

Fabrcio Henrique Roepcke

**LEVEL DESIGN E SISTEMA DE CASAS  
MODULARES PARA O JOGO CRYSTAL LEAGUE**

Linha de Pesquisa: Projeto de Animação  
Temática: Design de Animação; Design de jogos;  
Desenvolvimento 3D.

Projeto de Conclusão de Curso (PCC)  
submetido ao Programa de  
Graduação da Universidade Federal  
de Santa Catarina para a obtenção do  
Grau de Bacharel em Design.  
Orientador: Prof. Ms. Flávio Andaló

Florianópolis  
2018

Fabrcio Henrique Roepcke

**LEVEL DESIGN E SISTEMA DE CASAS  
MODULARES PARA O JOGO CRYSTAL LEAGUE**

Este Projeto de Concluso de Curso foi julgado adequado para obteno do Ttulo de Bacharel em Design Grfico, e aprovado em sua forma final pelo Curso de Design da Universidade Federal de Santa Catarina.

Florianópolis, 21 de Novembro de 2018

---

Professora Dr<sup>a</sup>. Marlia Matos Goncalves.  
Coordenadora do Curso

**Banca Examinadora:**

---

Prof. Flávio Andaló, Ms.  
Orientador  
Universidade Federal de Santa Catarina

---

Prof. Gustavo Boehs, Ms  
Universidade Federal de Santa Catarina

---

Prof<sup>a</sup>. Monica Stein, Dra.  
Universidade Federal de Santa Catarina

---

Prof<sup>a</sup>. Rafael Arrivabene, Ms.  
Universidade do Vale do Itajaí

## RESUMO

Este trabalho tem como propósito o estudo prático da atividade de *level design*, buscando conteúdos teóricos e práticos sobre processos de desenvolvimento de jogos 3D. Como estudo prático este projeto desenvolverá um cenário 3D jogável para o jogo *Crystal League*, aplicando os conceitos e processos estudados junto com a equipe de desenvolvimento do jogo.

**Palavras-chave:** *Level Design* 1. Modular 2. Iteração 3.



## **ABSTRACT**

This work has the purpose of the practical study of the level design activity, gathering theoretical and practical content about the process on the development of 3D games. The practical part of the project consists in applying the concepts and processes together with the development team of the game Crystal League.

**Keywords:** Level Design 1. Modular 2. Iteration 3.



## LISTA DE FIGURAS

Figura 1 - Cronograma de Tarefas seguindo o PreProduction Blueprint .....	15
Figura 2 - Peças modulares .....	16
Figura 3 - Unidade de tamanho de peças modulares.....	17
Figura 4 - Kit temático de peças modulares.....	18
Figura 5 - Processo de desenvolvimento usando iteração	19
Figura 6 - Pirâmide de Desenvolvimento .....	19
Figura 7 - Processo de desenvolvimento de fases para um jogo .....	20
Figura 8 - Gráfico das Chaves para a Diversão.....	24
Figura 9 - Taxonomia de Richard Bartle .....	25
Figura 10 - Atividade de Game Design .....	27
Figura 11 - Atividade de Level Design .....	27
Figura 12 - Manual do jogo The Legend of Zelda: Ocarina of Time.....	28
Figura 13 - Como Batman: Arkham Asylum ensina o jogador .....	29
Figura 14 - Início do jogo Megaman X.....	30
Figura 15 - Como Megaman X ensina o jogador.....	31
Figura 16 - Como Assassin's Creed: Origins aborda objetivos.....	32
Figura 17 - Como Uncharted 3 aborda objetivos .....	33
Figura 18 - Como as formas afetam o jogador.....	34
Figura 19 - Como Weenies guiam jogador .....	35
Figura 20 - Como a iluminação guia o jogador.....	36
Figura 21 - Como formas e linhas guiam o jogador.....	37
Figura 22 - Variações de um inimigo.....	39
Figura 23 - Verticalidade em Overwatch.....	40
Figura 24 - Exemplo de caminho linear .....	41
Figura 25 - Exemplo de caminho em grid.....	41
Figura 26 - Exemplo de caminho em teia .....	42
Figura 27 - Exemplo de caminho em pontos do espaço...	42
Figura 28 - Exemplo de caminho dividido em espaços ...	43
Figura 29 - Tipos de Organização Espacial .....	46
Figura 30 - Exemplo de jogo para apenas um jogador.....	47

Figura 31 - Exemplo de jogo para vários jogadores .....	48
Figura 32 - Exemplo de jogo para um e dois jogadores...	48
Figura 33 - Imagem de campanha para campeonato de World of Warcraft.....	49
Figura 34 - Modos de jogo Multiplayer .....	50
Figura 35 - Relação de posicionamento e comportamento do jogador .....	51
Figura 36 - Distância de combate entre jogadores .....	52
Figura 37 - Tipos de proteção .....	53
Figura 38 - Simetria e Assimetria de fases .....	54
Figura 39 - Balanceamento de tempo e distância.....	56
Figura 40 - Respostas da pesquisa sobre Tamanho dos Cenários .....	61
Figura 41 - Resposta da pesquisa sobre Tempo de Partida .....	62
Figura 42 - Jogo Surviv.io Battle Royale.....	62
Figura 43 - Resposta da pesquisa sobre Meios de Locomoção.....	63
Figura 44 - Resposta da pesquisa sobre Formato de Equipe .....	64
Figura 45 - Resposta da pesquisa sobre Fluxo na Partida	65
Figura 46 - Teste de Processamento .....	66
Figura 47 - Top Down Layout do Cenário.....	73
Figura 48 - Top Down Layout partes específicas .....	74
Figura 49 - Definição Visual 1.....	75
Figura 50 - Definição Visual 2.....	76
Figura 51 - Visão de Cima do Cenário em sua fase de Blocagem .....	77
Figura 52 - Composição Visual para Guiar o jogador aos Pontos Focais .....	77
Figura 53 - Níveis de Hierarquia do Sistema Modular ....	79
Figura 54 - Conceito Peças Modulares .....	80
Figura 55 - Primeira versão das peças 3D modulares .....	81
Figura 56 - Funcionamento das peças modulares de acordo com Pivô, Grid, Escala e hierarquia.....	81
Figura 57 - Primeiro modelo de casa modular montada ..	82
Figura 58 - Plantas Baixas das casas modulares .....	83

Figura 59 - Casas modulares montadas.....	83
Figura 60 - Casas modulares já detalhadas .....	84
Figura 61 - Textura de uma das peças modulares .....	85
Figura 62 - Modelos 3D no programa 3DS Max .....	86
Figura 63 - Modelos 3D no programa Unity 3D já com textura e material .....	86
Figura 64 - Exemplo de variação possível nas Casas modulares.....	87
Figura 65 - Estado do cenário para o Playtesting 2.....	89
Figura 66 - Estado do cenário antes do Playtesting 3 .....	91
Figura 67 - Funcionalidade principal para o gênero Battle Royale .....	93
Figura 68 - Interior do cenário .....	94
Figura 69 - Visão geral do cenário em perspectiva.....	94
Figura 70 - Interior do cenário e elementos .....	95
Figura 71 - Interior do Cenário e composição visual.....	95
Figura 72 - Affordance de Rampear .....	96
Figura 73 - Organização Espacial Linear.....	96
Figura 74 - Organização Espacial em Malha .....	97
Figura 75 - Composição Visual e Ponto Focal .....	97

## **LISTA DE ABREVIATURAS E SIGLAS**

**HUD** – *Heads Up Display*

**3D** – Três dimensões

**FPS** – *First Person Shooter*

**HUB** – Termo inglês que define ambiente central que conecta vários outros

**IA** – Inteligência Artificial

## SUMÁRIO

1 INTRODUÇÃO .....	10
1.1 OBJETIVOS .....	11
<b>1.1.1 Objetivo Geral</b> .....	11
<b>1.1.2 Objetivos Específicos</b> .....	11
1.2 JUSTIFICATIVA .....	12
1.3 metodologia projetual acadêmica.....	12
1.4 METODOLOGIA PREPRODUCTION BLUEPRINT .....	13
1.5 METODOLOGIA DE PRODUÇÃO MODULAR....	15
1.6 METODOLOGIA DE PRODUÇÃO ITERATIVA... 18	
<b>2 PROBLEMATIZAÇÃO</b> .....	21
3 COLETA DE DADOS E ANÁLISE .....	22
3.1 O QUE É UM JOGO?.....	22
3.2 QUATRO CHAVES PARA A DIVERSÃO .....	23
3.3 QUATRO TIPOS DE JOGADOR.....	24
3.4 GAME DESIGN VS LEVEL DESIGN.....	26
3.5 MECANISMOS DE ENSINO .....	28
3.6 AFFORDANCES.....	31
3.7 LEVEL DESIGN COM A DISNEY .....	34
3.8 ESPAÇOS.....	38
3.9 ARQUITETURA E ORGANIZAÇÃO ESPACIAL .	44
3.10 SINGLEPLAYER VS MULTIPLAYER.....	47
3.11 MULTIPLAYER LEVEL DESIGN .....	49
3.12 BALANCEAMENTO.....	55
4 SÍNTESE .....	57
4.1 IDEIA.....	57
4.2 CONTEXTO DO JOGO CRYSTAL LEAGUE .....	57
4.3 DEFINIÇÕES DE GAME DESIGN.....	58
4.4 MODO DE JOGO BATTLE ROYALE .....	59
4.5 DIREÇÃO DE ARTE.....	59
5 DESENVOLVIMENTO .....	60
5.1 PESQUISA com jogos referência .....	60
5.2 TESTES DE PROCESSAMENTO.....	66
5.3 PREPRODUCTION BLUEPRINT.....	67

<b>5.3.1 Ideia</b> .....	68
<b>5.3.2 Tema</b> .....	68
<b>5.3.3 Propósito do Projeto</b> .....	68
<b>5.3.4 Características Diferenciais</b> .....	69
<b>5.3.5 Referência Visual</b> .....	70
<b>5.3.6 História</b> .....	70
<b>5.3.7 Objetivos e Obstáculos</b> .....	71
<b>5.3.8 Ponto Focal</b> .....	71
<b>5.3.9 Top Down Layout</b> .....	72
<b>5.3.10 Definição Visual</b> .....	75
<b>5.4 BLOCAGEM</b> .....	76
<b>5.5 ASSETS DO PROJETO CRYSTAL LEAGUE</b> .....	78
<b>5.6 PLANEJAMENTO MODULAR</b> .....	78
<b>5.7 PLANEJAMENTO DE ITERAÇÃO E TESTES</b> .....	87
<b>5.7.1 Playtesting 1 e Iteração</b> .....	88
<b>5.7.2 Playtesting 2 e Iteração</b> .....	89
<b>5.7.3 Playtesting 3 e Feedback</b> .....	90
<b>5.8 EXEMPLOS DO RESULTADO DO CENÁRIO</b> .....	93
<b>6 COMUNICAÇÃO</b> .....	98
<b>6.1 COMUNICAÇÃO E APRESENTAÇÃO</b> .....	98
<b>6.2 CONCLUSÃO E RESULTADO</b> .....	98
<b>7 REFERÊNCIAS</b> .....	99



## 1 INTRODUÇÃO

A maioria dos jogos atuais tem como experiência principal para o usuário a exploração de um cenário. Por mais que as outras mecânicas e sistemas do jogo ajudem nesse processo, o cenário é onde a maioria das atividades podem ocorrer.

Por muito tempo a atividade de *level design*, que é a prática de desenvolver cenários de jogos pensando na experiência do jogador, foi exercida em conjunto com a de *game design*, hoje isso está mudando, as características específicas de cada um dos dois estão se destacando e se aprofundando, o que tende a promover um conteúdo melhor ao usuário.

Na questão de desenvolvimento 3D, é fácil verificar a complexidade de um projeto, principalmente de um jogo. Existem inúmeras variáveis onde as responsabilidades da equipe são divididas, e a do *level designer* se define pelo planejamento dos cenários do jogo e tudo que vai dentro deles, fazendo uma ligação entre a parte técnica e artística de um projeto.

Um dos maiores problemas no desenvolvimento de jogos, principalmente no panorama independente, é que algumas equipes tendem a seguir uma metodologia linear, de desenvolver a ideia do jogo, produzir o jogo e publicar o jogo. Outras equipes pioneiras trabalham com a questão de iteração, testando o funcionamento de todas as atividades do jogo desde a primeira versão jogável, que deve ser feita o quanto antes no processo. Isso, no fim, tende a resultar em um produto de melhor qualidade. Esses aspectos entram no escopo deste projeto, que visa trabalhar com uma metodologia modular para agilidade de produção e reutilização de *assets* e também uma metodologia de iteração, para analisar feedback e promover melhorias na experiência do jogo.

Para o estudo dos conceitos e desenvolvimento deste projeto de conclusão de curso com feita uma parceria para

o jogo *Crystal League* onde o *Level Design* para um de seus cenários será trabalhado. O jogo *Crystal League* é um jogo para múltiplos jogadores com vários personagens e modos de jogo diferentes.

Trabalhando dentro do universo do jogo *Crystal League*, este projeto vai passar por um processo de produção com uma equipe pequena, mas abstraído ao máximo o processo de grandes empresas pioneiras para o cenário do jogo, buscando otimizar tempo de desenvolvimento e criar um produto de boa qualidade.

## 1.1 OBJETIVOS

### 1.1.1 Objetivo Geral

Este projeto de conclusão de curso tem como objetivo desenvolver um cenário jogável com elementos modulares para um jogo digital seguindo conceitos de *Level Design* e utilizando técnicas de planejamento de produção 3D.

### 1.1.2 Objetivos Específicos

- Pesquisar, estudar e analisar conceitos de *level design*;
- Pesquisar conteúdo de planejamento e produção modular;
- Aplicar os conceitos de *level design* dentro do contexto do jogo *Crystal League*;
- Aplicar o conteúdo de planejamento e produção modular dentro do contexto do jogo *Crystal League*;
- Criar um planejamento modular para a produção dos modelos 3D;
- Criar um cenário funcional para ser aplicado dentro do jogo *Crystal League*

## 1.2 JUSTIFICATIVA

O cenário é onde o jogador vai praticar a maioria das ações do jogo, é um elemento de extrema importância no produto e necessita dos devidos estudos e planejamentos para sua criação.

Tanto jogos de tabuleiro quanto digitais possuem cenários que compõem o mundo do jogo, e esses cenários ajudam a proporcionar a experiência ao jogador. É onde o jogador habita.

A criação de um cenário de jogo consiste em mais do que apenas beleza artística, mas muitos outros conceitos que devem definir uma usabilidade clara e concisa. Pensando na questão de usabilidade e uma experiência de usuário lúdica, procuramos e definimos os melhores modos de criar o cenário do jogo.

Pensando em uma perspectiva de produção do jogo e também na experiência final do usuário jogador, existem métodos que facilitam o desenvolvimento do jogo para ambos os lados. Na questão de produção 3D temos a questão da modularidade e seus benefícios, e na questão de experiência temos a questão de iteração a partir de testes e análise de feedback.

No desenvolvimento do jogo *Crystal League* o *Level Design* é um dos pontos mais fortes. É de extrema importância entender tanto os processos de produção de um nível quanto os conceitos de design que criam a experiência do jogo.

## 1.3 METODOLOGIA PROJÉTUAL ACADÊMICA

Com a já citada relevância do *Level Design* para os jogos eletrônicos, a produção de um cenário modular para um jogo digital necessita de um forte planejamento, tanto conceitual quanto de produção.

Para o desenvolvimento deste projeto de conclusão de curso dentro do meio acadêmico, a metodologia escolhida para este projeto foi desenvolvida por Leonard Bruce Archer (1922 – 2005) e se chama *Systematic Method for Designers*. Ela define um processo dividido nas seguintes etapas:

- Problematização – Definir os objetivos do projeto e quais conteúdos serão necessários para seu desenvolvimento;
- Coleta de dados – Buscar e classificar os conteúdos que foram definidos como necessários. Coletar e armazenar informações;
- Análise – Estudar os materiais obtidos e analisar os conteúdos relevantes a serem abordados;
- Síntese – Aplicar o que foi estudado dentro do contexto do jogo *Crystal League* e definir os métodos e conceitos da pré-produção, junto com contexto do planejamento modular e de iteração do desenvolvimento do projeto;
- Desenvolvimento – Fazer a pré-produção do nível, planejamento modular e te iteração, criar as peças modulares, montar o cenário, refinar a qualidade dele através dos processos de iteração e posicionar os elementos de interação do jogo *Crystal League*;
- Comunicação – Apresentação do cenário dentro do jogo *Crystal League*.

#### 1.4 METODOLOGIA PREPRODUCTION BLUEPRINT

O *Preproduction Blueprint* foi desenvolvido por Alex Galuzi (2016) e é um guia para o planejamento e pré-produção de cenários 3D para jogos. O guia mostra um processo que vai desde as ideias do projeto até o desenvolvimento visual do cenário, levando em conta história, propósito do projeto, *game design* e *level design* e também fala um pouco sobre processos técnicos.

Dentro deste projeto vamos seguir algumas partes do *Preproduction Blueprint*, que serão:

1. IDEIA – Desenvolvimento de ideias para o projeto, tanto pontos conceituais e jogabilidade e *level design* quanto visual. O livro explica o quão importante é perceber os ambientes do mundo real e como abstrair para o mundo do jogo;
2. TEMA – Processo de criação do tema do cenário que irá ser criado, levando em conta ambientação e localização. É um processo conceitual onde transformamos a ideia de cenário em um local e suas características;
3. PROPÓSITO DO PROJETO – Muito importante para a motivação no desenvolvimento do projeto, essa parte se define o porquê da criação desse projeto, tanto em termos pessoas quanto profissionais;
4. CARACTERÍSTICAS DIFERENCIAIS – Criação e definição dos diferenciais do projeto, o que vai fazer o nível ser diferente dos outros e o que ele vai proporcionar de bom;
5. REFERÊNCIAS – Busca de referências visuais para deixar os conceitos concisos;
6. HISTÓRIA – Desenvolvimento da história do local, como os jogadores interagem com ela como o cenário foi se desenvolvendo dentro do seu mundo;
7. OBJETIVOS E OBSTÁCULOS – Definição dos objetivos que os jogadores precisam conquistar, relacionando com seus obstáculos;
8. PONTO FOCAL – Criação de pontos de referência no cenário, definindo a arquitetura e caminhos que o jogador vai percorrer;
9. *TOP-DOWN LAYOUT* – É a planta baixa do cenário, depois de ter todos os conceitos concisos é criado a primeira versão do *level design*;

## 10. DEFINIÇÃO VISUAL – Definições de arte e técnicas para o desenvolvimento 3D do cenário.

*Figura 1 - Cronograma de Tarefas seguindo o PreProduction Blueprint*

Preproduction Blueprint			
STAGE	TASK	STATUS	COMMENTS
IDEA	IDEA - Describe the level idea in one single sentence	TO DO	-
IDEA	IDEA - Environment Setting for the level	TO DO	-
IDEA	IDEA - Narrow down the Environment Setting to a location (specify)	TO DO	-
IDEA	IDEA - Define the Theme for the Environment (description, style and feel)	TO DO	-
PURPOSE	PURPOSE - What is the reason I am creating this environment?	TO DO	-
DESIGN	DESIGN - Features to define the elements of the Level	TO DO	-
REFERENCE	REFERENCE - Architecture - QUESTIONS	TO DO	-
REFERENCE	REFERENCE - Environment - QUESTIONS	TO DO	-
REFERENCE	REFERENCE - Lighting - QUESTIONS	TO DO	-
REFERENCE	REFERENCE - Props - QUESTIONS	TO DO	-
REFERENCE	REFERENCE - Inspiration and Style - QUESTIONS	TO DO	-
REFERENCE	REFERENCE - Answer the References - QUESTIONS	TO DO	-
STORY	STORY - What is the overall story of the environment?	TO DO	-
STORY	STORY - Description of this location from point of view of a character that lives there	TO DO	-
STORY	STORY - Why are we in this location?	TO DO	-
STORY	STORY - Show, don't tell	TO DO	-
STORY	STORY - Create a story of the player within the location as they navigate through it	TO DO	-
OBJECTIVE	OBJECTIVE - Define a set of objectives for the level	TO DO	-
OBJECTIVE	OBJECTIVE - Go through the objectives and story and see if it pushes forward	TO DO	-
OBJECTIVE	OBJECTIVE - Define obstacles, for each objective, that the player has to overcome	TO DO	-
OBJECTIVE	OBJECTIVE - Define directed set pieces/scripted events, for each objective and obstacles	TO DO	-
OBJECTIVE	OBJECTIVE - Define indirect set pieces/scripted events	TO DO	-
OBJECTIVE	OBJECTIVE - Define environmental set pieces/scripted events to add atmosphere	TO DO	-
OBJECTIVE	OBJECTIVE - Objective Progression Chart for the level	TO DO	-
FOCAL POINT	FOCAL POINT - Go through all that has been done and extract/create a specific Landmarks to be the focal point	TO DO	-
FOCAL POINT	FOCAL POINT - Choose the focal point(s) within the environment - REFERENCE IMAGES	TO DO	-
TOP DOWN LAYOUT	TOP DOWN LAYOUT - Define the type of the level	TO DO	-
TOP DOWN LAYOUT	TOP DOWN LAYOUT - Create the Top Down Layout - 2D	TO DO	-
TOP DOWN LAYOUT	TOP DOWN LAYOUT - Create the Visual Progression Guide from the 2D top down layout	TO DO	-
TOP DOWN LAYOUT	TOP DOWN LAYOUT - Define iconics for the elements on the Top Down Layout	TO DO	-
TOP DOWN LAYOUT	TOP DOWN LAYOUT - Create the 3D Level assets of the Top Down Layout	TO DO	-
VISUAL	VISUAL - Define the style of the environment	TO DO	-

Fonte: Autor

Seguindo todos os passos o cenário do jogo vai criando vida e essa documentação se torna muito importante para a centralização de todos esses conceitos além de ser um grande motivador de desenvolvimento.

Após a parte de definição visual o projeto tomará um caminho mais próprio, seguindo o planejamento de iteração que está proposto. A figura 1 mostra uma planilha de todos os passos da metodologia.

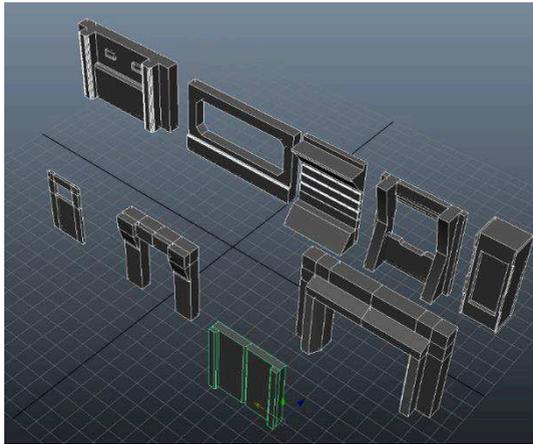
### 1.5 METODOLOGIA DE PRODUÇÃO MODULAR

O conceito de design modular já é explorador por empresas e funciona para reaproveitar peças e acelerar a produção. Isso pode parecer que o cenário pode ficar repetitivo visualmente, mas não necessariamente.

Ao trabalhar com design modular é necessário um planejamento, e sua profundidade depende do escopo do projeto. Algumas empresas usam para criar todos os cenários de um jogo, no caso deste projeto o intuito é trabalhar com modularidade dentro de um cenário do jogo *Crystal League*, mas um cenário relativamente grande.

A técnica permite que possamos reutilizar peças para a montagem do cenário, e seu planejamento funciona para que tudo encaixe perfeitamente, como mostra a figura 2. Dependendo da equipe e do projeto o processo pode ser aperfeiçoado de maneiras diferentes, mas existem pontos-chaves que precisam ser levados em conta para que o planejamento funcione.

*Figura 2 - Peças modulares*



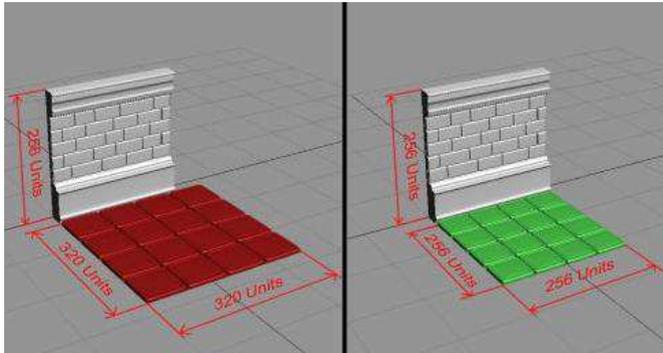
Fonte: <http://hu-james-et15.blogspot.com>

Antes de entrar na parte conceitual da criação das peças modulares é preciso identificar a métrica que vai ser trabalhada, tanto dos personagens quanto das peças a serem criadas. Muitos desenvolvedores usam a potência de dois e seus múltiplos como métrica (figura 3), isso porque fica muito mais claro como o trabalho está sendo feito e a

conexão das peças funciona quase que naturalmente criando detalhes no cenário.

Existe mais um ponto crucial para o processo além das definições de métrica, que é o Grid. O grid precisa auxiliar a conexão das peças e o seu não uso (ou mau uso) pode acarretar em erros de tamanho e junção.

*Figura 3 - Unidade de tamanho de peças modulares*



Fonte: Gamasutra, Creating Modular Game Art for Fast Level Design

Quando as definições de métrica e grid estiverem encaminhas e ajustadas no projeto, entra-se na fase conceitual. Nessa fase são definidos os conceitos técnicos e visuais do jogo ou cenário. Como exemplo, um jogo com uma fase de neve, uma de deserto, uma de floresta e uma de cidade. Com isso são criados os conceitos dos *kits*. *Kits* são conjuntos das peças visuais que cada fase vai precisar (exemplo na figura 4), algumas fases precisam de peças específicas e os *kits* suportam isso, mas também podem ser gerais, servindo tanto em uma fase quanto em outra.

As peças que suportam um *kit* que vão ser criadas seguindo a métrica e posicionadas de acordo com o grid do projeto.

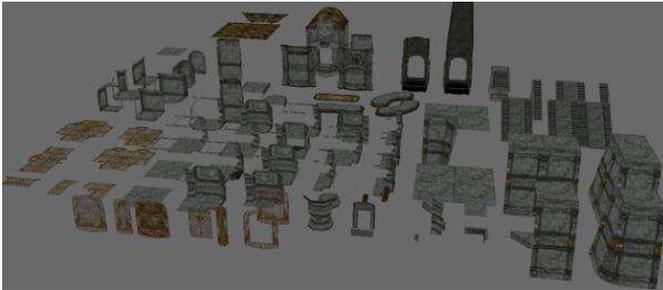
Quando os *kit* estão definidos o *level designer* começa então a criar e testar as peças do kit, isso para entender como o cenário deve ser montado e quais as

principais peças. As peças dentro dos *kits* podem ser definidas por nível de hierarquia, dependendo da quantidade de uso e do tempo de aparição no jogo.

Nessa parte de teste entra também a questão de nomenclatura de cada peça, como o *level designer* não trabalha sozinho, mas sim com uma equipe, as peças devem estar muito bem especificadas para o entendimento do projeto.

Assim que o *level designer* deixa definido as especificações das peças, passando pelos teste, a equipe de arte começa a trabalhar em cima delas e então o processo de teste e iteração começa a acontecer. Antes mesmo da arte do jogo estar pronta. Nesse processo os problemas vão sendo encontrados e resolvidos para um próximo teste.

*Figura 4 - Kit temático de peças modulares*



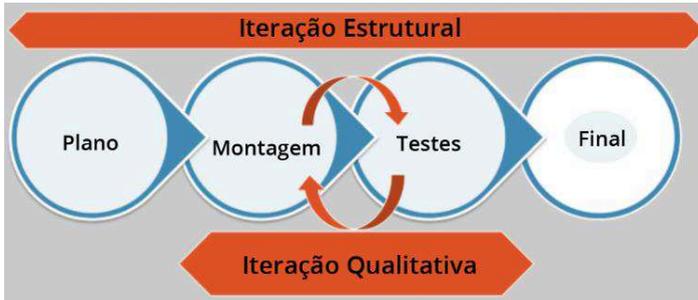
Fonte: Gamasutra, Skyrim's Modular Approach to Level Design

## 1.6 METODOLOGIA DE PRODUÇÃO ITERATIVA

O conceito de iteração dentro do desenvolvimento de jogos consiste em planejar, criar e testar o conteúdo até que ele esteja polido e pronto para publicação. É um ciclo de repetição, buscando problemas a serem resolvidos e aprimorando o conteúdo;

O processo (figura 5) segue até o fim com uma iteração qualitativa, ou seja, melhorando o conteúdo que está sendo criado.

Figura 5 - Processo de desenvolvimento usando iteração



Fonte: blog.joelburgess.com - Adaptado

A pirâmide de desenvolvimento do projeto pode ser vista da forma como Joel Burgess (2014) mostra, na figura 6, evidenciando os testes e sua importância para a qualidade do produto final.

Figura 6 - Pirâmide de Desenvolvimento

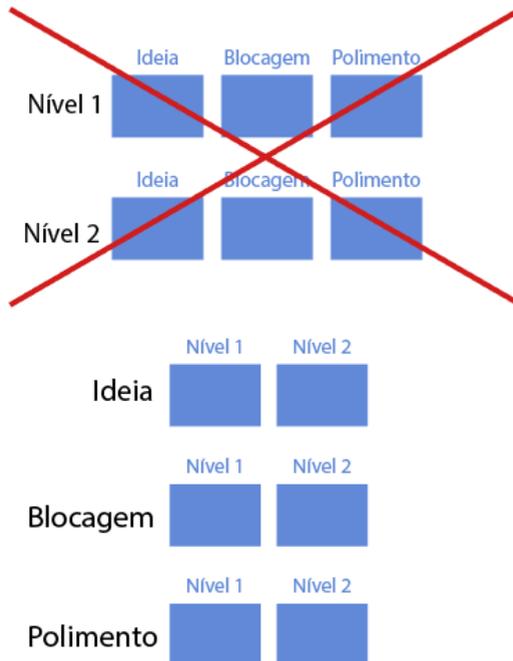


Fonte: Fonte: blog.joelburgess.com – Adaptado

Entrando na questão de *level design*, tudo segue o mesmo caminho, onde as diferenças ficam na parte estratégica de cada equipe. A maneira que Joel Burgess

(2014) identifica como ideal para sua equipe, como fala na sua palestra para a GDC, *The Iterative Level Design Process Used to Ship Fallout 3 and Skyrim, 2014*, é usando as fases de desenvolvimento e iteração fase por fase, uma após a outra enquanto elas estão dentro do mesmo contexto de produção, como na figura 7. Por exemplo, em um jogo com quatro fases, cada fase teria primeiramente a criação de sua blocagem, uma por uma, e só depois de todas terem sua blocagem pronta a equipe passaria para outra etapa de desenvolvimento, como polimento por exemplo.

Figura 7 - Processo de desenvolvimento de fases para um jogo



Fonte: Autor

## 2 PROBLEMATIZAÇÃO

Desenvolvendo primeiramente a base teórica do assunto, usaremos conceitos com relevância no panorama atual para buscar conteúdo tanto de referência quanto de produção. O desafio do projeto entra tanto na parte conceitual de criação do cenário funcional quanto a produção de suas peças modulares.

O cenário de um jogo digital precisa ser muito bem estudado até chegar na parte de produção. Esse estudo é normalmente feito pelo *game designer* ou pelo *level designer*, que são responsáveis pela experiência que o jogador vai ter enquanto explora o cenário. Para auxiliar o *level designer* nesse processo existem definições, conceitos e estudos que quando aplicados possuem um grande peso na experiência do jogo. Este projeto vai buscar esse conteúdo relevante e usá-lo em seu desenvolvimento.

A proposta do projeto é fazer um cenário com elementos modulares, o que nos leva a necessitar de um planejamento modular para definir as variáveis da produção, como delimitação de tamanho das peças e como será o funcionamento de encaixe delas. Esse planejamento leva em conta todas as definições de como as peças 3D precisam ser feitas e como elas irão funcionar.

O que fará o desenvolvimento conceitual e o planejamento modular andarem juntos é o processo de iteração. Esse processo tem como objetivo realizar testes de funcionalidade do jogo e identificar problemas o quanto antes. Esses problemas são então resolvidos para uma próxima versão. O processo consiste em arrumar erros e adicionar elementos a cada nova versão, até o prazo final de entrega, que neste caso será a data para entrega do Projeto de Conclusão de Curso 2.

### 3 COLETA DE DADOS E ANÁLISE

#### 3.1 O QUE É UM JOGO?

Existem várias definições diferentes sobre o que é um jogo, e isso depende de como tal autor se encaixa na área e seus tópicos de estudo. Levando isso em consideração, existem pontos mais importantes do que a definição em si do que é um jogo, que são os conteúdos que os jogos possuem e como esses conteúdos conversam com o jogador. Jesse Schell, em seu livro *The Art of Game Design: A Book of Lenses* (2010), traz algumas qualidades chave que são importantes para os jogos.

- Q1. Jogos são entrados intencionalmente;
- Q2. Jogos possuem objetivos;
- Q3. Jogos possuem conflito;
- Q4. Jogos possuem regras;
- Q5. Jogos podem ser ganhados ou perdidos;
- Q6. Jogos são interativos;
- Q7. Jogos possuem desafios;
- Q8. Jogos podem criar seu próprio valor interno;
- Q9. Jogos engajam seus usuários;
- Q10. Jogos são sistemas formais e fechados.

Usando essas qualidades como conteúdo que um jogo precisa ter, Schell define um jogo com suas palavras, segundo os dez pontos que foram citados acima; “*Um jogo é uma atividade de solução de problemas abordada com uma atitude de brincadeira*” (Schell, 2010).

Para auxiliar no desenvolvimento de um jogo e na atividade de *game design* podemos usar essas qualidades como um compasso.

### 3.2 QUATRO CHAVES PARA A DIVERSÃO

Para entender melhor como os usuários se sentem em relação aos jogos, Nicole Lazzaro (2004) em seu artigo *Why We Play Games: Four Keys to More Emotion in Player Experiences*, definiu quatro tipos de emoções que os jogadores podem ter enquanto interagem com um jogo (figura 8). Essa pesquisa foi feita ignorando a questão da narrativa e focando nas ações que os jogadores podem fazer dentro dos jogos.

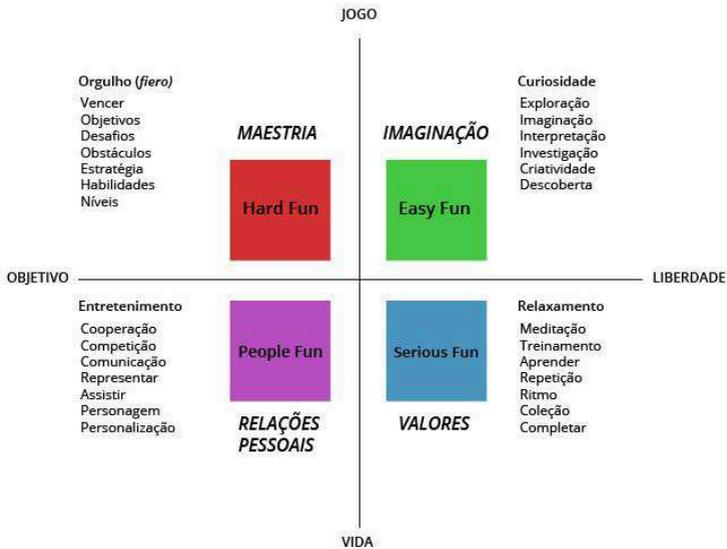
As quatro emoções definidas são as seguintes:

- *Easy Fun* – onde a questão de ganhar ou perder é fraca, o jogador tem interesse em descobrir o mundo de jogo e viver a fantasia com curiosidade;
- *Hard Fun* – onde o sentimento de vencer é o motivo pelo qual o jogador interage com o jogo. Ele coloca suas habilidades em prova e deseja completar o jogo da melhor forma possível;
- *Serious Fun* – onde o importante para o jogador é como ele se sente enquanto joga. A conexão entre jogo e jogador é profunda e importante;
- *People Fun* – onde jogar chega a ser uma desculpa para interagir com outras pessoas, mesmo que o tema de jogo não seja interessante para o usuário.

Cada uma das quatro emoções apresentadas representam conjuntos de ações e interações que existem entre jogo e jogador. Os objetos a serem estudados são exatamente essas interações e o que suas combinações proporcionam ao jogador.

Não como uma regra, mas como uma direção, a pesquisa pode ser usada para entender como o jogador se sente, o que é extremamente importante para entendermos qual experiência queremos passar.

Figura 8 - Gráfico das Chaves para a Diversão



Fonte: XEODesign – Adaptado

### 3.3 QUATRO TIPOS DE JOGADOR

Mesmo entendendo como os jogadores se sentem e como é a comunicação entre jogo e jogador, é preciso entender suas motivações e interesses dentro de um jogo. Richard Bartle (1996) desenvolveu quatro arquétipos de jogadores (figura 9), levando em consideração os jogadores em si, o mundo do jogo, ações dentro do jogo e interações dentro do jogo. Os arquétipos são os seguintes:

- *Matadores* – são os jogadores competitivos que agem sobre outros jogadores buscando algum tipo de competição;
- *Socializadores* – são os jogadores cooperativos, que buscam interagir com outros jogadores a fim

de desbravar o conteúdo do jogo de um modo amigável;

- *Realizadores* – são os jogadores que se interessam em agir sobre o jogo, buscando conquistar os objetivos que o jogo dispõe.
- *Exploradores* – são os jogadores que interagem com o jogo para explorar tudo que for possível. O mundo do jogo é muito interessante para eles.

Mesmo esse estudo não é completo em relação ao desejo humano, existem altos e baixos, mas com esses arquétipos é possível entender um pouco mais a fundo os objetivos que os jogadores podem ter dentro de um jogo.

*Figura 9 - Taxonomia de Richard Bartle*



Fonte: The Art of Game Design, Book of Lenses – Adaptado

### 3.4 GAME DESIGN VS LEVEL DESIGN

“*Game design é o ato de decidir como o jogo deve ser*” (Schell, 2010). A atividade de *game design* é feita pelo *game designer*, e a responsabilidade cai sobre todos os aspectos que aparecem no jogo.

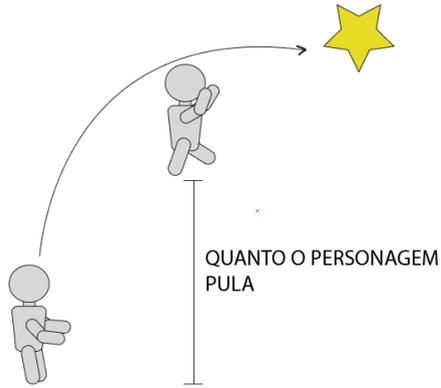
Não necessariamente precisa ser um programador de jogos ou artista, o *game designer* precisa saber e entender a melhor forma de passar a experiência do jogo para o jogador (figura 10). Por exemplo, em um jogo onde o personagem pula quando jogador aperta o botão, quão alto o personagem pula? Ele consegue chegar ao objetivo em sua frente apenas dando um pulo? Ou ele precisa primeiro pular em uma plataforma suspensa e então ele consegue alcançar o objetivo? Essas são questões que um *game designer* precisa resolver, o quão alto o personagem pula e suas sequências para alcançar o objetivo.

Seguindo os mesmos preceitos da atividade de *game design*, o *level design* é feito pelo *level designer*. Ele também é responsável por tomar as decisões de como o jogo deve ser, mas a sua atuação ocorre nos cenários do jogo (figura 11), mais especificamente onde o personagem anda, corre, pula, escala etc.

Algumas responsabilidades de *level design* são: posicionamento dos elementos no cenário do jogo; distância dos elementos no cenário do jogo; criação dos cenários do jogo seguindo a experiência principal do jogo (jogo para um jogador? Jogo de equipe contra equipe?); progressão do jogador sobre o cenário; e todo o resto que faz conexão entre jogador e cenário do jogo.

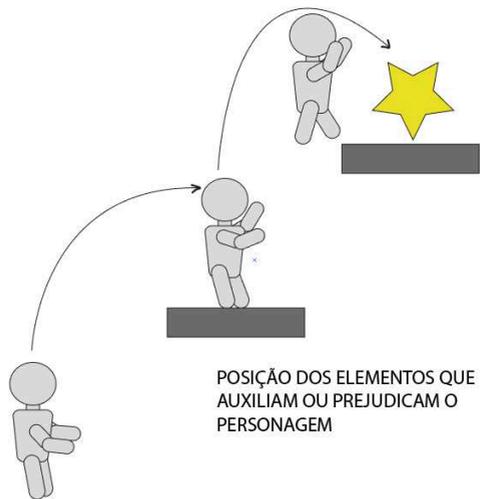
“*O game designer formula as regras do jogo, enquanto o level designer as interpreta criando seus resultados.*” (Kremers, 2009). O *level designer* aplica as decisões de *game design* para dentro dos cenários que irá desenvolver, criando a progressão do jogador sobre o jogo.

*Figura 10 - Atividade de Game Design*



Fonte: Autor

*Figura 11 - Atividade de Level Design*



Fonte: Autor

As duas atividades sempre irão andar juntas, mesmo que suas responsabilidades sejam separadas em equipes ou pessoas diferentes. Como o objetivo do trabalho é criar um cenário para ser usado no jogo *Crystal League*, as atividades a serem seguidas serão as de *level design*.

### 3.5 MECANISMOS DE ENSINO

O *level designer* vai guiar o jogador pelo mundo do jogo; ele vai definir os caminhos para alcançar os objetivos e também criar os desafios a serem superados.

Esses caminhos que o *level designer* cria para o jogador precisam estar sempre concisos com a experiência que o jogo deve passar. Não só isso, muitas vezes o jogador precisa aprender a jogar o jogo, e quem ensina isso é para ele é o *level designer*.

O jogador pode ter um manual do jogo, que o explica como cada botão do controle funciona e como cada mecânica do jogo funciona. Essa versão pode não ser a ideal, e o motivo é porque quebramos a imersão do jogador ao passarmos essas informações explicitamente e fora do ambiente do jogo.

Figura 12 - Manual do jogo *The Legend of Zelda: Ocarina of Time*



Fonte: *The Legend of Zelda: Ocarina of Time Manual*

Na imagem acima o manual do jogo *The Legend of Zelda: Ocarina of Time* explica como funciona o uso da espada dentro do jogo.

Com o tempo esse tipo de manual foi sendo menos usado e os ensinamentos começaram a aparecer dentro do próprio jogo. Alguns jogos ainda usam ele explicitamente como na figura 12.

*Figura 13 - Como Batman: Arkham Asylum ensina o jogador*



Fonte: *Batman: Arkham Asylum*

Analisando a figura 13, fica claro o que o jogador precisa fazer, e ele não precisou abrir um manual do jogo para entender como as coisas funcionam. Ainda assim, esses tutoriais não guiam o jogador, eles falam explicitamente o que o jogador precisa fazer. É uma opção viável para muitos jogos, mas isso pode atrapalhar a satisfação dos jogadores de entender o mundo do jogo. Mostrar por escrito o que o jogador precisa fazer, junto com qual botão ele deve apertar não é um bom exemplo de como guiar os jogadores.

Existem formas de guiar o jogador sem usar mensagens explícitas e dizer qual tecla ele deve apertar e onde, o que pode proporcionar uma experiência mais suave e sem interrupções na imersão com o mundo do jogo.

Os jogos *Megaman*, por exemplo, podem ser considerados exemplos de como ensinar o jogador através do seu *level design*.

*Figura 14 - Início do jogo Megaman X*



Fonte: *Megaman X*

No primeiro nível do jogo *Megaman X* o jogador aprende quase todas as mecânicas do jogo, e o mais importante disso é que ele aprende sem aparecer nenhuma caixa de diálogo e sem ler uma página de manual, o jogo o ensina enquanto ele joga.

Na figura 14 acima podemos perceber que o personagem só pode andar para sua esquerda pois existe uma parede ao seu lado. Após andar alguns passos um primeiro inimigo aparece (figura 15), e apenas duas coisas podem acontecer: uma o jogador é atingido pelo inimigo e perde vida, entendendo que ele possui pontos de vida; e a outra é que ele pode pular por cima do inimigo. Mesmo ele sendo atingido pelo inimigo, esse acontecimento se repete até ele conseguir pular por cima dele.

Figura 15 - Como Megaman X ensina o jogador



Fonte: *Megaman X*

Seguindo esses preceitos o jogo continua, mostrando para o jogador como tudo funciona sem tirar a sensação de satisfação que ele pode ter ao ultrapassar um desafio.

Buscando sempre usar esse último exemplo de *level design* e ensinamentos ao jogador, podemos aplicar muitos conceitos do mundo real e do dia-a-dia para dentro do jogo, facilitando o entendimento para o jogador.

### 3.6 AFFORDANCES

O termo *Affordance* vem do inglês e pode ser traduzido de algumas formas. Uma delas são *reconhecimento* e *possibilidade*. *Affordance* significa que o objeto em questão não precisa de uma explicação prévia de como deve ser usado, ele em si e seu contexto já deixam clara a sua funcionalidade.

Dentro do contexto de jogos digitais o conceito é usado para melhorar a imersão do jogador, ou seja, o quanto mais o jogo parecer um mundo verídico para ele melhor. Questões como elementos de HUD são definidos aqui, pois do mesmo jeito que podem ajudar o jogador, eles acabam nos lembrando de que aquilo é apenas um jogo.

Figura 16 - Como *Assassin's Creed: Origins* aborda objetivos



Fonte: *Assassin's Creed Origins*

No exemplo com a figura 16, o jogo *Assassin's Creed Origins* mostra os variados pontos de objetivos usando elementos de *HUD*. Existem quatro categorias de *Affordances*, definidas por Emilia Schatz (2017), explicados em seu artigo *Defining Environment Language for Video Games*, para o site *80 Level*:

1. *Explicit* – quando as *affordances* aparecem de forma explícita, quebrando o realismo do jogo. Exemplo, quando o jogo fala qual botão no controle o jogador deve apertar;
2. *Defined* – quando as mecânicas do jogo criam uma associação com o seu visual. Exemplo, todas as escadas e paredes escaláveis do jogo são amarelas;
3. *Cultural* – quando conseguimos reconhecer o funcionamento porque funcionam como no mundo real. Exemplo, uma escada e uma válvula em um cano;
4. *Inferred* – os jogadores deduzem seu funcionamento de acordo com seu conhecimento sobre as regras do jogo. Exemplo, se o jogador fica preso em um buraco ele precisa sair, podendo pular, escalar, e qualquer outra maneira que o jogo

permita (usar uma corda do inventário, por exemplo).

*Figura 17 - Como Uncharted 3 aborda objetivos*



Fonte: *Uncharted 3*

Em *Uncharted 3* fica claro por onde o jogador consegue escalar, e não existe nenhum elemento de HUD que o faz seguir esse caminho (figura 17).

Dentro das quatro categorias de *affordances*, as duas primeira devem ser evitadas, pois quebram a imersão do jogador para o mundo, e devem ser usadas apenas quando não há outra opção. Já as duas últimas devem ser usadas o máximo possível.

Junto com o conceito de *Affordances* podemos levar em consideração as formas geométricas primitivas (figura 18), buscando familiarizar o jogador com o mundo. Cada uma das três pode falar com o jogador de uma forma diferente.

Figura 18 - Como as formas afetam o jogador



Fonte: Emilia Schatz - Adaptado

Todos esses princípios são trabalhados em conjunto onde o *level designer* os utiliza para criar um ambiente imersivo para o jogador, como um parque de diversões, interior de um castelo ou mesmo uma floresta nunca habitada.

### 3.7 LEVEL DESIGN COM A DISNEY

Parques temáticos têm como objetivo levar uma pessoa de uma atração para outra da maneira mais eficaz possível, que é como um nível de jogo deveria funcionar. Não apenas guiar a pessoa de um lugar para o outro, o próprio lugar conta uma história com a sua temática, usando arquitetura e ambientação de todos os seus elementos, e um jogo não é diferente disso.

Um dos conceitos mais atuais e usados tanto nos parques da Disney quanto em níveis de jogo é de *Weenies*. *Weenies* são pontos de localização que servem tanto para guiar o jogador até um local quanto para ele se localizar em um espaço. Criando essa referência de navegação ele tem o controle do que pode fazer e não se perde no espaço do jogo.

Figura 19 - Como Weenies guiam jogador



Fonte: *Journey*

Em *Journey* (figura 19), por exemplo, a montanha no fundo funciona como um *Weenie*, deixando claro para onde se deve ir. Um *Weenie* nem sempre está visível para o jogador, mas em pontos cruciais ele precisa aparecer e ajudar o jogador nas suas decisões de qual caminho prosseguir.

Algo em que os parques da Disney são muito fortes e causam referência para a criação de jogos e experiência do jogador é a estrutura do parque em si e seus caminhos. Criando caminhos que levam até os *weenies*, existem algumas formas diferentes de posicionar os elementos e criar o próprio caminho, e dependendo do modo escolhido o resultado pode ser uma experiência totalmente diferente. Muitos elementos posicionados em um ambiente pode criar uma área maior de liberdade e exploração, ao mesmo tempo que um caminho mais linear pode ser mais atrativo com muitos elementos visuais fora do escopo de alcance do jogador.

Além de *Weenies* e os caminhos do jogador, existem outros métodos para guiar jogadores em seu caminho sem lhes dizer explicitamente o que fazer e para onde ir. A iluminação dentro de um ambiente é muito forte também,

usando pontos de luz isolados e indicando um local importante a visitar (exemplo na figura 20). A ausência de luz pode ser um indicador também, tendo como exemplo a Disney, quando as luzes dos parques apagam quer dizer que é hora de ir embora.

*Figura 20 - Como a iluminação guia o jogador*



Fonte: *Uncharted 4*

Outro método muito importante e muitas vezes ignorado é a questão de composição do cenário. Suas formas e linhas de ação também guiam o jogador para seu objetivo, como mostra a figura 21.

Figura 21 - Como formas e linhas guiam o jogador



Fonte: *Uncharted 3* – Vídeo de Mark Brown

O formato quadrado da janela indica ao jogador que é um espaço onde ele deve passar e a linha no topo do telhado indica o caminho que ele deve seguir. Essa cena inteira do jogo *Uncharted 3* segue esses princípios até seu fim. Usando iluminação, formatos, linhas de ação *weenies* e *affordances*.

Dentro da montagem dos ambientes, Scott Rogers explica na sua palestra para a *Game Developers Conference* de 2009 três tipos de experiências que podem ser passadas ao jogador, chamadas de ilusões:

- *The Illusion of Setting* – que significa ilusão de configuração, onde a distribuição dos elementos e caminhos do jogo permitem ao jogador escolher quais elementos ele quer interagir ou não; se ele quer ir direto para a atração principal ou se ele quer conhecer as pequenas atrações pelo caminho;
- *The Illusion of Freedom* – que significa ilusão de liberdade, onde o jogador tem seu objetivo a chegar mas existem variados caminhos para chegar lá,

dando liberdade ao jogador a escolher por onde ele quer ir e quais pequenos elementos ele quer ver enquanto se dirige ao objetivo principal;

- *The Illusion of Exploration* – que significa ilusão de exploração, onde os elementos são distribuídos de uma forma com que ele interaja o máximo possível com o cenário, deixando o jogador explorar o ambiente como ele quiser pelos caminhos que ele quiser.

Esses conceitos estão presentes dentro dos parques da Disney e servem de auxílio para a montagem de níveis para os jogos digitais. Levando em consideração todos esses conceitos, fica claro o quão importante é ensinar e guiar o jogador ao seu objetivo, criando tanto elementos que o ajudam nessa progressão quanto os empecilhos.

### 3.8 ESPAÇOS

Espaços podem ser definidos como conceitos tanto cognitivos quanto físicos. Na parte cognitiva são levadas em consideração as informações que um jogo contém e como elas interagem com o jogador, junto com as informações que o jogador pode ter sobre o jogo.

Em um jogo de Xadrez, por exemplo, os dois jogadores possuem uma visão do campo e podem saber os possíveis acontecimentos, mas cada uma segue uma estratégia em sua cabeça.

Jogos podem possuir muita informação dentro de si, se caso muitas informações forem forçadas ao jogador ele pode não ter controle disso, deixando-o frustrado. Assim, a tentativa é seguir a Regra dos Sete (Geroge A. Miller, 1955), que nos diz que a capacidade do ser humano de absorver informações é perto do número sete. Podemos perceber esse conceito em jogos clássicos e bem sucedidos, como *Doom*, que possui sete inimigos diferentes, por exemplo.

Analisando os inimigos do jogo *The Legend of Zelda: Breath of the Wild* percebe-se que o número de espécies de inimigos não é grande e fica perto do número sete, mas cada um dos inimigos possui variações visuais que indicam a sua força (exemplo na figura 22). Mesmo deixando o número de inimigos diferentes muito grande, é possível identificar suas variações.

Figura 22 - Variações de um inimigo



Fonte: *The Legend of Zelda: Breath of the Wild*

O espaço que a mente humana tem de receber e canalizar informações não pode ser subestimado, e levando isso em consideração o espaço físico de um jogo digital pode ser definido. Pontos como tamanho do ambiente, altura de escadas, altura dos degraus de uma escada, tamanho de carros etc. ajudam o jogador a não se perder dentro do jogo e todas as métricas devem ser pensadas.

Como exemplo o jogo *Overwatch*, suas fases possuem os espaços totalmente pensados. É possível perceber que os elementos de verticalidade (andares, muros puláveis etc.) possuem uma altura muito próxima um do outro, se não a mesma (exemplo na figura 23). Isso causa uma sensação de familiaridade para o jogador, e mesmo se for uma fase nova dentro do jogo, os princípios estão ali e o ambiente não vai parecer totalmente estranho para ele.

Figura 23 - Verticalidade em Overwatch



Fonte: *Overwatch*

Quando se fala na questão física de espaços para jogos, a primeira coisa a se pensar é a câmera e como ela vai funcionar. Existem alguns tipos de câmeras para jogos 3D, e dependendo do jogo o jogador tem o poder (ou não) de controlar a câmera do jogo. A câmera é a principal referência que o jogador tem para todos os elementos do cenário.

Usando a câmera o jogador percebe como o seu personagem se movimenta, a qual distância ele está do objetivo, e ela deve ajudá-lo a entender melhor suas habilidades. Por exemplo, se um personagem tem duas magias para usar em um alvo e essas magias possuem uma distância mínima para funcionarem, a câmera precisa passar essa referência de espaço para o jogador.

Dentro da questão de espaço também é levado em conta como funciona o caminho que o jogador percorre. No livro *The Art of Game Design: A Book of Lenses*, Jesse Schell explica cinco tipos de caminhos:

1. Caminho Linear – Quando o jogador tem apenas uma direção para seguir, algumas vezes ele pode ir pra frente e para trás, mas existe apenas um caminho. Exemplo figura 24;

*Figura 24 - Exemplo de caminho linear*



Fonte: *Crash Bandicoot 3*

2. Caminho em Grid – Quando a locomoção pelo mundo do jogo se baseia em se movimentar dentro de um grid. Não só jogos de turnos podem possuir um caminho em grid (como Xadrez), mas outros jogos também. Usando grid fica fácil os jogadores entenderem a movimentação dos inimigos, por exemplo, e a matemática da programação fica mais fácil. Exemplo figura 25;

*Figura 25 - Exemplo de caminho em grid*



Fonte: *The Legend of Zelda*

3. Caminho em Teia – Quando o jogador possui várias maneiras diferentes de chegar a um objetivo seguindo caminhos diferentes que se conectam entre si. Exemplo figura 26;

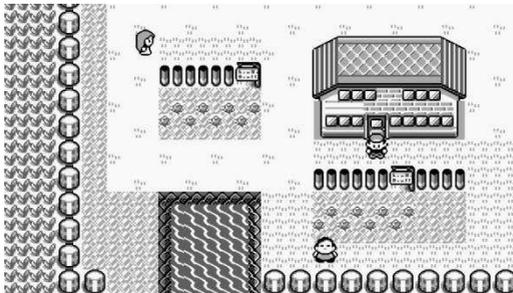
*Figura 26 - Exemplo de caminho em teia*



Fonte: *Grand Theft Auto: Vice City*

4. Caminho em Pontos no Espaço – Quando jogos possuem objetivos distribuídos em um espaço e eles podem aproximar esses objetivos quando acharem interessante. Exemplo figura 27;

*Figura 27 - Exemplo de caminho em pontos do espaço*



Fonte: *Pokemon Red*

5. Caminho Dividido em Espaços – Quando jogos tentam usar o mapa real, dividindo os espaços no mundo do jogo por áreas irregulares. Cada área possui características específicas. Exemplo figura 28.

Figura 28 - Exemplo de caminho dividido em espaços



Fonte: *The Legend of Zelda: Ocarina of Time*

Esses cinco tipos de caminhos são usados para criar a progressão do jogador, podem ser vistos tanto em jogos *Singleplayer* (de um jogador) quanto em jogos *Multiplayer* (vários jogadores simultâneos). A combinação de cada tipo de caminho também é possível, jogos podem possuir mais de um estilo de caminho dependendo da progressão do jogador dentro do jogo, diferentes lugares podem ser abordados de formas diferentes. A decisão de qual estilo seguir sempre vai depender a experiência que queremos passar para o usuário.

### 3.9 ARQUITETURA E ORGANIZAÇÃO ESPACIAL

A arquitetura e seus fundamentos também podem fazer parte do conceito de fases para jogos. *Francis D. K. Ching* define alguns princípios de arquitetura e organização espacial em seu livro *Arquitetura: forma, espaço e ordem*.

Para criar ordem em uma composição de arquitetura *Ching* explica alguns pontos que ele chama de “Princípios de Ordem”. Esses princípios são usados para criar uma unidade a composição, unindo todos os seus pontos. Os princípios listados são:

- Eixo – Uma reta que é estabelecida por dois pontos no espaço em que se pode definir os dois espaços separados cada um por seu contexto ou função;
- Simetria – A distribuição dos elementos é equivalente em lados opostos de um plano divisor ou centro de um estrutura;
- Hierarquia – Algum elemento se destaca entre os outros a partir de qualquer ponto que é usado na estrutura (forma, espaço, tamanho, localização etc.);
- Ritmo – Existe um elemento unificador que é criado por repetição na estrutura. Essa repetição deve criar um padrão na sua sequência;
- Dado – Um plano ou volume na estrutura é usado para definir e organizar o padrão dos elementos da estrutura;
- Transformação – Princípio em que os elementos de uma estrutura podem ser alterados por manipulações feitas que respeitam um contexto, sem perder a identidade do conceito da estrutura.

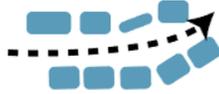
Os princípios são usados para equilibrar a estrutura e balancear seus elementos para que a composição inteira crie uma identidade e sirva para seu conceito.

Outro conceito que *Ching* descreve em seu livro é o de organizações espaciais (figura 29). Cada organização de arquitetura pode ter uma forma diferente e existem alguns preceitos de formas que tendem a ser eficazes para diferentes objetivos. Os cinco tipos de organização espacial são:

- Organização Linear – É uma sequência linear de espaços repetitivos. Serve para criar um senso de direção forte a quem está no espaço, normalmente tem uma ligação direta a um objetivo;
- Organização Aglomerada – São espaços agrupados por compartilharem alguma característica em relação ao contexto em que estão. Podem ser usados para criar pequenos ambientes com diferenciação (como objetivos);
- Organização em Malha – São espaços organizados dentro de uma malha estrutural. Podem servir como labirintos ou com o objetivo de fazer o usuário percorrer todos os pontos do ambiente;
- Organização Centralizada – São espaços que possuem um elemento central dominante onde ao seu redor outros espaços secundários são agrupados;
- Organização Radial – São espaços que se desenvolvem a partir de seu elemento central de uma maneira radial.

Figura 29 - Tipos de Organização Espacial

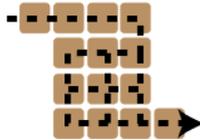
### Organização Linear



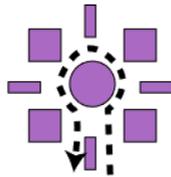
### Organização Aglomerada



### Organização em Malha



### Organização Centralizada



### Organização Radial



Fonte: Arquitetura: forma, espaço e ordem - Adaptado

### 3.10 SINGLEPLAYER VS MULTIPLAYER

Existem jogos que são para um jogador e também jogos para vários jogadores. Esse tipo de definição é aplicada desde o começo e o ponto mais importante da aplicação disso é o *level design*.

Jogos de apenas um jogador (*Singleplayer*) são os mais comuns (figura 30). O principal foco é contar uma história através de uma atividade interativa, a narrativa é um ponto muito forte. Como apenas um jogador irá interagir com o mundo jogo, e não vários, o controle de como as coisas devem funcionar no jogo não é tão difícil (comparando com um jogo de vários jogadores).

*Figura 30 - Exemplo de jogo para apenas um jogador*



Fonte: *Super Meat Boy*

Jogos de vários jogadores (*Multiplayer*) também existem e não são poucos (figura 31). Eles buscam desenvolver uma conexão entre os jogadores e o mundo do jogo, criando experiências de competição e cooperação.

Já nesses jogos que existe o componente de interação entre jogadores, alguns pontos ficam mais complexos de se desenvolver. O controle de como os jogadores vão interagir com o jogo já não é tão linear, o balanceamento precisa de muito mais aperfeiçoamento e a narrativa pode ser entendida de maneiras diferentes.

*Figura 31 - Exemplo de jogo para vários jogadores*



Fonte: *World of Warcraft*

Não necessariamente um jogo deve ser de um jogador ou de vários jogadores (figura 32), ele pode ser dividido em modos de jogo que aplicam os dois conceitos, passando experiências diferentes.

*Figura 32 - Exemplo de jogo para um e dois jogadores*



Fonte: *Donkey Kong Country: Tropical Freeze*

Este projeto tem como objetivo criar uma fase para o jogo *Crystal League*, os tópicos a serem aprofundados serão na questão de *Multiplayer*, levando em conta tanto cooperação quanto competição.

### 3.11 MULTIPLAYER LEVEL DESIGN

Pensando na experiência a ser passada dentro de jogos (ou fases) para vários jogadores, as principais experiências a levar em conta são de cooperação e competição, tanto usando apenas uma delas ou ambas.

A experiência de cooperação se define em os jogadores precisarem se unir para alcançar um objetivo em comum, isso pode ser tanto como derrotar um chefe muito forte quanto obter informações importantes através de conversas dentro do jogo. A cooperação também pode ser encontrada dentro de experiências competitivas, onde dois times enfrentam para vencer em uma batalha.

A competição pode ser definida como um objetivo que só será cumprido derrotando os jogadores inimigos ou não os deixando vencer. Duas equipes lutam entre si e ganha a primeira que derrotar o núcleo do time adversário. Ou também equipes lutam para terminar uma masmorra mais rápido que a outra, sendo que a que tiver o melhor tempo é a vencedora, como *World of Warcraft* faz (figura 33).

*Figura 33 - Imagem de campanha para campeonato de World of Warcraft*



Fonte: *Blizzard Entertainment*

Os objetivos dentro de jogos *multiplayer* são definidos de várias formas, e uma delas são os modos que um jogo pode ter, onde cada fase por ser específica para um

modo de jogo ou possuir vários modos diferentes. Cada um dos modos de jogo deve passar uma experiência diferente, usando as mesmas mecânicas.

Alguns modos mais famosos para jogos *multiplayer*, como mostra a figura 34, são:

- Capturar a Bandeira – Um objeto carregável é usado para conseguir pontuação;
- Capturar o Ponto – Capturar um lugar específico no mapa antes da equipe adversária;
- VIP – Apenas uma pessoa específica da equipe consegue pontuar;
- *Death Match* – A equipe que mais matar vence;
- Plantar a Bomba – Carregar e posicionar a bomba em um lugar específico no mapa;
- Escolta – Defender um jogador ou NPC e leva-lo para um lugar determinado no mapa;
- *Battle Royale* – Quando um grande número de jogadores entra em um cenário extenso e vence o último que sair vivo.

Figura 34 - Modos de jogo Multiplayer



Fonte: Bobbyross.com - Adaptado

Definir o modo de jogo para a fase é um dos primeiros conceitos a se levar em conta, e todo o resto da fase será pensado em prol do objetivo que o modo proporciona.

Pensando nos jogadores dentro de um jogo *multiplayer*, existem cinco comportamentos que eles tendem a ter dentro do jogo (figura 35):

Figura 35 - Relação de posicionamento e comportamento do jogador



Fonte: Bobbyross.com - Adaptado

Esses comportamentos mostram como os jogadores atuam dentro do cenário, de acordo com o modo de jogo e o ambiente que estão. Existem os *Rushers*, que são os jogadores que correm até o objetivo o mais rápido possível, criando um risco muito alto para si mesmo; Os *Roamers* são os jogadores que ficam a uma média distância e estão sempre avançando ao objetivo, mas tomam mais cuidado que os *Rushers*. Os *Snipers* jogam a uma longa distância, tomando muito cuidado e prestando atenção na movimentação dos inimigos. Atuam como suporte aos que buscam o objetivo. Os *Campers* são os jogadores que ficam totalmente escondidos esperando os adversários cometerem um erro ou aparecem a sua frente sem uma chance de defesa. Os *Stealths* são os jogadores que sempre tomam um caminho adicional para chegar aos inimigos por trás. Tentam sempre passar despercebidos e são muito cuidadosos.

Esses cinco comportamentos são levados muito a sério e em competições profissionais de *e-sports* acabam sendo a base para a criação de um time.

O *level designer* precisa pensar e criar momentos para os comportamentos que o jogador pode ter, sendo esses cinco ou outros a mais que possam ser vistos em seu jogo. Esses ambientes precisam ser pensados e balanceados para que um comportamento não fique com uma vantagem grande sobre o outro.

O espaço do ambiente a ser jogado precisa aceitar distâncias variadas para o combate, dependendo dos elementos que o jogador usa (armas, machados, magias etc.). Essas distâncias podem ser balanceadas tanto dentro do *game design*, fazendo modificações na força das armas, quanto pelo *level design*, criando áreas passíveis a tal tipo de combate, como na figura 36.

Figura 36 - Distância de combate entre jogadores



Fonte: Bobbyross.com - Adaptado

Dentro dos espaços e comportamentos é levado em conta os tipos de proteção que a fase pode ter para proteger os jogadores (figura 37). As proteções são definidas por “total” e “média”. A proteção total cobre todo o corpo do personagem dentro da fase, onde ele não é visto; a proteção

média acontece quando os elementos da fase não cobrem todo o corpo do personagem, algumas partes dele aparecem dependendo de como ele se posiciona e pode ser visto e atingido pelos adversários. Dentro dessas duas proteções entram mais dois conceitos, de proteção “dura” e proteção “fofa”. A proteção “dura” protege o jogador de todos os ataques sobre ele, sendo uma parede onde as balas não penetram; e a proteção “fofa” podem esconder o personagem visualmente mas ele pode ser atingido por balas, como um arbusto, por exemplo.

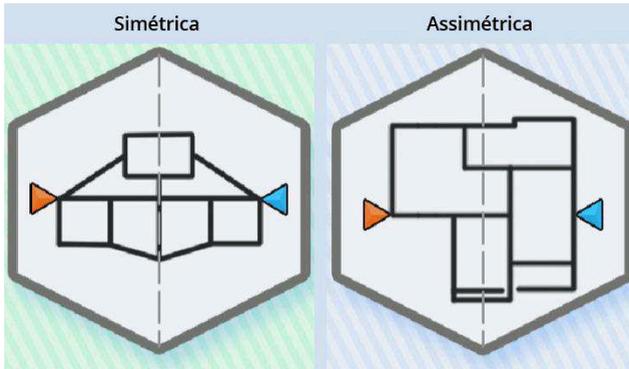
Figura 37 - Tipos de proteção



Fonte: Bobbyross.com - Adaptado

A simetria das fases é levada em conta também (figura 38). Existem as fases que são simétricas, onde ela é espelhada, cada metade é igual uma a outra; e a fases assimétricas, onde sua planta não é simétrica, tendo pontos diferentes em todo o mapa.

Figura 38 - Simetria e Assimetria de fases



Fonte: Bobbyross.com – Adaptado

As fases simétricas são menos complexas de fazer, junto com um balanceamento rápido, mas precisa de um contraste visual forte para os jogadores navegarem pela fase e entenderem os seus principais locais. Já as fases assimétricas são mais complexas de se criar, precisam de muitos testes para seu balanceamento mas deixa muito mais clara a navegação do jogador.

Não existe melhor tipo de fase para determinado modo de jogo, isso pode ser estudado e testado de acordo com as mecânicas que o jogo possui.

Os estudos de Ross (2015) mostram um caminho a seguir no desenvolvimento de *level design* para um jogo *Multiplayer*. Os exemplos de mecânicas usadas no seu estudo são de jogos de *FPS*, mas isso não limita os conceitos para esse único gênero. Tanto os modos de jogo quanto o conceito de comportamento dos jogadores podem ser aplicados de outra maneira, como no jogo *Crystal League*, por exemplo. Esses tópicos vão ser sintetizados para o contexto do jogo *Crystal League* mais a frente neste projeto.

### 3.12 BALANCEAMENTO

O balanceamento é uma parte crucial no desenvolvimento de jogos, mesmo em jogos de um jogador, mas principalmente em jogos *Multiplayer*. Não apenas no *level design*, o balanceamento é visto também nas mecânicas, como armas, munições e velocidade do jogador.

Um jogo continua sendo balanceado em suas mecânicas até a sua data de lançamento, até porque mudar os atributos de armas e munições é mais fácil que ajustar elementos de uma fase. Por esse motivo são gastos muito tempo em testes e QA, porque no processo de produção da fase o que pode ser mudado são tanto as mecânicas quanto os elementos do cenário.

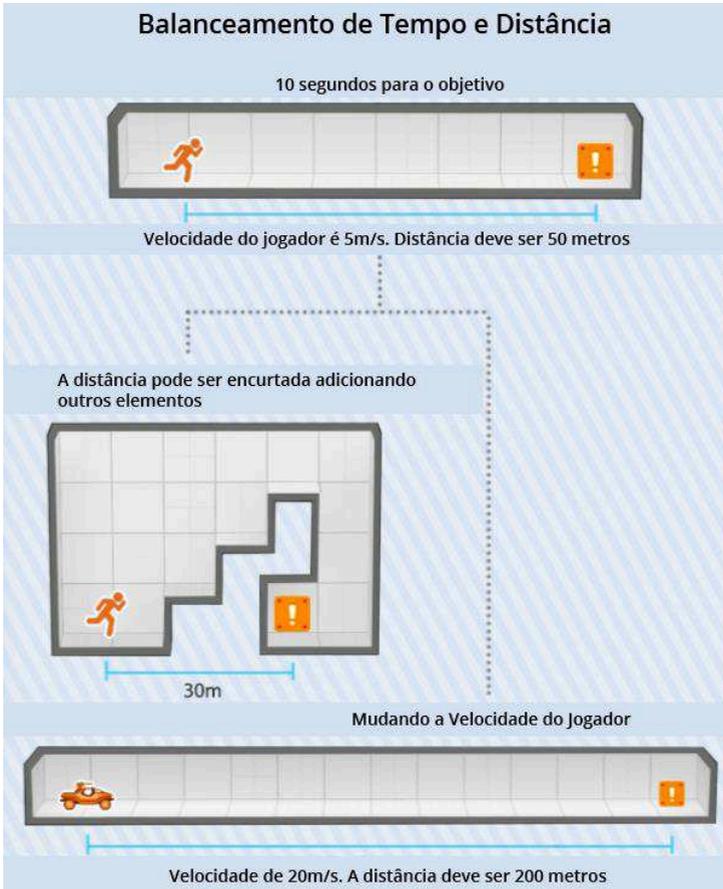
O que consiste em um balanceamento é deixar a fase jogável para todos os jogadores e equipe, sem um tendo vantagem inicial ou constante. Vantagem levando o termo no modo geral, pois em algumas ocasiões um time vai ter vantagem sobre o outro (e vice-versa) para que o conflito seja desenvolvido.

Dentro do balanceamento de *level design* alguns pontos que são levados em consideração são:

- Tamanho da fase;
- Tamanho dos pontos de combate da fase;
- Distância do objetivo para as duas equipes;
- Tempo para o jogador chegar no objetivo;
- Posicionamento de Power Ups;
- Modo de navegação para cada equipe;
- Local de Início dos jogadores;
- Simetria e assimetria que fortalecem o objetivo;
- Risco VS Recompensa;
- Elementos de cooperação e competição.

Isso tudo entra na questão de balanceamento, de um modo geral. Cada um dos pontos listados podem ser abordados de maneira mais profunda e relacionados com alguma atividade dentro do jogo, onde o balanceamento é necessário (figura 39).

*Figura 39 - Balanceamento de tempo e distância*



Fonte: Bobbyross.com

## 4 SÍNTESE

Esta etapa do Projeto de Conclusão de Curso aborda a inclusão do conteúdo estudado junto com o jogo *Crystal League*, levando em conta os modos de desenvolvimento da equipe e suas definições de *game design*.

### 4.1 IDEIA

A proposta do cenário é aplica-lo em um novo modo de jogo, o *Battle Royale*, seguindo a pré-produção (que será criada na etapa de Desenvolvimento deste projeto), junto com a criação de peças modulares e etapas de iteração.

As peças modulares vão servir, a princípio, para a criação do cenário deste projeto, mas a sua utilização pode abranger outros futuros cenários que o jogo possa ter.

As etapas de iteração do desenvolvimento vão ter um planejamento e vão seguir métodos de *Playtesting*, onde o estado atual será testado pela equipe interna e possíveis eventos de testes da região ou da própria equipe para o público.

### 4.2 CONTEXTO DO JOGO CRYSTAL LEAGUE

O conteúdo que será apresentado sobre o jogo *Crystal League* terá ligação com o intuito deste projeto, que trabalhar a atividade de *level design* com modularidade e iteração.

*Crystal League* é um jogo do gênero *Multiplayer Online Battle Arena (MOBA)* com personagens “cartunizados” que podem se movimentar pelos cenários dentro e fora dos veículos. Os diferentes modos de jogo podem ser jogados de forma cooperativa em times ou cada um por si.

A narrativa do jogo se passa ao redor de um parque de diversões que concentra diversos *HUBS* temáticos, com

atrações que levam os jogadores aos estágios em si. Cada *hub* concentrará um número de estágios e terá um tema próprio, como “medieval”, “espacial” e “faroeste”.

#### 4.3 DEFINIÇÕES DE GAME DESIGN

O jogo possui alguns modos de jogo e o intuito deste projeto é fazer um cenário para um novo modo de jogo, *Battle Royale*.

O jogo é 3D e a câmera de controle pode ser em terceira ou primeira pessoa.

Os personagens que os jogadores controlam não possuem diferenças em ações, vantagens ou desvantagens sobre os outros, a diferença é em questão de estética. Eles andam, correm e pulam, sobem escadas e podem flutuar de paraquedas. Todas essas ações acontecem de acordo com o controle do jogador.

Enquanto controlam os personagens os jogadores podem entrar em veículos pelo cenário. Os veículos podem possuir distinção estética e física. Existem veículos que podem carregar um jogador, dois jogadores até cinco jogadores, mas apenas da mesma equipe. Eles também podem possuir diferentes atributos como peso, aceleração e velocidade.

O jogo possui itens que podem ser coletados pelo cenário e serem usados pelos jogadores quando eles acharem melhor. Os itens consistem em atacar inimigos, defender o próprio personagens, aumentar sua velocidade e voar. Os itens se aplicam tanto dentro quanto fora dos veículos.

Além dos itens de uso, os cenários do jogo possuem elementos posicionados estrategicamente para ajudar ou prejudicar o jogador, como aumentar a velocidade e jogar os personagens para o ar.

#### 4.4 MODO DE JOGO BATTLE ROYALE

O modo de jogo *Battle Royale* está forte no cenário atual. Ele cria uma experiência combinando sentimentos de exploração, sobrevivência e combate em um cenário com múltiplos jogadores (entre cinquenta e cem jogadores), e o último que ficar de pé vence a partida.

A partida progride com os jogadores chegando ao cenário, escolhendo onde querem começar e o limite territorial de exploração diminui até que reste apenas um jogador (ou equipe) vivo.

#### 4.5 DIREÇÃO DE ARTE

O estilo artístico para o desenvolvimento desse projeto deve seguir os conceitos do projeto.

Todos os elementos devem ter um motivo para estar ali e fazerem diferença na jogabilidade e refletirem a história e contexto do jogo.

Os objetos a serem criados devem seguir as formas geométricas primitivas com texturas simples e “cartunizadas”. Cada objeto deve conter sua forma, textura e cor de acordo com seu objetivo e devem mostrar que o ambiente está vivo, com variações de tamanho e formas.

As cores a serem usadas devem ser vibrantes e estarem muito bem distribuídas dentro do cenário.

Levando em conta os objetivos estéticos acima precisa-se pensar nas necessidades técnicas para aplicação dos elementos no cenário. Isso inclui economia de polígonos, aplicando detalhes em mapas de textura; reutilização para economizar na produção, contendo variação por proporção ou de texturas; elementos distantes e não acessíveis aos jogadores podem ter texturas em baixa resolução.

## 5 DESENVOLVIMENTO

### 5.1 PESQUISA COM JOGOS REFERÊNCIA

Antes de começar o desenvolvimento do conceito do cenário a ser feito, foi resolvido fazer uma pesquisa de jogos referência e também com os usuários dos títulos que usam o modo de jogo *Battle Royale*. Essas pesquisa servem para identificar mecânicas e padrões dos jogos de sucesso e aplica-las dentro do contexto do jogo *Crystal League*.

Os pontos que estão sendo estudados de acordo com os jogos referência são:

- Tamanho do Cenário;
- Tempo de Partida;
- Meios de Locomoção;
- Formato de Equipe;
- Fluxo do Jogador na Partida.

Na pesquisa com os jogadores foi percebido que o tamanho do cenário (figura 40) é algo bem estável com a experiência de cada jogo, poucas respostas foram diferentes de “Tamanho Bom” (que significa “não tenho objeções”). Isso levou a identificar o ponto de que independentemente do modo jogo *Battle Royale* o tamanho do cenário se define pelas mecânicas principais de cada jogo.

Figura 40 - Respostas da pesquisa sobre Tamanho dos Cenários



Fonte: Autor

O tempo das partidas (figura 41) também foi um ponto que teve poucas respostas diferentes de “Tamanho do tempo Bom” (que significa “não tenho objeções”). Isso reforça a questão de que as mecânicas de cada jogo são mais importantes que o sistema genérico do modo de jogo *Battle Royale*.

Precisa ser levado em conta também que o tempo de partida é variado para cada pessoa dentro da própria partida, ainda mais que normalmente um número grande de jogadores entram na partida mas apenas um número muito pequeno vai até o fim dela. Isso leva a perceber que o tempo pode ser um fator subjetivo na questão da partida já que alguns jogadores são derrotados antes e outros percorrem até o fim.

Levando em conta o tempo das partidas dos jogos referência e aplicando os conceitos dentro do contexto do jogo *Crystal League* foi percebido que o tempo de partida e o tamanho geral do cenário deveria ser consideravelmente menores que os jogos referência. Dentro do contexto *Crystal League*, o jogo que mais chega perto da experiência de tamanho de cenário e partida é o *Surviv.io Battle Royale*

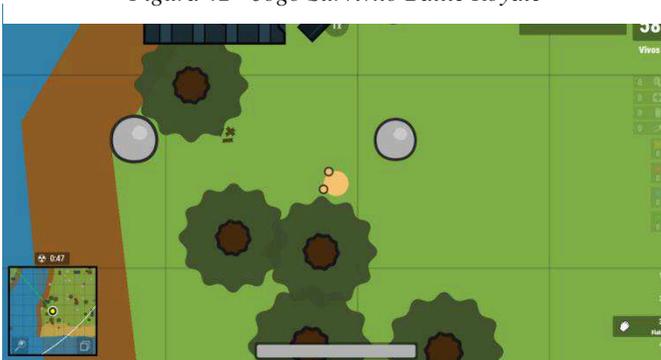
(figura 42). O jogo é um *Top Down Shooter* (jogo de tiro com visão de cima) onde as partidas duram menos de dez minutos mas que ainda assim possuem um cenário grande (comparado com o imaginado para o jogo *Crystal League*).

Figura 41 - Resposta da pesquisa sobre Tempo de Partida



Fonte: Autor

Figura 42 - Jogo *Surviv.io Battle Royale*

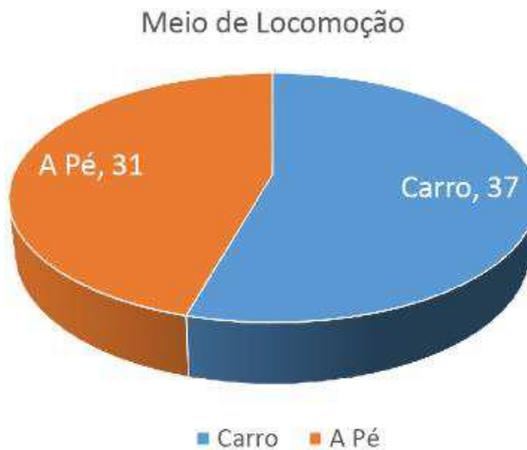


Fonte: <http://surviv.io>

A questão dos meios de locomoção são muito importantes a serem definidas para criarmos a experiência de jogo, principalmente pelo fato de que o jogo *Crystal League* possui tanto a locomoção a pé dos personagens quanto dentro de veículos. A pesquisa (figura 43) mostra que a preferência dos jogadores é da locomoção dentro de veículos.

Existem vantagens para se usar veículos quanto para percorrer os caminhos a pé e precisamos aplicar essas diferenças para que a partida fique balanceada.

*Figura 43 - Resposta da pesquisa sobre Meios de Locomoção*



Fonte: Autor

O formato de equipe (figura 44) precisa ser levado em consideração também, e as respostas da pesquisa indicam que os interesses dos jogadores são balanceados e também são definidos primeiramente pelas mecânicas principais de cada jogo. Também é importante levantar que uma pessoa que não possui amigos que jogam o mesmo jogo que ela prefira jogar suas partidas sozinha, e quem possui

amigos que jogam o mesmo jogo preferem jogar em equipe com esses amigos.

Figura 44 - Resposta da pesquisa sobre Formato de Equipe

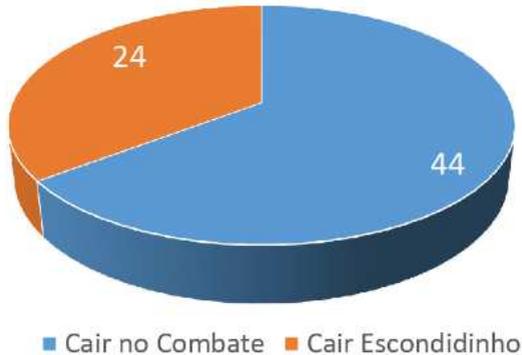


Fonte: Autor

O último dos pontos mais importantes da pesquisa é o fluxo dos jogadores durante a partida (figura 45). A comunidade prefere definir os pontos da partida como *Early*, *Mid* e *Late Game* (começo, meio e fim da partida, respectivamente). No cenário competitivo jogadores tomam diferentes caminhos e decisões dependendo do estado em que a partida se encontra.

Como uma das mecânicas principais dos jogos *Battle Royale* os jogadores podem escolher onde eles querem começar a partida (dentro de um limite, normalmente) e isso fornece duas escolhas principais logo no começo da partida: iniciar o *early game* “caindo no combate” ou “caindo escondidinho”.

Figura 45 - Resposta da pesquisa sobre Fluxo na Partida



Fonte: Autor

As questões do fluxo da partida definem o tipo de experiência que os jogadores querem ter, como uma partida de combate e muita adrenalina ou estratégia e tensão. Dentro do contexto *Crystal League* podemos intensificar as experiências de acordo com os princípios de jogo e o tempo de partida. Como as partidas dentro do jogo *Crystal League* tendem a ser mais rápidas e os cenários menores identifica-se uma predominância da experiência de “cair no combate”.

Levando em conta as mecânicas de funcionamento do modo *Battle Royale* nos jogos referência podemos identificar algumas que podem ser aplicadas ao jogo *Crystal League* sem danificar suas mecânicas principais:

- Área principal e possível acesso a diferentes áreas da fase no começo da partida;
- Campo de Força;
- Veículos temporários;
- Itens tanto de ataque quanto de defesa;
- Itens e Veículos especiais nascerem em locais determinados durante o decorrer da partida.

Essas mecânicas foram definidas pensando tanto no funcionamento do modo *Battle Royale* quanto no contexto do jogo *Crystal League* e o funcionamento de cada uma delas será definido e explicado na parte de desenvolvimento de ideia do cenário mais a frente.

## 5.2 TESTES DE PROCESSAMENTO

O teste de processamento tem como objetivo identificar a quantidade de jogadores possível de jogarem em uma mesma partida e quantos elementos de interação podem ser aplicados (veículos, itens de uso etc.) afetando o mínimo o processamento das máquinas. Imagem dos testes na figura 46.

Figura 46 - Teste de Processamento



Fonte: Autor

Esse teste não permite perceber todas os elementos que serão adicionados ao cenário, mas sim os elementos principais e com mais funcionalidades dentro que acabam sendo os mais pesados em questão de processamento (física de personagens e veículos, funcionamento da IA, etc.).

Como resultado estável para uma primeira produção e testes para uma fase *Battle Royale* o número de 26 personagens na partida se mostrou interessante a partir das

definições gráficas (mostradas a seguir no texto). Desses 26, 1 a 4 podem ser controlados por jogadores humanos e o resto é controlado pelo sistema de IA que o jogo possui.

Levando em conta esse preceito o tamanho do cenário irá seguir uma experiência para esse número de jogadores (26 jogadores) e o número de elementos de interação também será de acordo. Os números dos elementos poderão ir variando de acordo com os passos de iteração e testes no decorrer do projeto.

O uso de elementos modulares criam uma otimização no processamento, isso acontece pois os elementos dentro do cenário são os mesmos mas posicionados em locais diferentes e o sistema de renderização da engine faz os cálculos como se fosse um só. Isso vale não só para as peças 3D, mas também as texturas e materiais alocadas à ela.

As questões de processamento gráfico e funcionamento do jogo são relativas e dependem da máquina do usuário, para estes testes foi usado um computador com uma placa gráfica *GeForce* 1060 6GB.

O processamento da fase vai depender de todos os elementos que serão colocados no cenário, e ele pode passar por processos de otimização nos seus estágios finais do desenvolvimento do jogo *Crystal League* para criar uma estabilidade nas máquinas alvo do jogo. Esse estágio de otimização pode não ser aplicado a este projeto de conclusão de curso por questões de ferramentas ainda não obtidas pela equipe do jogo *Crystal League* e também pelo fluxo de desenvolvimento do projeto como um todo.

### 5.3 PREPRODUCTION BLUEPRINT

O processo do *Preproduction Blueprint* se desenvolve em algumas partes, onde cada parte se cria e pensa na temática do cenário e na sua funcionalidade de acordo com a experiência desejada.

### 5.3.1 Ideia

A ideia do cenário se define como um cenário para o modo de jogo *Battle Royale* dentro do contexto *Crystal League*; Desenvolvimento de uma fase jogável seguindo o universo de competição e torneio que o jogo *Crystal League* possui. O cenário deve ser grande e necessita de elementos modulares para seu desenvolvimento

### 5.3.2 Tema

O tema do cenário foi definido pela sua ambientação e localizações referência para seu desenvolvimento. O cenário foi criado pela equipe do torneio *Crystal League* para receber a competição *Battle Royale*. Como ambientação ele terá elementos de floresta e planícies e pequenos espaços e áreas específicos com um contexto mais moderno.

As localizações referência para o cenário são:

- Floresta;
- Vilas;
- Praças modernas.

Com a ambientação e as localizações referência definidas o tema pode ser descrito como “um ambiente criado pela equipe *Crystal League* que possui em sua maioria um espaço rural e algumas praças modernas de diferenciação.”

### 5.3.3 Propósito do Projeto

Os propósitos definidos para o desenvolvimento desse cenário para o jogo *Crystal League* se identificam um pouco com o propósito geral deste projeto de conclusão de curso. Deixando mais específico para o contexto do jogo, os propósitos são:

- Desenvolver uma fase com o modo de jogo *Battle Royale* para o jogo *Crystal League* aplicando seu contexto;
- Desenvolver o cenário com elementos modulares para agilizar o processo e reutilizar *assets*.
- Desenvolver o cenário através de um processo de iteração que analisa feedback dos usuários para melhorar o resultado.

Ainda como propósito, mesmo o cenário não ficando em seu estágio final no fim deste projeto de conclusão de curso, seu desenvolvimento irá valer como pesquisa de produção para vários outros cenários para o jogo *Crystal League*, tanto para o modo de jogo *Battle Royale* quanto outros modos de jogo.

#### 5.3.4 Características Diferenciais

As características do cenário se definem pela experiência que ele vai passar e as funcionalidades necessárias para ele conseguir passar essa experiência.

As principais a serem descritas são:

- Partidas rápidas com um número menor de jogadores (seguindo jogos referência);
- Narrativa de competição e olimpíadas do jogo *Crystal League*;
- Aplicação do gênero *Battle Royale* com as mecânicas no jogo *Crystal League* criando uma diferenciação da saturação do gênero no mercado;
- Aplicar elementos modulares para acelerar sua produção e tentando evitar repetição aparente para os usuários;
- Desenvolvimento através de um processo de iteração que busca analisar *feedback* dos usuários e modificar os elementos do cenário de acordo com esse *feedback*.

### 5.3.5 Referência Visual

Uma pesquisa de referências visuais ajuda na criação e concretização da ambientação que se busca adquirir. As referências pesquisadas são de cinco propósitos diferentes:

- Referências de Arquitetura;
- Referências de Meio Ambiente;
- Referências de Iluminação;
- Referências de *Props* (pequenos elementos);
- Referências de Estilo e Inspiração.

Na etapa de Definição Visual deste relatório as principais imagens das pesquisas vão ser mostradas junto com o primeiro formato de conceito do cenário.

### 5.3.6 História

Como história são desenvolvidos os pontos quem levaram à existência do cenário e para que ele serve dentro da narrativa do jogo.

“O ambiente foi criado pela equipe *Crystal League* para receber as competições *Battle Royale* do evento. Os participantes da competição entram no cenário e se enfrentam até que apenas um saia vencedor.”

Junto com o desenvolvimento da história cada um dos elementos do cenário devem ter um propósito para existir, tanto veículos para locomoção quanto monumentos de pura atração visual.

Para definir partes dessa história foi preciso identificar as ações possíveis que os jogadores podem fazer dentro do cenário e como esperamos que ele se desenvolva na partida.

### 5.3.7 Objetivos e Obstáculos

Os objetivos e obstáculos do cenário e do modo de jogo criam a experiência de jogo para os jogadores. Como objetivos foram definidos os seguintes pontos dentro da partida:

- Visualizar o cenário e descobrir onde começar a partida;
- Explorar o cenário atrás de veículos e itens de uso;
- Estar sempre dentro do ambiente permitido para não ser eliminado da partida;
- Derrotar outros jogadores;
- Ser o último jogador vivo.

Esses objetivos são basicamente as ações que o jogador precisa fazer para avançar na partida.

Como obstáculos foram definidos alguns elementos, tanto do cenário e do jogo como também com a interação de outros jogadores:

- Ataque de outros jogadores;
- Habilidade para acessar certos ambientes do cenário;
- Estratégias diferentes de acordo com o fluxo que a partida está tomando para determinado jogador;
- Ambiente jogável da fase diminui com o tempo.

### 5.3.8 Ponto Focal

Como pontos focais são definidos algumas partes do cenário que servem para guiar o jogador pelos seus objetivos e interesses. Os pontos focais foram definidos em três categorias:

- Primários – Três prédios colocados no cenário que definem os espaços principais dele, suas praças modernas. Esse ambiente possui o maior número de veículos e de itens de uso;

- Secundários – Vilas e grandes monumentos que servem primeiramente para guiar o jogador usando composição de visualização para os pontos primários, mas também servem para adquirir itens e veículos e ambientes de batalha;
- Terciários – Pequenos elementos de *level design* que podem proporcionar alguma vantagem ou desvantagem para os jogadores (rampas, aumentadores de velocidade, etc.).

### 5.3.9 Top Down Layout

Depois de ter todos esses conceitos definidos e analisados criou-se a planta baixa do cenário (*Top Down Layout*). Essa planta baixa (figura 47 e 48) vai ajudar a desenvolver o cenário dentro da produção 3D.

Figura 47 - Top Down Layout do Cenário

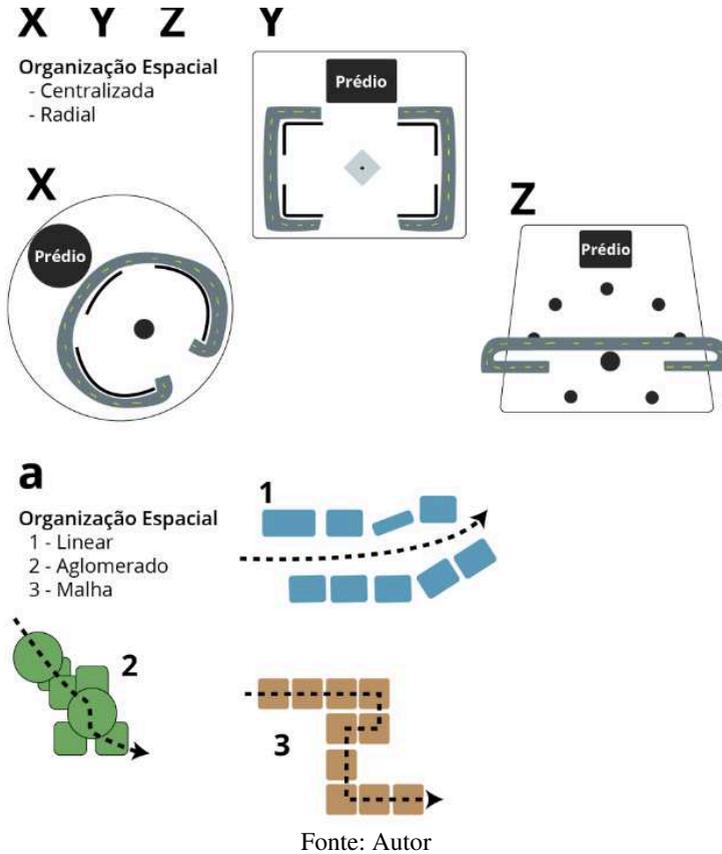


Fonte: Autor

Dentro do *Top Down Layout* foram definidos como se dá a organização espacial do cenário inteiro. As praças listadas são ambientes mais voltados ao combate e interação com outros jogadores; as vilas e obstáculos servem para

exploração do cenário e para guiar o jogador aos seus objetivos.

Figura 48 - Top Down Layout partes específicas

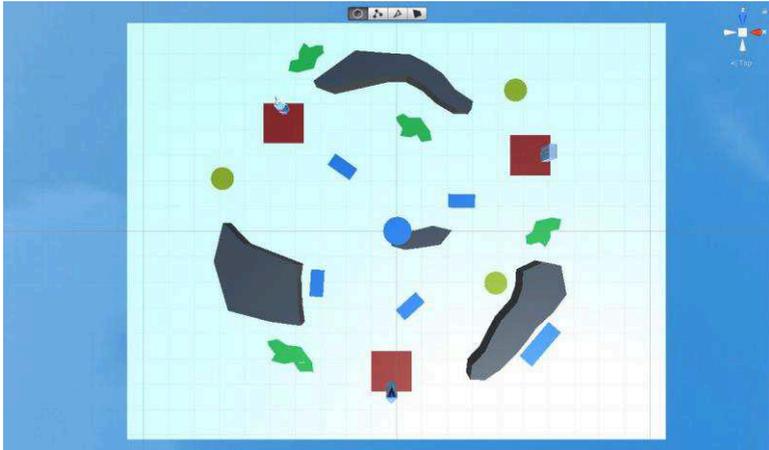


O *Top Down Layout* serve como principal referência para o desenvolvimento 3D do cenário, mas pode não ser a estrutura final a ser seguida, já que vários outros pontos podem ser percebidos durante o desenvolvimento 3D do cenário na sua blocagem.





*Figura 51 - Visão de Cima do Cenário em sua fase de Blocagem*



Fonte: Autor

*Figura 52 - Composição Visual para Guiar o jogador aos Pontos Focais*



Fonte: Autor

Na figura 52 é possível perceber que a composição visual cria uma linha de ação que leva a um dos pontos focais do cenário. Isso ajuda o jogador a descobrir objetivos e a explorar o cenário. Os outros pontos focais também se desenvolvem assim, isso se dá por conta do esquema triangular que as praças principais criam na composição do cenário e como os grandes obstáculos criam essa linha de ação para as praças.

## 5.5 ASSETS DO PROJETO CRYSTAL LEAGUE

O projeto *Crystal League* é grande e possui várias fases em desenvolvimento, cada uma com temática diferente e dentro disso ele possui um banco de *assets* grande. Esses *assets* irão ser usados em grande parte do cenário a ser desenvolvido nesse projeto. A parte artística a ser desenvolvida pelo autor são as casas modulares e o funcionamento do seu sistema modular junto com a aplicação dentro do cenário.

Os *assets* consistem em:

- Prédios;
- Árvores;
- Arbustos;
- Texturas e Materiais;
- Ruas modulares;
- *Props* de cenário em geral.

## 5.6 PLANEJAMENTO MODULAR

Para desenvolver o sistema das casas modulares e todas as suas peças e funcionamento precisa-se fazer um planejamento para a produção dessas peças, fazendo-as funcionar umas com as outras.

A primeira parte do planejamento acontece definindo cada tipo de peça e suas funcionalidades. Na figura 53 mostra como este projeto pretende seguir com o desenvolvimento na funcionalidade de cada peça aplicando

um nível de hierarquia para elas, de acordo com a estrutura do elemento modular.

*Figura 53 - Níveis de Hierarquia do Sistema Modular*

### **Níveis de Hierarquia**

#### **1 - Estrutura**

- a. Parede
- b. Chão
- c. Teto / Telhado
- d. Conexões chão / teto
- e. Detalhes parede

#### **2 - Funcionamento**

- a. Muros / Cercas
- b. Portas / Entradas
- c. Janelas
- d. Escadas
- e. Acessos (ponte, arco etc.)

#### **3 - Props**

- a. Monumentos
- b. Totens
- c. Rampas
- d. Chunk de Pedra / Madeira
- e. Poço
- f. Barril

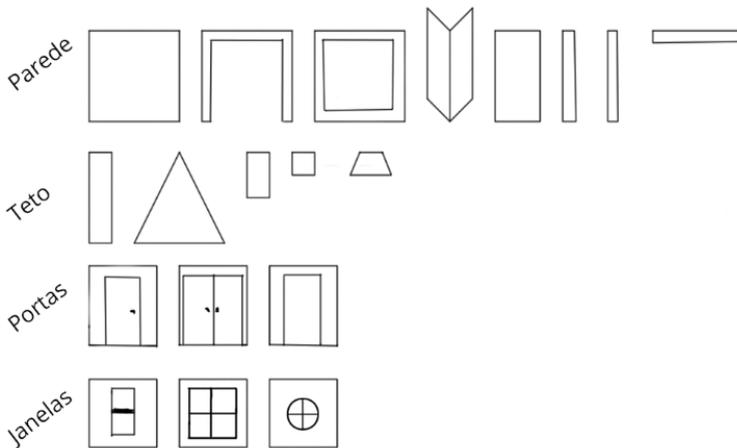
Fonte: Autor

Essa hierarquia é levada em consideração no desenvolvimento das peças 3D, pela importância de cada uma e objetivo no resultado. Os elementos modulares do cenário serão casas que criarão vida às vilas e seus espaços no cenário.

Após a definição dos níveis de hierarquia da estrutura dos elementos criou-se os conceitos (figura 54) das peças modulares a serem desenvolvidas, pensando no seu funcionamento no sistema.

Figura 54 - Conceito Peças Modulares

## CASAS E VILAS (1º & 2º)

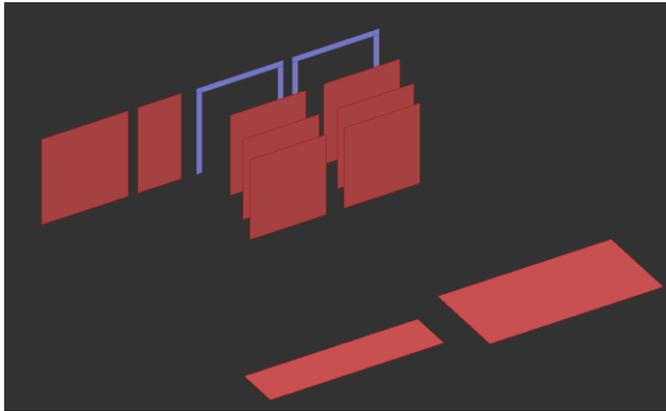


Fonte: Autor

Esses conceitos serviram para definir os tipos de variação que os elementos das casas podem ter e o funcionamento da estrutura para seu desenvolvimento mais a frente. Também não são o resultado final das peças, já que modificações podem aparecer para o melhor funcionamento do sistema e dos elementos.

Os primeiros modelos 3D a serem feitos para teste e aplicação de montagem dentro da *Unity 3D* foram os da figura 55, que são dos dois primeiros níveis de hierarquia (paredes, portas, janelas, conexões, etc.).

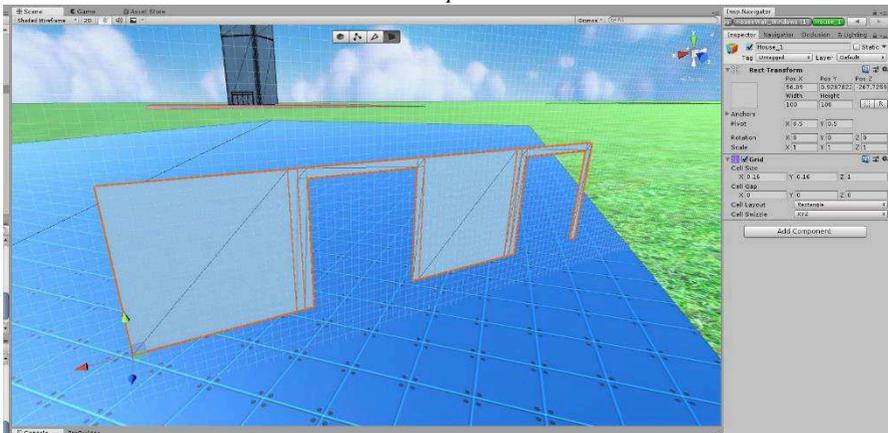
*Figura 55 - Primeira versão das peças 3D modulares*



Fonte: Autor

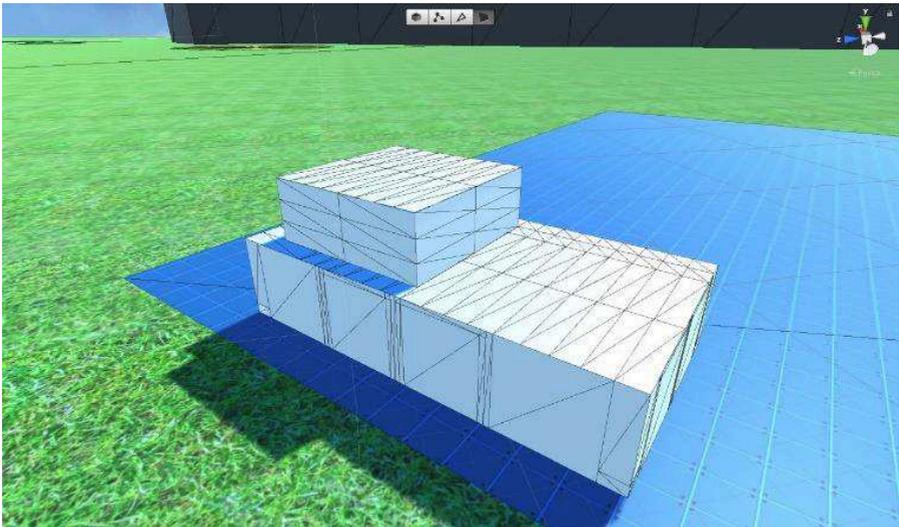
Com o teste a aplicação dentro da *Unity 3D* foi possível montar uma primeira versão de uma das casas modulares, como nas figuras 56 e 57.

*Figura 56 - Funcionamento das peças modulares de acordo com Pivô, Grid, Escala e hierarquia*



Fonte: Autor

*Figura 57 - Primeiro modelo de casa modular montada*



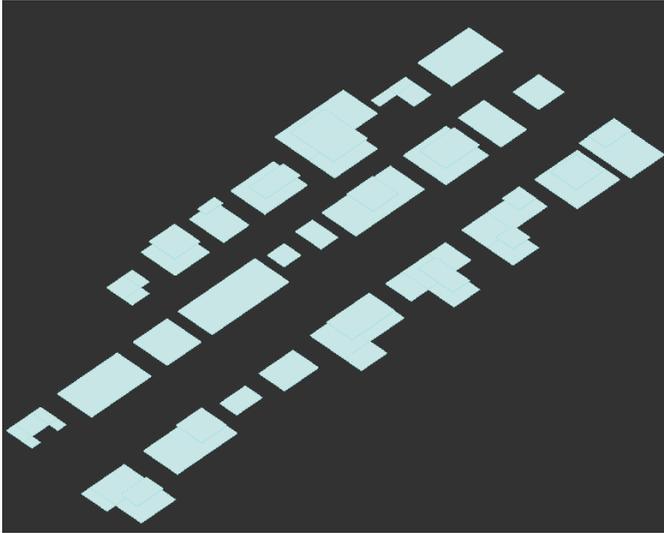
Fonte: Autor

Na figura 57, é possível perceber que o número de polígonos é muito elevado, mais que a sua necessidade. A parte superior da casa não funcionou com o sistema modular montado. Também, uma das peças criou uma dificuldade na montagem pois necessita que exista uma simetria nas peças para que o sistema funcione corretamente.

A solução para esses problemas acima foram a eliminação da peça dos cantos, que necessitava de uma simetria na casa e também a criação de plantas baixas para as casas, que facilita a montagem delas e também permite a criação de conceito de funcionalidade das casas antes de chegar ao 3D.

Pensando no funcionamento de todas as casas e suas funcionalidades e também variações que podem existir, foram criadas 26 plantas baixas de casas para serem montadas, como mostra a imagem abaixo.

*Figura 58 - Plantas Baixas das casas modulares*



Fonte: Autor

As plantas baixas (figura 58) consistem tanto no chão de cada casa como também um possível segundo andar para criar verticalidade ao cenário. Como resultado desse processo o sistema modular funcionou perfeitamente e a montagem das casas foi fácil (figura 59).

*Figura 59 - Casas modulares montadas*

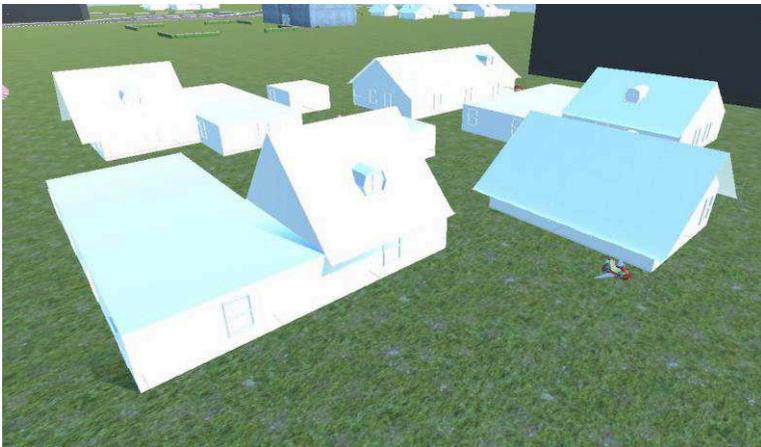


Fonte: Autor

A partir daqui o processo de iteração começou e ajustes e melhorias começaram a aparecer para criar um melhor resultado no cenário. Um desses ajustes foi a necessidade de criar telhados para as casas e diminuir essa questão dos seus segundos andares. Isso se deu pelo motivo de que o visual das casas não estava casando com o conceito que foi trabalho até agora sobre as vilas e o cenário.

Junto com a aplicação dos telhados um outro elemento de variação foi criado para fazer parte das casas, uma janela que fica nos telhados. Resultado na figura 60.

*Figura 60 - Casas modulares já detalhadas*

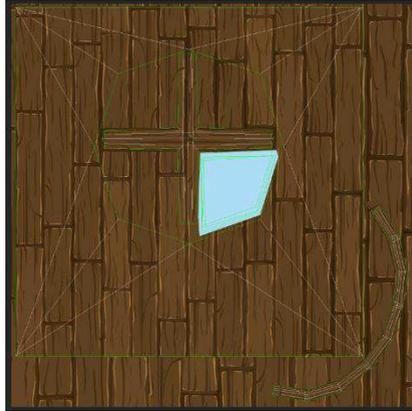


Fonte: Autor

A partir deste ponto todas as peças modulares já possuem seus últimos detalhes no modelo 3D, a continuação do processo se dá no desenvolvimento das texturas e materiais para cada elemento das casas.

A imagem abaixo consta como exemplo um dos mapas de textura para uma das peças modulares das casas, todas as outras peças tem um mapeamento parecido.

Figura 61 - Textura de uma das peças modulares



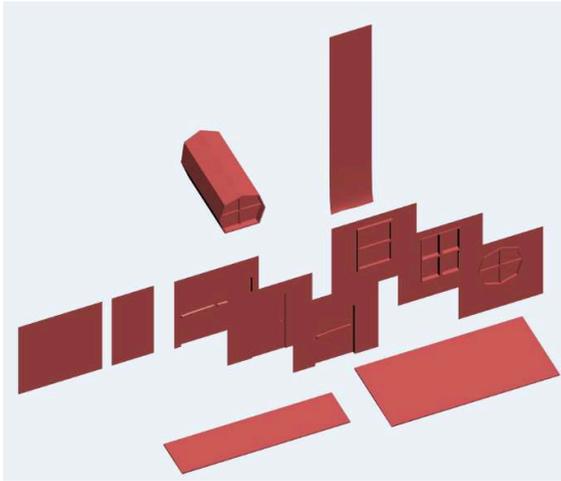
Fonte: Autor

A figura 61 mostra o mapeamento e texturização se deu pela organização de uma textura infinita (onde a textura tem uma padronagem que pode se repetir infinitamente sem percepção de mudanças) para ser aplicada nas casas. A manipulação dos elementos dessa textura em cima do mapeamento dos modelos 3D resultaram no funcionamento da texturização no sistema modular sem erros visíveis.

O resultado do sistema modular consiste na possível criação de inúmeras casas diferentes, assimétricas ou simétricas seguindo o contexto visual do jogo *Crystal League*.

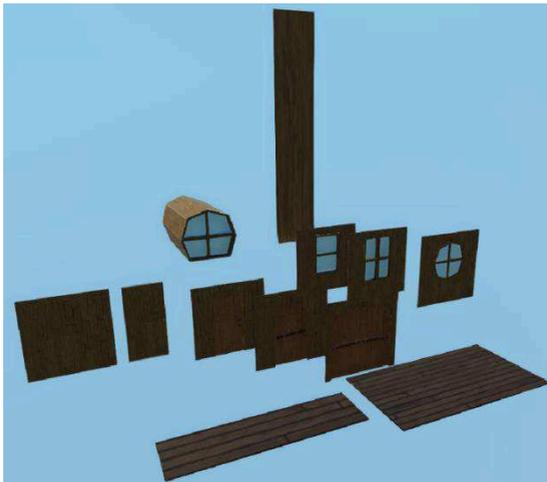
Abaixo estão as figuras 62,63 e 64 com o resultado tanto das peças 3D e sua evolução com as texturas quanto casas 3D montadas especificamente para exemplificar as possíveis diferenças de criação.

*Figura 62 - Modelos 3D no programa 3DS Max*



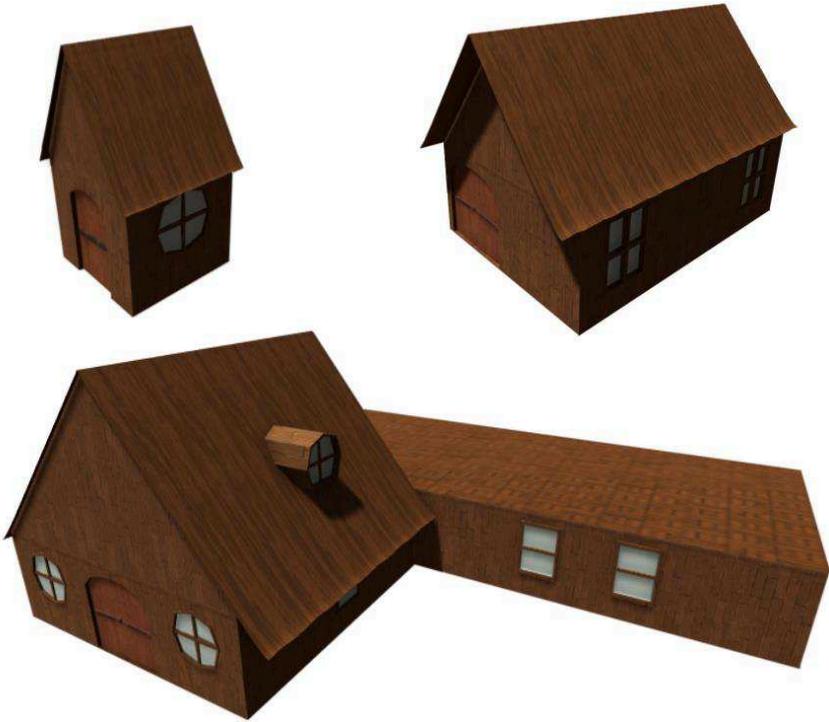
Fonte: Autor

*Figura 63 - Modelos 3D no programa Unity 3D já com textura e material*



Fonte: Autor

Figura 64 - Exemplo de variação possível nas Casas modulares



Fonte: Autor

## 5.7 PLANEJAMENTO DE ITERAÇÃO E TESTES

Para os processos de iteração foram definidos alguns *playtestings* a serem feitos, tanto internamente ou com entidades parceiras em alguns momentos. O desenvolvimento do cenário se deu pelo processo de iteração, onde através de feedbacks e análise de usabilidade dos usuários mudanças são identificadas para melhorar o resultado da experiência.

Houveram três testes majoritários que funcionaram para o desenvolvimento deste projeto e vão ser descritos individualmente.

### 5.7.1 Playtesting 1 e Iteração

O *playtesting* 1 aconteceu primeiramente de uma forma interna, onde o autor realizou testes para entender o tamanho do cenário, seus pontos principais e suas funcionalidades. Após esse teste interno houve uma interação com a *SC Games* de Florianópolis, onde os alunos e os professores do projeto testaram o jogo. Não apenas a fase em desenvolvimento deste projeto mas todo o jogo para melhor entrarem no contexto do jogo *Crystal League*.

Os maiores pontos de iteração e modificações que resultaram destes testes foram:

- As casas estavam com uma escala muito pequena em referência aos personagens, necessário aumentar sua escala;
- Pouca verticalidade no cenário, necessário criar momentos que permitem a verticalidade;
- O cenário como um todo ficou muito grande e houve pouco encontro com os personagens IA;
- Os espaços das vilas estavam grandes e muito “amontoados”, pouco suscetíveis a acesso;
- Poucos itens de uso pelo cenário, apenas em pontos chave do cenário eles não estão fazendo muita diferença;
- Criar um córrego no centro da fase.

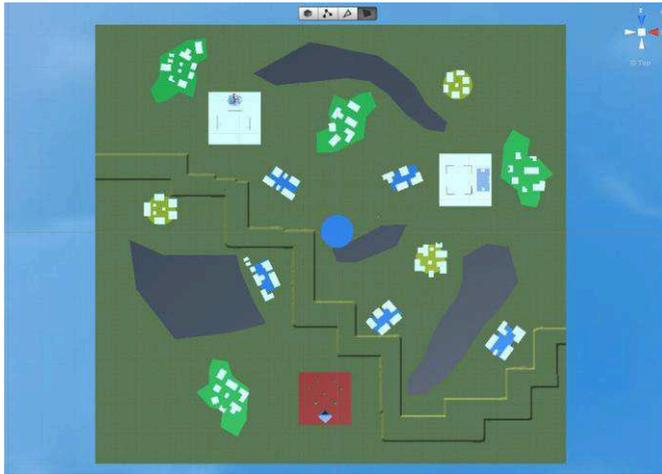
Não houve imagem no banco de imagens para mostrar o estado do cenário quando esses primeiros testes foram feitos.

Após esses primeiros testes o desenvolvimento continuou, com a modularidade das casas e o *level design* de todo o cenário em conjunto. As mudanças listadas foram aplicadas para melhorar a experiência e foi marcado um novo *playtesting*.

### 5.7.2 Playtesting 2 e Iteração

Esse segundo *playtesting* foi feito também em conjunto com a *SC Games* de Florianópolis, mas com um grupo diferente de pessoas. O estado em que o cenário ficou antes para o teste, com as mudanças anteriores aplicadas foi como na figura 65.

Figura 65 - Estado do cenário para o Playtesting 2



Fonte: Autor

Os teste ainda estavam sendo focados na estrutura da fase e no funcionamento dos espaços, levando pouco em conta a estética das casas modulares. O *feedback* destes teste resultaram na mudança das casas modulares para terem telhados.

As mudanças importantes listadas destes testes foram:

- Tamanho do cenário ficou interessante, as casas poderiam ser um pouco mais com o ambiente do cenário;
- Os espaços no interior das vilas estão pequenos e pouco acessíveis ainda, interessante aumentar o espaço interior delas;
- Quantidade ainda baixa de itens de uso e também é necessário uma melhor distribuição deles;
- O córrego no centro da fase pareceu muito estranho e está com uma estética que foge dos conceitos;
- Elementos de interação maiores e mais visíveis para acesso.

A partir deste *feedback* o cenário começou a ficar mais conciso na parte estética e sua produção. As casas foram detalhadas e os *assets* principais do projeto foram colocados para funcionar dentro do cenário. Estas mudanças foram todas estudadas e aplicadas para melhorar o cenário e o resultado é a imagem a seguir, que ficou como estado atual para o último teste antes da entrega do relatório deste projeto de conclusão de curso.

### 5.7.3 Playtesting 3 e Feedback

O estado atual para o terceiro *playtesting* foi o da figura 66. Todos os elementos anteriores foram trabalhados e principalmente a parte estética tanto das casas quanto do resto do cenário começou a tomar seus últimos passos.

Figura 66 - Estado do cenário antes do Playtesting 3



Fonte: Autor

Com a parte estética agora sendo muito bem aplicada e em seus passos finais, a questão de processamento do cenário começou a ser vista nas máquinas. Para a montagem evitando problemas de processamento peças começaram a ser evitadas e a estrutura se desenvolveu com o menor número de *assets* possíveis.

Este terceiro teste aconteceu na UFSC (Universidade Federal de Santa Catarina) junto com o grupo *G-Dev* de estudantes da UFSC. Foi possível perceber que o cenário ficou mais conciso e claro que as suas outras versões e os elementos de interação dentro dele começaram a ficar visíveis e acessíveis para os jogadores.

O cenário foi desenvolvido para que o modo de jogo *Battle Royale* aconteça nele. Por questões de produção as funcionalidades principais definidas para o modo de jogo e o cenário ainda não foram desenvolvidas pela equipe para que funcionem no cenário. Mas isso não evitou o questionamento e estudos para melhorias e testes para seguir com o processo de desenvolvimento.

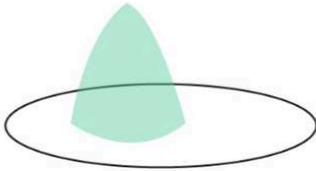
Os pontos mais importantes de *feedback* deste terceiro *playtesting* foram os seguintes:

- Raio da área inicial muito grande;
- Suavizar as curvas do córrego;
- Maior número de itens de uso;
- Maior número de itens de aceleração;
- Mini-mapa.

Analisando os *feedbacks* percebe-se que eles começam a ser tornar mais específicos que os anteriores e são para algumas áreas do cenário. Para a entrega do relatório o resultado final do cenário seguira das imagens a seguir, mas o processo de iteração e desenvolvimento da fase e do modo de jogo continuam, e para a etapa de Comunicação deste projeto e apresentação o cenário contará com protótipos da funcionalidade dos gêneros *Battle Royale*, que é o campo de força, definido pela figura 67.

*Figura 67 - Funcionalidade principal para o gênero Battle Royale*

## 1 - CAMPO DE FORÇA



Delimita a área jogável dentro de etapas;  
 Campo de força vai fechando para delimitar área da etapa;  
 Por baixo dos panos delimita toda a área jogável do mapa no começo da partida.

### **Variáveis Importantes**

- Quantidade de Etapas
- Duração de cada Etapa
- Velocidade que o campo fecha em cada Etapa

Fonte: Autor

Esta será a funcionalidade que será apresentada junto com o cenário no dia de apresentação deste projeto de conclusão de curso, ela estará em forma de protótipo para testes na funcionalidade de todo o cenário.

### 5.8 EXEMPLOS DO RESULTADO DO CENÁRIO

As figuras 68, 69, 70, 71, 72, 73, 74 e 75 a seguir mostram o resultado da aplicação dos conceitos e das metodologias usadas, junto com a narrativa da história.

*Figura 68 - Visão geral do cenário em perspectiva*



Fonte: Autor

*Figura 69 - Interior do cenário*



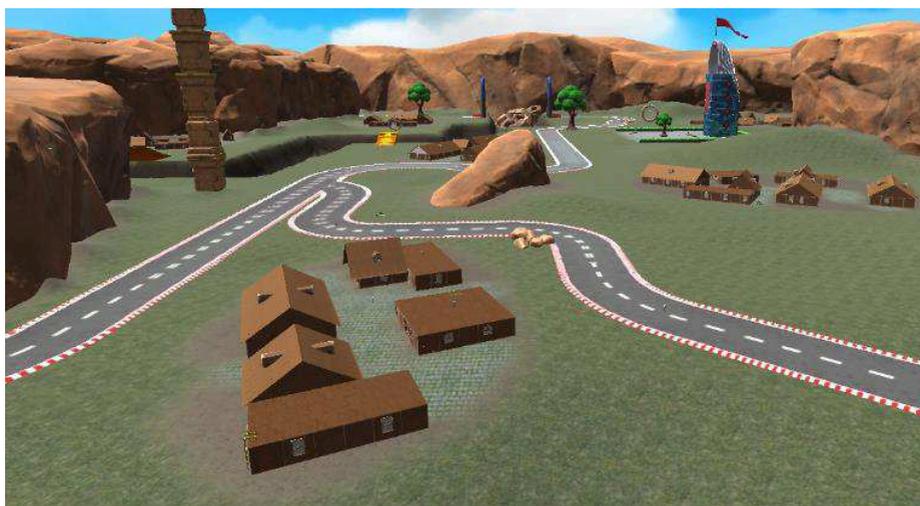
Fonte: Autor

*Figura 68 - Interior do cenário e elementos*



Fonte: Autor

*Figura 69 - Interior do Cenário e composição visual*



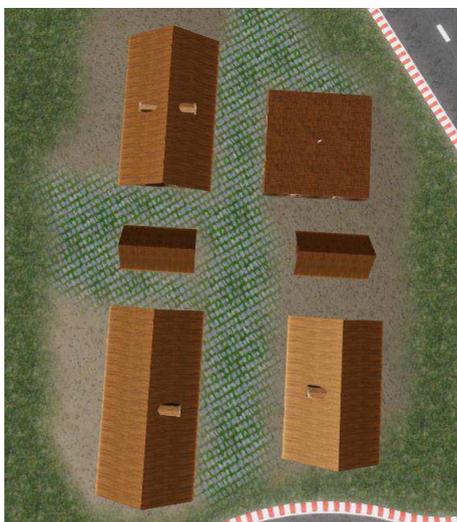
Fonte: Autor

*Figura 70 - Affordance de Rampear*



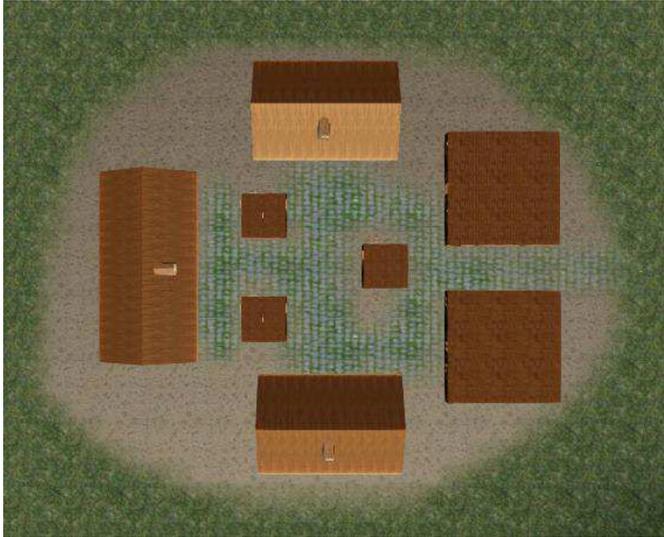
Fonte: Autor

*Figura 71 - Organização Espacial Linear*



Fonte: Autor

*Figura 72 - Organização Espacial em Malha*



Fonte: Autor

*Figura 73 - Composição Visual e Ponto Focal*



Fonte: Autor

## 6 COMUNICAÇÃO

### 6.1 COMUNICAÇÃO E APRESENTAÇÃO

A comunicação se dará pela apresentação do Projeto de Conclusão de Curso 2, mostrando o funcionamento do cenário dentro do jogo *Crystal League*, junto com o relatório de produção de todo o projeto.

### 6.2 CONCLUSÃO E RESULTADO

Através deste projeto de conclusão de curso o autor pode adquirir e aplicar conhecimentos das áreas do design de animação e de jogos diretamente em um cenário profissional.

Mesmo o resultado do cenário final apresentado neste relatório não sendo o definitivo para o jogo, todas as pesquisas e testes valeram imensamente para a concretização do conceito do jogo e do cenário e seguirão junto no desenvolvimento do jogo *Crystal League*, assim que for possível e viável a aplicação do cenário e do modo de jogo, eles serão aplicados junto ao conteúdo que será lançado oficialmente.

As principais dificuldades encontradas no desenvolvimento deste projeto foram as casas modulares, na sua função do cenário, estética e produção. Para que sua função continuasse funcional, houveram algumas mudanças nos seus conceitos e formas durante os testes de iteração, o que já ajuda a definir o quão importante são os testes.

Não só o cenário, o sistema modular e as casas modulares vão poder ser reaproveitadas para outros cenários que o jogo *Crystal League* precisar e serem viáveis de aplicação.

## 7 REFERÊNCIAS

NEVES, André et al (Org.). **BRUCE ARCHER: Método Sistemático para Designers**. Pernambuco: Centro de Artes e Comunicação Ddesign | Pós-graduação, 2017. 10 p. Disponível em: <<http://www.vigha.com/wp-content/uploads/2017/06/Artigo-Bruce-Archer-final.pdf>>. Acesso em: 19 mar. 2018.

BARTLE, Richard. **HEARTS, CLUBS, DIAMONDS, SPADES: PLAYERS WHO SUIT MUDDS**. 1996. Disponível em: <<http://mud.co.uk/richard/hcdds.htm>>. Acesso em: 19 jun. 2018.

CHING, Francis D. K.. **Arquitetura, Forma, Espaço e Ordem**. São Paulo: Martins Fontes, 1998. 399 p.

EVERYTHING I Learned About Level Design I Learned from Disneyland. San Francisco, 2009. (59 min.), son., color. Disponível em: <<http://www.gdcvault.com/play/1305/Everything-I-Learned-About-Level>>. Acesso em: 29 maio 2018.

FALLOUT 4 Modular Level Design. San Francisco, GDC. (62 min.), son., color. Disponível em: <<https://www.gdcvault.com/play/1023202/-Fallout-4-s-Modular>>. Acesso em: 01 jun. 2018.

GALUZIN, Alex. **Preproduction Blueprint: How to Plan Game Environments and Level Designs**. 2. ed. Usa: Create Space, 2016.

GEORGE A. MILLER. **The Magical Number Seven, Plus or Minus Two: Some Limits on Our Capacity for Processing Information**. 1955. Disponível em: <<http://www.psych.utoronto.ca/users/peterson/psy43>>

0s2001/Miller GA Magical Seven Psych Review 1955.pd>. Acesso em: 13 jun. 2018.

KREMERS, Rudolf. **Level Design: Concept, Theory & Practice**. Estados Unidos: A K Peters, Ltd, 2009.

LAZZARO, Nicole. **Why We Play Games: Four Keys to More Emotion in Player Experiences**. 2004. Disponível em: <[http://twvideo01.ubm-us.net/o1/vault/gdc04/slides/why\\_we\\_play\\_games.pd](http://twvideo01.ubm-us.net/o1/vault/gdc04/slides/why_we_play_games.pd)>. Acesso em: 19 jun. 2018.

LEVEL Design in a Day: How We Used Iterative Level Design to Ship Skyrim and Fallout 3. San Francisco, 2014. (58 min.), son., color. Disponível em: <<https://www.youtube.com/watch?v=PhW8CY8XkFg&t=8s>>. Acesso em: 31 maio 2018.

ROGERS, Scott. **Level Up: Um guia para o design de grandes jogos**. São Paulo: Blucher, 2012.

ROSS, Bobby. **Visual Guide: Multiplayer Level Design**. 2015. Disponível em: <<http://bobbyross.com/library/mpleveldesign>>. Acesso em: 29 maio 2018.

SCHATZ, Emilia. **Defining Environment Language for Video Games**. 2017. Disponível em: <<https://80.lv/articles/defining-environment-language-for-video-games/>>. Acesso em: 19 jun. 2018.

SCHELL, Jesse. **The Art of Game Design: A Book of Lenses**. Estados Unidos: Elsevier, 2010.

SEQUELITIS - Mega Man Classic vs. Mega Man X. [s.i.]: Egoraptor, 2011. Son., color. Disponível em:

<<https://www.youtube.com/watch?v=8FpigqfcvIM>>.  
Acesso em: 19 jun. 2018.

WHY Nathan Drake Doesn't Need a Compass | Game  
Maker's Toolkit. [s.i.]: Mark Brown, 2015. (8 min.),  
son., color. Disponível em:  
<[https://www.youtube.com/watch?v=k70\\_jvVOcG0](https://www.youtube.com/watch?v=k70_jvVOcG0)>.  
Acesso em: 13 jun. 2018.