que é externo ao level design. Por fim, o capítulo de feedback explicaria a música e efeito visual de pegar uma estrela, os efeitos de morte, o som de pegar uma moeda, a aceleração da música quando o tempo está acabando etc.

Esse documento inteiro poderia ter menos de dez páginas. Lembre-se, nesse caso não me importa a backstory ou os sonhos e motivações dos Goombas. Não me importa se ele é uma aberração genética criada no laboratório do Kamek e que se sente inseguro tendo em vista a quantidade de irmão idênticos ele tem. Me importa que ele se move da direita para a esquerda até colidir com algo e mudar de direção, que o jogador perde uma vida se Mario colidir lateralmente com ele e que ele morre se Mario pular em cima dele.

Projetos maiores, com mais membros na equipe e mais tempo, precisam de uma documentação bem mais detalhada e extensa, que nós informal e jocosamente chamamos de bíblia. A documentação aqui pode ser detalhada e completa, abordado cada aspecto de cada parte do jogo.

Devido ao tamanho destes projetos, seus documentos acabam se tornando Leviatãs gigantes que estão o tempo todo sendo construídos, editados e modificados. Desta forma, é preciso criar dentro da equipe regras sobre quando e como documentar qualquer conteúdo. Já falamos do problema de documentar

precocemente elementos que não estão certos ainda pois podem solidificar conceitos que ainda não precisam ser e isso acaba limitando a criatividade, mas esse problema é crítico quando uma das normativas é documentar todas as possibilidades, mesmo que ainda não totalmente testadas. Isso abre espaço para que outros membros da equipe visitem esse conteúdo enquanto ele ainda é Work in Progress (WIP) e, não percebendo isso, acabam por assumir que a versão inacabada é o conteúdo final. Então, sempre que estiver trabalhando em um projeto com GDD gigante, faça questão de deixar bem claro qual conteúdo ainda esta sendo trabalhado e qual já é definitivo e crie números para as versões do GDD. É importante que todos na equipe entendam isso e é sua obrigação ter certeza que todos dominem essas regras.

Tratamos aqui majoritariamente de jogos com foco em gameplay, mas se seu jogo tem foco em conteúdo, tem bastante narração e o desenvolvimento dos personagens é relevante ao gameplay em si do jogo, talvez esse seja seu caso. Backstory dos personagens não pertence ao GDD se não for relevante ao gameplay, mas se seu gameplay é baseado no storytelling a história deve estar no GDD sim. Mas não livremente como se fosse um livro feito em uma aula de escrita criativa! A informação narrativa do GDD deve estar organizada e hierarquizada em categorias para facilitar o acesso da equipe.

Mas pense bem antes de escolher ou acabar caindo nesse tipo de documentação. Lembre-se que cada hora gasta documentando é uma hora perdida desenvolvendo. É comum jovens que ainda estão pegando o jeito da coisa tentarem ser

sucintos e acabarem com uma bíblia. É mais comum ainda esse jogo documentado nunca ser feito e esses jovens partirem para a documentação-bíblia de outro jogo. Cuidado para não perder o escopo do projeto, da equipe e do tempo que você tem. Mas isso também é tema para próxima unidade

Linguagem técnica

Nossa linguagem pode parecer precisa e clara, mas na verdade ela permite muito espaço para diferentes interpretações se não tomarmos cuidado. Se eu te digo uma palavra simples como “carro”, o que vem na sua cabeça? Pense antes de continuar lendo.

Possivelmente, não foi um Jeep 76 amarelo, que foi o primeiro carro que eu tive e por isso é o primeiro que aparece, para você o primeiro carro a visitar sua mente será outro. Assim, se na minha documentação eu coloco que em determinado momento um carro aparece no jogo e o artista recebe essa documentação e faz, por exemplo, uma Ferrari, ele não está exatamente errado, não é mesmo? Ferrari é um carro, mesmo não sendo o carro que eu pensei quando escrevi o documento, é certamente um objeto que entra na categoria de carro. O erro não foi do artista, o erro foi meu por não usar linguagem técnica e específica sobre o que eu estava me referindo a.

Quando a gente fala "jogo multiplayer" parece que fica claro a quantidade de trabalho que vai estar envolvido. Mas não, essa terminologia geral esconde muitas particularidades que mudam

completamente a quantidade de trabalho envolvida, como por exemplo se assíncrono ou simultâneo, se é local ou online, qual o tamanho das equipes, co-op ou competitivo, o tipo de jogo, engine, servidor, etc. Esse mesmo conceito "multiplayer" pode significar a diferença entre habilitar as teclas setas e AWSD ao mesmo tempo para dois jogadores dividirem o mesmo teclado (algo relativamente simples na maior parte das engines) e ter 20 programadores back-end fazendo um banco de dados por seis meses. E a quantidade de trabalho de cada parte do jogo não está clara, como dividir igualmente entre todo mundo da equipe e estimar a entrega corretamente? Isso é essencial em jams, por exemplo.

Então sempre que estiver escrevendo algo, evite usar adjetivos desnecessários e se foque em descrever exatamente o que é esperado desse item, sem margem para erros ou interpretações. Para isso, idealmente você vai escrever algo e, logo em seguida, reler tentando encontrar outra forma de interpretar a exata mesma frase. Se você encontrou, como você pode reestruturá-la para evitar essa interpretação errada? Se não encontrou, leia de novo, você deve ter deixado algo passar. Se na terceira vez, realmente tudo parece ok, só aí vá para próxima frase. Essa precisão não deve se limitar a forma como você escreve sobre o jogo mas também onde estas descrições estão contidas, para facilitar que sejam encontradas e, por consequência, lidas. Um bom GDD tem linguagem clara e é estruturado de forma que mais faça sentido para a equipe que o utiliza.

Não existe exatamente um tipo de limitação de mídia e formato baseado em extensão e profundidade de GDD e uma

recomendação geral é faça o que funciona para sua equipe. Mas essa recomendação é tão geral que pode fazer todo sentido ou nenhum, dependendo da sua experiência e do seu time. Então vamos falar mais específico.

Independentemente do tamanho do GDD, uma regra geral que funcionou em todas as equipes que eu já estive é: todos entendem mecânicas muito melhor com um gif ou uma esquematização visual do que com qualquer número de parágrafos que você jogue neles. Ver o funcionamento em vez de ter que ler sobre ele é a diferença entre ver um animal, mesmo que só em um foto, e ler sobre ele.

Se você trabalha sozinho, um documento local, uma apresentação em ppt ou até um caderno ou folha de A3 podem ser uma documentação suficiente. Uma abordagem comum em jams é anotar coisas importantes em post-its e ir agrupando eles tematicamente, para ficar mais fácil entender a relação entre os elementos e tirar partes com mais facilidade.

Mas se você tem mais de uma pessoa na equipe, precisa pensar como todos os membros vão estar atualizados sobre a evolução e mudanças no documento. É por isso que o Google docs é tão popular entre desenvolvedores. Todos os membros da equipe, não importa o tamanho dela, tem acesso ao mesmo material e podem fazer alterações simultaneamente.

No GDD tipo bíblia é ainda mais importante a estrutura do texto e a forma do que em qualquer outro tipo. A quantidade de texto e material envolvido quase impossibilitam que você use um

documento linear, tipo Word. Suas alternativas ficam entre montar uma wiki com uma página para cada assunto ou usar alguma ferramenta colaborativa digital (como Google Docs) e carregar nos links entre seções.

Existem incontáveis formas de se fazer um GDD e nenhuma é certa e ou errada; tudo depende do escopo projeto e como a equipe se sente mais confortável. O GDD é uma ferramenta para ajudar no desenvolvimento do projeto e guiar sua equipe para a melhor versão de jogo possível. Se ele não estiver fazendo isso, está na hora de repensar essa relação.

PALAVRAS FINAIS

Com essas discussões, espero que você tenha uma visão mais clara do que game design diz respeito em um jogo.

Esse livro não vai te tornar num game designer da mesma forma que nenhum outro livro te tornaria. Conseguimos ir apenas até certo ponto utilizando somente palavras para descrever como pensar como um game designer. O resto do caminho depende totalmente de você de dos jogos que você faz e como aprende com eles. É apenas fazendo jogos, vendo pessoas testando, identificando melhorias possíveis e entendendo o comportamento humano que você vai se tornar um game designer.

Tendo dito isso, espero que esse livro tenha te mostrado que não há nada de mágico ou excepcional nessa carreira que seja inatingível para você. Todo mundo nasce um pouco game designer, criando as próprias regras para brincadeiras, o desafio está em entender esse processo e conseguir cultivá-lo de uma forma que crie experiências legais para mais pessoas além de você. Esse livro não te torna um game designer por que você já é um, só tem que praticar (e praticar muito) e eu espero que este livro tenha lhe dado alguma estrutura de por onde começar e como.

Não vou te dar uma carteirinha de game designer porque, surpresa, você já tinha uma! Dentro de você!

Se me permite uma última sugestão, comece fazendo pequenos projetos, testando pequenos conceitos e vendo pessoas jogando esses projetos. Aprendemos mais começando, desenvolvendo, terminando e (principalmente) lançando dez pequenos jogos de um mês de desenvolvimento do que em um único projeto de um ano inteiro. Claro, há certas coisas que apenas projetos mais longos vão te ensinar, mas a princípio comece com essa dica de min/max e *faça uns joguinho* curtos.

REFERÊNCIAS

Adams, Ernest. The Designer's Notebook: Difficulty Modes and Dynamic Difficulty Adjustment. Gamasutra, 14 mai. 2008. Disponível em: http://www.gamasutra.com/view/feature/3660/the_designers_notebook_.php . Acesso em: 26 mar. 2021.

Adams, Ernest; Rollings, Andrew. **Andrew Rollings and Ernest Adams on Game Design**. Berkeley: New Riders, 2003.

Bartle, Richard. **Early MUD history**. Living Internet, 15 nov. 1990. Disponível em: http://www.livinginternet.com/d/di_major.htm. Acesso em: 15 set. 2009.

Brathwaite, Brenda. Schreiber, Ian. **Challenges for game designers**. Boston: Course Technology, 2009.

Caillois, Roger. **Los juegos y los hombres**: la máscara y el vértigo. Ciudad del Mexico: Instituto Armando Reveron, 1986.

Chen, Jenova. **Flow in games**. 2008. 20 f. Dissertação (Mestrado em mídias interativas) – Interactive Media Division, University of Southern California, Los Angeles, 2008.

Csikszentmihályi, Mihaly. **Flow**. New York: HarperCollins, 2008.

Fullerton, Tracy. **Game design workshop**: a playcentric approach to creating innovative games. Burlington: Elsevier Morgan Kaufmann, 2008.

Hunicke, Robin; LeBlanc, Marc; Zubek, Robert. MDA: A formal approach to game design and game research. In: **Proceedings of the challenges in game AI Workshop**, Nineteenth National Conference on Artificial Intelligence (AAAI '04), 19, 2004. San Jose. Menlo Park: AAAI Press, 2004.

Huizinga, Johan. **Homo ludens**. São Paulo: Perspectiva, 2000.

Koster, Raph. **A theory of fun**. Scottsdale: Paraglyph, 2005.

Juul, Jesper. Games telling stories? A brief note on games and narratives. **Game Studies**, v.1, n. 1, 2002.

Juul, Jesper. The game, the player, the world: looking for a heart of gameness. In: COPIER, Marinka; RAESSENS, Joost (Eds.). **Level Up: Digital Games Research Conference Proceedings**. Utrecht: Utrecht University, 30-45. 2003.

Rodenberg, Cassie. Grand Theft Auto V makes it cool to pick up – even kill – prostitutes. **The Guardian**, 27 dez 2013. Disponível em: <https://www.theguardian.com>

Schell, Jesse. **The art of game design: a book of lenses**. Burlington: Elsevier Morgan Kaufmann, 2008.

Suits, Bernard. **The grasshopper: Games, life and utopia**. Peterborough: Broadview Press, 2005.

Sutton-Smith, Brian. Play and ambiguity. *In*: Salen, Katie. Zimmerman, Eric. **The game designer reader: a rules of play anthology.** Cambridge: MIT Press, 2005.

Weiller, Thais. Mine, yours and ours: how narrative structures may influence User Generated Content. *In*: Almeida, Cristiba Miranda de; Ciasrellardi, Matteo; Scolari, Carlos A. (Orgs.). **Understanding media, today.** Barcelona: Book Printing S.L., 2011, v. 01, p. 319-328.

Weiller, Thais. **Game Start:** Lições de Game Design para seu Videogame. Curitiba: Game Start, 2015.

Weiller, Thais. **Pense Pequeno:** e termine seu primeiro videogame. Curitiba: Game Start, 2017.

Weiller, Thais. Teaching People What They Already Know: Designing Game Design Courses. **Gamevironments**, Bremen, n. 15, p.380-392, dez., 2021.

WOLF, Mark J. P. (Org.). **The video game explosion:** a history from PONG to playstation and beyond. Westport: Greenwood, 2007.