

Juliano de Souza Pinto Pereira

**DESENVOLVIMENTO DE MÉTODO PARA JOGOS DIGITAIS:
CONSTRUÇÃO DO PROTÓTIPO DE BREAKFAST
DETECTIVES**

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. William Machado de
Andrade

Florianópolis
2018

Ficha de identificação da obra elaborada pelo autor,
através do Programa de Geração Automática da Biblioteca Universitária da UFSC.

Pereira, Juliano de Souza Pinto
Desenvolvimento de Método para Jogos Digitais :
Construção do Protótipo de Breakfast Detectives /
Juliano de Souza Pinto Pereira ; orientador,
William Machado de Andrade, 2018.
50 p.

Trabalho de Conclusão de Curso (graduação) -
Universidade Federal de Santa Catarina, Centro de
Comunicação e Expressão, Graduação em Design,
Florianópolis, 2018.

Inclui referências.

1. Design. 2. Desenvolvimento de Jogos Digitais.
3. Metodologia. 4. Gráficos 2D. I. Machado de
Andrade, William . II. Universidade Federal de
Santa Catarina. Graduação em Design. III. Título.

Juliano de Souza Pinto Pereira

**DESENVOLVIMENTO DE MÉTODO PARA JOGOS DIGITAIS:
CONSTRUÇÃO DO PROTÓTIPO DE BREAKFAST
DETECTIVES**

Este Projeto de Conclusão de Curso foi julgado adequado para obtenção do Título de “Bacharel em Design”, e aprovado em sua forma final pelo Programa de Graduação da Universidade Federal de Santa Catarina.

Florianópolis, 5 de Dezembro de 2018.

Prof.^a Dr.^a Marília Matos Gonçalves
Coordenadora do Curso

Banca Examinadora:

Prof. William Machado de Andrade
Orientador
Universidade Federal de Santa Catarina

Prof.^a Dr.^a Mônica Stein
Universidade Federal de Santa Catarina

Prof. Gustavo Boehs
Universidade Federal de Santa Catarina

Este trabalho é dedicado a Guilherme,
Ridio e Lhaion, parceiros do Curso de
Graduação de Design de Jogos.

AGRADECIMENTOS

Meus sinceros agradecimentos ao Prof. William Machado de Andrade pela orientação e paciência. Aos meus pais e amigos pelo apoio e especialmente à amiga Denise pelas dicas de programação.

RESUMO

Este relatório de Projeto de Conclusão de Curso se propõe a apresentar um método de desenvolvimento de jogos digitais voltado ao desenvolvedor brasileiro independente, sem a necessidade de programação extensa e de um alto orçamento. O método é validado com o desenvolvimento do protótipo de *Breakfast Detectives*, um jogo 2D *point-and-click*. Para criá-lo, foram pesquisadas e analisadas as metodologias já utilizadas na indústria, e em relação à natureza do jogo, foi estudada a aplicação dos gráficos 2D, assim como a história do gênero *point-and-click* e a viabilidade da criação de um jogo desse estilo no mercado atual.

Palavras-chave: Desenvolvimento de Jogos Digitais. Metodologia. Gráficos 2D.

ABSTRACT

This graduation project report proposes to present a method of digital game development aimed at the independent Brazilian developer, without the need for extensive programming and a high budget. The methodology is validated with the development of the prototype of *Breakfast Detectives*, a 2D point-and-click game. In order to create it, the methodologies already utilized in the industry were researched and analyzed, and in relation to the nature of the game, the application of the 2D graphics was studied as well as the history of the point-and-click genre and the viability of the creation of a game of that style in the current market.

Keywords: Digital Game Development. Methodology. 2D Graphics.

LISTA DE FIGURAS

Figura 1 – Exemplo de um <i>blueprint</i> na Unreal Engine 4	2
Figura 2 – <i>Walk cycle</i> do personagem criado no Photoshop.....	3
Figura 3 – Captura de tela de <i>Breakfast Detectives</i> , jogador encontra uma pista.....	6
Figura 4 – Captura de tela de <i>Breakfast Detectives</i> , diálogo com NPC.....	6
Figura 5 – Captura de tela de <i>Breakfast Detectives</i> , jogador perto dos cavaletes e da moldura	7
Figura 6 – Metodologia de desenvolvimento de jogos de Rogers	9
Figura 7 – Diagrama do processo iterativo	10
Figura 8 – Etapas do desenvolvimento	12
Figura 9 – Modelo para <i>game design</i> iterativo: <i>playtest</i> , avaliação, revisão.....	13
Figura 10 – Processo de desenvolvimento iterativo de três estágios	14
Figura 11 – Interface inicial do protótipo versão TIC.....	23
Figura 12 – Interface inicial do protótipo versão PCC.....	23
Figura 13 – Jogador em diálogo com o NPC Det. Knife	24
Figura 14 – Jogador observando o inventário de objetos coletados.....	25
Figura 15 – Captura de tela do <i>gameplay</i> de <i>Breakfast Detectives</i> versão TIC.....	26
Figura 16 – <i>Sprites</i> do <i>walk cycle</i> baixo do Det. B. Read, o personagem do jogador	26
Figura 17 – <i>Sprites idle</i> do NPC Dr. Hazel.....	27
Figura 18 – <i>Sprites idle</i> do NPC Ms. Carrot.....	27
Figura 19 – Jogador na parte inferior do cenário, onde apareceriam as ondas do mar.....	28
Figura 20 – Jogador acabando de ganhar a <i>quest</i> e aparecendo os objetivos no lado direito da tela	29
Figura 21 – Jogador acessando a interface <i>in-game</i>	29
Figura 22 – Jogador só consegue iniciar diálogo com o NPC Det. Knife....	30
Figura 23 – Mensagem de <i>feedback</i> após coletar todas as pistas.....	31
Figura 24 – Interface para salvar o jogo aberta durante o <i>gameplay</i>	32
Figura 25 – Captura de tela da cena inicial do curta-metragem de animação <i>Breakfast Detectives</i>	33

LISTA DE TABELAS

Tabela 1 – Comparação das versões do TIC e do PCC de <i>Breakfast Detectives</i>	22
--	----

SUMÁRIO

1	INTRODUÇÃO.....	1
1.1	OBJETIVOS	1
1.1.1	Objetivo Geral.....	1
1.1.2	Objetivos Específicos	2
1.1.3	Justificativa	3
1.1.4	Delimitações	4
2	FUNDAMENTAÇÃO TEÓRICA	5
2.1	ESTUDO ESTÉTICO E DE LINGUAGEM	5
2.1.1	Construção do Método	8
3	PESQUISA.....	17
4	DESENVOLVIMENTO	19
4.1	MÉTODO	19
4.1.1	Estado Inicial e Desenvolvimento do Protótipo	21
5	CONCLUSÃO	35
	REFERÊNCIAS	36
	ANEXO A - Registro de Desenvolvimento do protótipo versão PCC.....	37
	GLOSSÁRIO DE TERMOS EM INGLÊS	38

1 INTRODUÇÃO

Na atualidade, percebe-se que há possibilidades gratuitas de construção de jogos eletrônicos que possuem adequada pré-configuração e assim permitem realização de produtos sem obrigatória necessidade de grande (e custoso) corpo de programadores voltado às técnicas 3D. Assim sendo, como utilizá-las em sentido comercial para a estética 2D sem descaracterizá-las e ainda assim entregar um produto mercadologicamente viável? Como resposta e solução à essa questão, tem-se a criação de um método para o desenvolvimento de jogos digitais, validada por meio da construção de um jogo 2D em plataforma gratuita e imersiva 3D para narrativas interativas de entretenimento com baixo custo de produção e potencial de comercialização, no sentido de abrir possibilidades criativas para desenvolvedores independentes de exploração de demanda latente na indústria de jogos eletrônicos.

1.1 OBJETIVOS

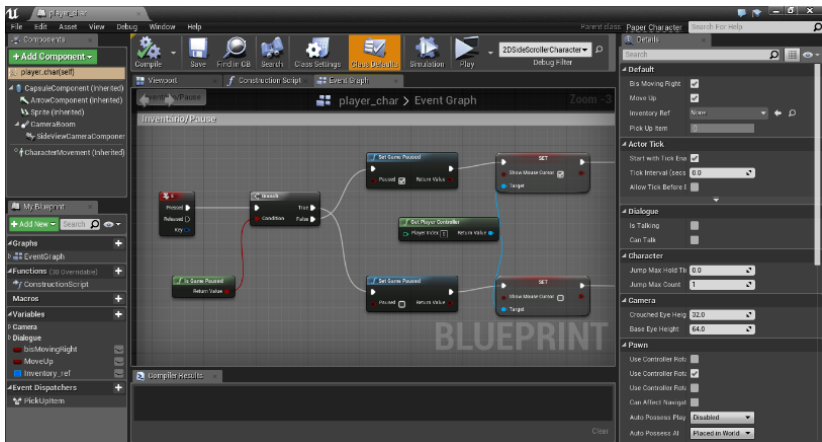
1.1.1 Objetivo Geral

O objetivo deste Projeto de Conclusão de Curso (PCC) é desenvolver um método de desenvolvimento de jogos digitais, voltado ao desenvolvedor independente brasileiro não necessariamente com experiência em programação, e sim com perfil de artista ou *designer*. Este método é criado a partir de metodologias já existentes na indústria dos jogos e então validado pela criação de um protótipo de jogo *point-and-click* 2D com o motor de jogo (em inglês, “*engine*”) Unreal Engine 4, contendo no mínimo uma fase (neste caso, um cenário explorável) totalmente funcional, com o mínimo de custo possível e sem programação extensa. Isto foi feito a partir de um protótipo de jogo já existente chamado *Breakfast Detectives*, que foi desenvolvido como parte do Trabalho de Iniciação Científica (TIC - semelhante ao PCC do curso de Design da Universidade Federal de Santa Catarina) do autor no Curso de Graduação de Design de Jogos e Entretenimento Digital no Campus Florianópolis da Universidade do Vale do Itajaí (UNIVALI). Não obstante, muitas das características planejadas para o protótipo durante a realização do TIC não foram desenvolvidas (um exemplo é que não há animação no jogo), portanto é almejado concretizá-las neste PCC.

1.1.2 Objetivos Específicos

Como citado anteriormente, este PCC visa demonstrar um método de baixo custo de desenvolvimento de jogos 2D para artistas que possuem pouco ou nenhum conhecimento em programação. Os recursos tecnológicos escolhidos para facilitar este processo são a *engine* Unreal Engine 4, por ser gratuita (tanto para estudantes como para profissionais), por ter o recurso dos *blueprints* (ferramenta de *scripting* visual, que facilita a programação, Figura 1) e o sistema Paper2D (conjunto de recursos para criação de jogos 2D dentro do programa); e a assinatura do Adobe Creative Cloud, um conjunto de programas e serviços da Adobe Systems que dá acesso aos seus assinantes uma coleção de *softwares* de edição gráfica (como o Photoshop, utilizado para criar as partes gráficas do jogo, Figura 2), de vídeo, de áudio, entre outros, tudo por um preço mensal ou anual razoável.

Figura 1 – Exemplo de um *blueprint* na Unreal Engine 4.



Fonte: Desenvolvido pelo autor.

Figura 2 – *Walk cycle* do personagem criado no Photoshop.



Fonte: Desenvolvido pelo autor.

Tendo isso determinado, cabe agora observar o que de fato se pretende realizar dentro deste objetivo, durante a aplicação do método no protótipo. É visado criar animações (de menu/interface, personagens, cenário, *cutscene*, etc) para o protótipo do jogo; aplicar um sistema de *quest* (mecânica de missões, que envolve também a criação de um inventário totalmente funcional), para que haja objetivos para o jogador cumprir e que as mecânicas do protótipo se relacionem organicamente; criar interface adequada à identidade visual do jogo (tanto a do menu principal como a *in-game*), incluindo a aplicação de sons e música (para fins de feedback auditivo e ambientação).

1.1.3 Justificativa

É esperado que o método apresentado neste PCC possa servir de auxílio a acadêmicos da UFSC que desejam começar seus projetos de jogos digitais tendo em vista as dificuldades de custo e energia requeridos, e que assim possam crescer em experiência e desenvolver novas técnicas que ajudem o desenvolvimento da indústria brasileira de jogos, assim como aproveitar oportunidades de publicação e monetização no crescente mercado de jogos independentes e atinjam o padrão internacional.

1.1.4 Delimitações

Para explicitar os limites deste PCC, é necessário afirmar que: não foi feito um jogo inteiro (um produto comercializável) na duração de desenvolvimento do PCC; o PCC não foi um tutorial de desenvolvimento de jogos 2D na Unreal; não foi abordado nada sobre Realidade Virtual (VR); e não foi demonstrado como fazer animações para jogos 2D, temas estes abordados em outros projetos de conclusão. Por último, vale lembrar que apesar da Unreal Engine 4 ser mais voltada para a produção de jogos 3D, no presente projeto serão utilizados recursos da *engine* que possibilitam a criação de jogos 2D (o Paper2D, mencionado anteriormente), e as ferramentas normalmente usadas para jogos 3D serão adaptadas para funcionar em 2D (como por exemplo a câmera, ajustada frontalmente para a visualização dos gráficos bidimensionais do jogo).

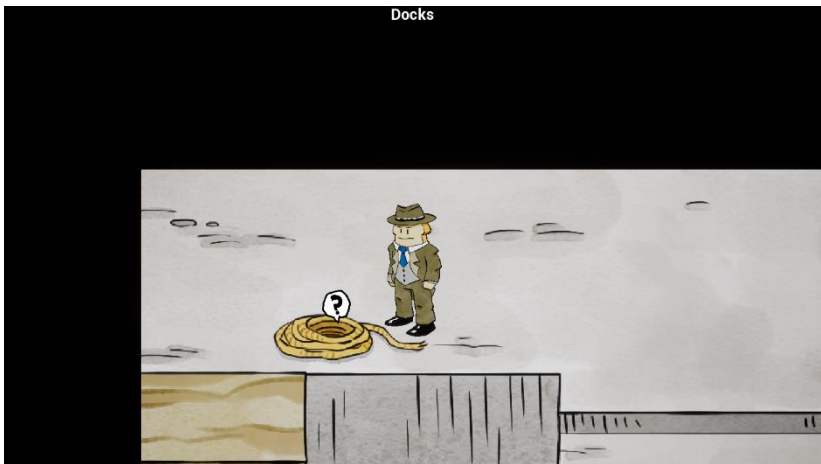
2 FUNDAMENTAÇÃO TEÓRICA

2.1 ESTUDO ESTÉTICO E DE LINGUAGEM

Uma característica que é importante para que o jogo tenha sucesso como um produto é a eficiência em que transmite suas informações ao usuário (jogador), como estímulo ou resposta a uma ação do jogador (*feedback*). No jogo, essas informações são principalmente transmitidas por meios visuais e auditivos. Voltando à atenção aos meios visuais (gráficos 2D) do jogo *Breakfast Detectives* trabalhado neste PCC, é possível fazer um estudo estético e de linguagem na intenção de atingir a melhor legibilidade possível, com apoio dos livros *Arte e Percepção Visual* de Arnheim R.(2014) e *A Imagem* de Aumont J.(2011).

Assim sendo, pode-se dizer que a maior parte da aqui chamada “superfície” (tela ocupada pela interface, personagens e cenário) é ocupada pelo cenário, que tem seu papel na imersão proporcionada pelo jogo, que é passar a impressão de um universo “vivo” e real dentro do contexto do jogo, não somente um coadjuvante à ação do jogador. Para isso há os estímulos visuais com durações variadas que devido ao sentido espacial da visão, são percebidos pelo jogador (AUMONT, p.26), como animações de elementos do cenário (por exemplo, uma goteira dentro de um cômodo ou uma luminária piscando com defeito). No atual estado do protótipo de *Breakfast Detectives*, não há animações de cenário implementadas. Porém animações do cenário não são a única fonte de estímulo visual para o jogador, por exemplo em *Breakfast Detectives* quando o jogador chega perto de um ponto de interesse ou uma possível pista, a interface mostra um símbolo com uma interrogação (“?”) sobre o local (Figura 3), e se o jogador se afastar sem ter interagido (ou coletar a pista), o estímulo some.

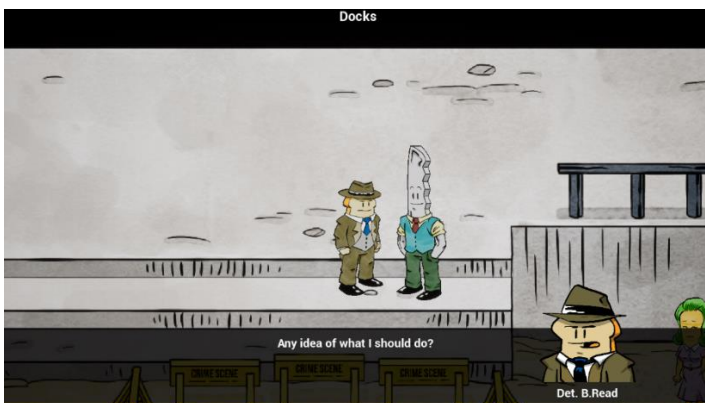
Figura 3 – Captura de tela de *Breakfast Detectives*, jogador encontra uma pista.



Fonte: Desenvolvido pelo autor.

Outro exemplo de *feedback* visual com variação rápida é durante um diálogo no jogo entre o personagem do jogador e um NPC (sigla em inglês para *Non-Playable Character*, personagem não-jogável), onde as falas dos personagens vão aparecendo uma após a outra na tela, junto do “retrato” de quem está falando no momento (Figura 4).

Figura 4 – Captura de tela de *Breakfast Detectives*, diálogo com NPC.



Fonte: Desenvolvido pelo autor.

É comum em jogos do gênero *point-and-click* a movimentação do personagem do jogador se dar por meio de o jogador clicar em um ponto sem obstáculos no cenário e o personagem se mover até lá. Porém, em *Breakfast Detectives*, é almejado que o jogador ativamente explore e investigue o cenário “manualmente”, ou seja, movimentando o personagem diretamente em tempo real e não esperando que ele execute uma ação conforme o comando. E, sendo que a percepção de movimento é em parte relacional, ela é facilitada por pontos fixos de referência (AUMONT, p.46), não aparentando que é o cenário que se move enquanto o personagem está parado, por exemplo.

Com o intuito de ilustrar exemplos de pontos fixos de referência no atual protótipo, a moldura preta estática delimita o espaço explorável pelo jogador na presente fase (Figura 3), assim como as faixas de polícia (cavaletes amarelos) que separam a cena do crime do resto do cenário e também delimitam o espaço explorável (Figura 5).

Figura 5 – Captura de tela de *Breakfast Detectives*, o jogador perto dos cavaletes e da moldura.



Fonte: Desenvolvido pelo autor.

Ainda sobre a movimentação do personagem, podemos afirmar que os aspectos mais específicos do movimento, tais como direção e velocidade, são também percebidos de acordo com as condições que prevalecem no campo visual (ARNHEIM, p.374). Na prática (em *Breakfast Detectives*), a velocidade do personagem não muda

independente da direção, sempre em velocidade de caminhada, e a câmera segue o personagem na mesma velocidade (não há atraso). Foi escolhido ser desta maneira o movimento porque as situações do jogo não teriam ação diretamente (ao menos na parte do jogo que cabe o protótipo), e se houvesse uma velocidade maior que a de caminhada, seria limitada a contextos específicos.

2.1.1 Construção do Método

Na pesquisa bibliográfica para este PCC foram selecionados alguns livros sobre Design de Jogos (*Game Design*, em inglês), mais especificamente livros que demonstrem metodologias que servirão de base para a criação do método do PCC. Porém é constatado ser comum que livros de *game design* tendem a focar em conceitos, teoria e/ou programação, não envolvendo tanto a prática, como o livro *Design de Games - Uma Abordagem Prática* de Schuytema P.(2008). Esse livro aborda conceitos gerais que são essenciais para um *game designer*, mas não oferece um método no qual é possível saber por onde começar e como continuar um projeto de jogo. É compreensível, considerando que há certos aspectos do desenvolvimento de jogos onde se pode aprofundar ou dirigir mais o foco do livro, como na parte de gráficos, programação, som etc.

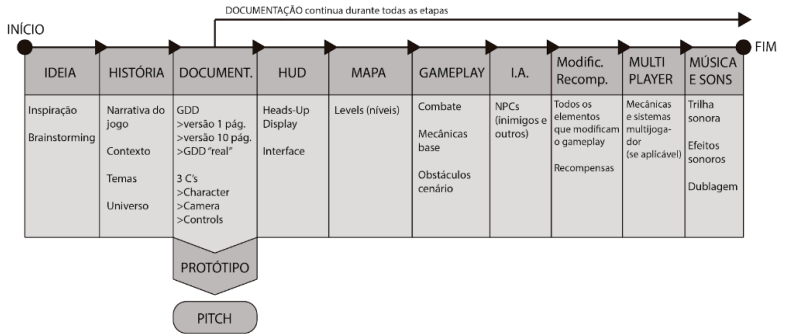
Outro fator importante na pesquisa bibliográfica é o acompanhamento das novidades, pois os jogos, sendo uma mídia tecnológica e digital possui um avanço muito rápido, onde por exemplo a capacidade gráfica dos jogos aumenta exponencialmente, tornando técnicas novas de modelagem 3D obsoletas em pouco tempo. E também, por mais que um livro de *game design* seja de qualidade, os jogos que nele forem citados revelarão a sua idade, e por consequência a sua relevância atual.

Mas há autores que abordam mais a prática geral envolvida, trazendo sugestões de metodologia, como Rogers S.(2010), Fullerton T.(2008) e Novak J.(2012), que apresentam séries de tópicos, esquemas e explicações para cada parte fundamental do desenvolvimento do projeto em seus respectivos livros.

Rogers S.(2010), em seu livro *Level Up! The Guide To Great Video Game Design*, apresenta uma metodologia tradicional pelo livro todo, dando ênfase ao GDD (sigla para *Game Design Document*, que significa Documento do Design do Jogo, documento utilizado durante o desenvolvimento de um projeto para detalhar e registrar tudo sobre os aspectos do jogo) ao mesmo tempo que apresenta cada faceta principal

de um jogo. Na imagem a seguir (Figura 6), é visto que após determinado o conceito principal geral do jogo, é construído um GDD básico de cerca de dez páginas, que é suficiente para que um *pitch* do jogo seja criado para apresentar ao “publicador” (produtora de jogos).

Figura 6 – Metodologia de desenvolvimento de jogos de Rogers.



Fonte: Desenvolvido pelo autor.

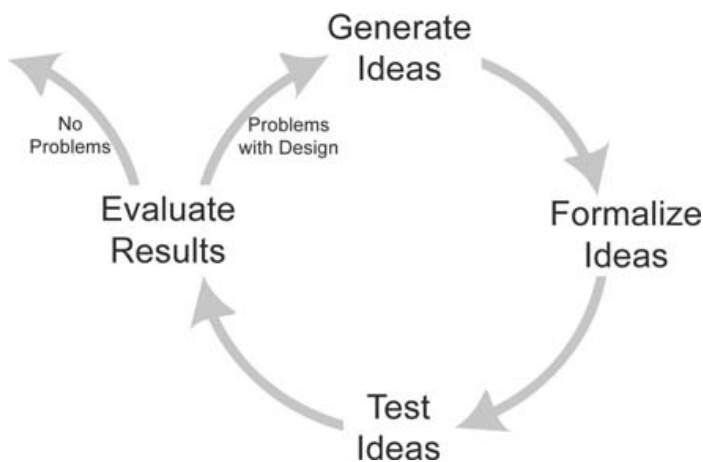
O autor do livro recomenda também que junto com o *pitch* seja apresentado um protótipo que contenha algo que mostre a essência do projeto, o diferencial ou ainda porque é um projeto que “vale a pena”. É importante que seja realmente *gameplay* (jogabilidade) de alguma forma o protótipo, entretanto o autor não entra em detalhes sobre a criação do protótipo em si. O livro então continua sobre as etapas de criação de um jogo, ordenando-as na ordem em que seriam desenvolvidas, juntamente com a atualização do GDD sobre cada uma delas. Porém isto não é demonstrado de uma maneira muito orgânica, porque a prática de desenvolvimento de um jogo demanda que certas partes sejam realizadas ao mesmo tempo, ou que pela natureza do projeto certa parte do jogo será adicionada mais tarde, logo indo contra a ordem da metodologia de Rogers.

No livro *Game Design Workshop - A Playcentric Approach To Creating Innovative Games*, a autora Fullerton T.(2008) apresenta a metodologia baseada na iteração (Figura 7) e foco na experiência do jogador, que consiste em:

- Determinar metas da experiência do jogador (quais emoções ele vai sentir, quais serão as impressões sobre algo etc).

- Ideia ou sistema é concebido.
- Ideia ou sistema é formalizado (é escrito ou é feito protótipo).
- Ideia ou sistema é testado contra as metas.
- Resultados são avaliados e priorizados.
- Se os resultados são negativos e a ideia ou sistema parecem fundamentalmente falhos, se volta ao primeiro passo.
- Se há progresso nos resultados, são modificados e testados novamente.
- Se os resultados são positivos e a ideia ou sistema parecem bem-sucedidos, o processo iterativo foi completado.

Figura 7 – Diagrama do processo iterativo.



Fonte: Livro *Game Design Workshop - A Playcentric Approach To Creating Innovative Games* (2008).

Tendo o processo iterativo em mente, a metodologia segue as seguintes etapas:

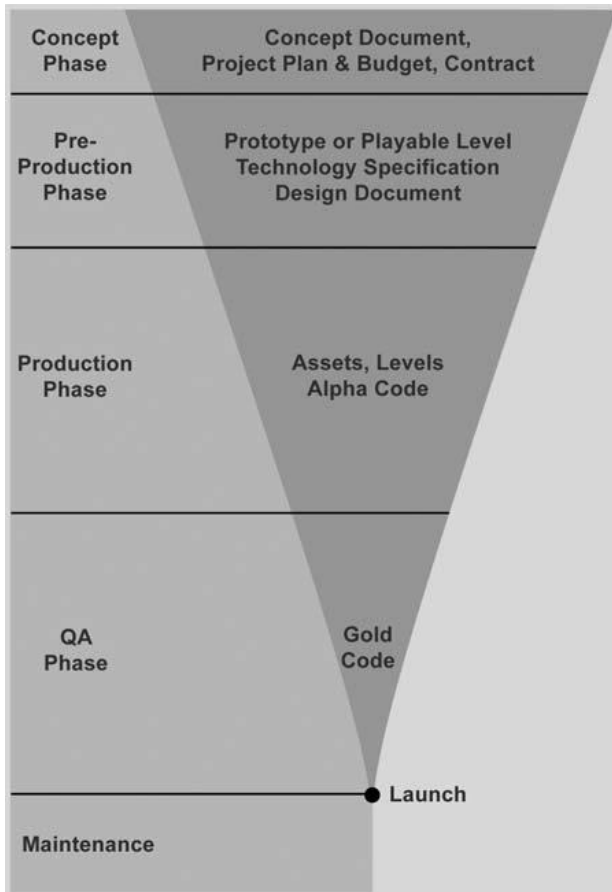
1) *Brainstorming*

- Metas da experiência do jogador são criadas
- Se imagina conceitos ou mecânicas que poderiam atingir as metas

- São separadas as três melhores ideias
 - Descrição curta de cada ideia
 - Testar com potenciais jogadores (pode-se usar recursos visuais para ajudar nesta parte, onde se determina se a ideia é bem aceita entre os jogadores)
- 2) Protótipo físico
- *Playtest* de um protótipo físico simples do jogo
 - Se funcionar e atingir as metas, descrever como o jogo funcionaria
- 3) *Pitch* (opcional)
- Artes iniciais (para passar impressões do visual e estilo do jogo)
 - *Gameplay* básico (se já houver um protótipo digital)
- 4) Protótipo *software*
- Equipe de protótipo pronta, protótipos iniciais das mecânicas
 - *Playtest*
 - Se o protótipo *software* demonstrar que está perto das metas almejadas, avançar para a etapa de documentação
- 5) Documentação
- Documentação de todo aspecto do jogo (GDD)
 - *Wiki* como GDD (opcional, é mais flexível e orgânico de ficar atualizando durante a produção)
 - Quando terminar a 1ª versão do GDD, mover para produção
- 6) Produção
- Trabalhar com todos os membros da equipe para saber se tudo é alcançável e corretamente descrito na documentação
 - Criação da arte e programação “para valer”
 - Manter o processo iterativo
- 7) Controle de qualidade
- Testes do jogo já quase como produto
 - Ter certeza que está perto das metas

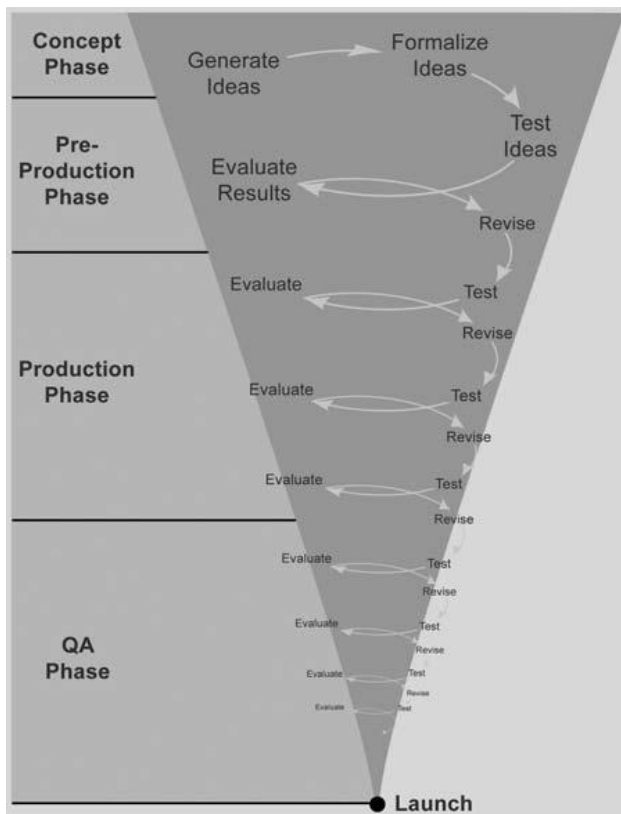
Visto a metodologia em tópicos, é possível então simplificá-la nas etapas do desenvolvimento (Figura 8) e como modelo de *game design* iterativo (Figura 9):

Figura 8 – Etapas do desenvolvimento.



Fonte: Livro *Game Design Workshop - A Playcentric Approach To Creating Innovative Games* (2008).

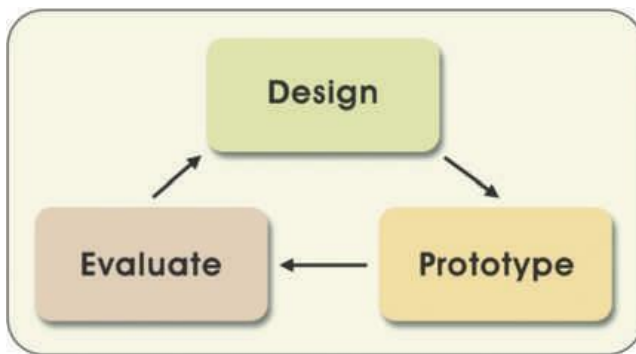
Figura 9 – Modelo para *game design* iterativo: *playtest*, avaliação, revisão.



Fonte: Livro *Game Design Workshop - A Playcentric Approach To Creating Innovative Games* (2008).

Para Novak J.(2012) em seu livro *Game Development Essentials*, a autora apresenta também o desenvolvimento iterativo (Figura 10), afirmando que é o processo que mais se adequa à produção de jogos, visto que são *softwares*, logo tendo um processo de criação muito diferente de que um filme, por exemplo.

Figura 10 – Processo de desenvolvimento iterativo de três estágios.



Fonte: Livro *Game Development Essentials* (2012).

O diagrama e teoria da autora são basicamente os mesmos que os de Fullerton T.(2008), no que se diz ao processo iterativo (inclusive sobre o processo estar aplicado em toda a metodologia, repetido sempre até o jogo virar um produto), porém na metodologia em si vemos diferenças:

1) Conceito

- Ideia para o jogo é concebida
- Identificar o público-alvo
- Avaliar recursos da empresa/estúdio
- Escrever o documento de conceito (documento curto que descreve a ideia original do jogo)

2) Pré-produção

- Parecendo promissor o projeto, documentação adicional é criada
- Guia do estilo da arte do jogo é criado
- Plano de produção é criado
- Termina com criação do GDD

3) Protótipo

- Protótipo físico é criado e testado pela equipe antes do digital (opcional)
- Criação de um protótipo digital tangível, que mostre a essência e diferencial do jogo (foco no *gameplay* com as mecânicas principais)

- Protótipo digital é apresentado no *pitch*, e o resultado determina se o projeto continua

4) Produção

- Fase mais longa, atual produção do jogo
- Resultado é o jogo “básico”

5) Alfa

- Jogo "básico", jogável de início ao fim
- Texto do jogo na língua principal completo
- Interface básica com documentação preliminar
- Compatibilidade com a maioria das configurações de *hardware* e *software*
- Requisitos mínimos de sistema testados
- Interface manual testada para compatibilidade
- Arte e áudio *placeholder* inseridos
- Funcionalidade do *multiplayer* testada (se aplicável)

6) Beta

- *Assets* (recursos visuais e outros) finais são integrados
- Produção encerra
- Eliminação de *bugs*
- Solução de problemas de performance
- Testes em todas as plataformas suportadas são realizados
- Todos os seguintes elementos devem estar completados:
 - Código do jogo
 - Conteúdo do jogo
 - Interface do usuário
 - Compatibilidade de *hardware/software*
 - Compatibilidade da interface manual
 - Arte e áudio do jogo

7) Gold

- Jogo está pronto
- Mandado para produção de discos (e/ou para venda online por *download*)
- Jogo é lançado no mercado

8) Pós-produção

- Aumento da longevidade e manutenção do jogo

- Criação de *patches* (atualização do jogo para consertar *bugs* pós-lançamento)
- Criação de *updates* (conteúdo adicional criado para realçar/melhorar o jogo original, normalmente gratuito)
- Criação de DLC (sigla para *Downloadable Content*, conteúdo extra para o jogo, normalmente pago)

Explicadas as metodologias dos autores, podemos perceber que as de Fullerton T.(2008) e Novak J.(2012) são muito semelhantes (como afirmado anteriormente), ambas ressaltam o uso do desenvolvimento iterativo durante todo o projeto e possuem estágios parecidos. Rogers S.(2010) é o que mais difere, sugerindo guiar mais a produção do jogo por etapas conceituais seguindo o modelo de GDD, entretanto os três autores dão atenção especial ao *pitch*, o que mostra a importância dessa parte para que o projeto seja aceito e continue a produção de modo saudável.

3 PESQUISA

O gênero de jogo escolhido para *Breakfast Detectives* é o *point-and-click*, um subgênero do jogo de aventura, classicamente caracterizado como um jogo onde o jogador assume o papel do protagonista em uma história interativa conduzida pela exploração de cenários e solução de quebra-cabeças.

O jogo de aventura original (surgido nos anos 70) era baseado em texto, onde a situação era apresentada e o jogador digitava a sua ação, tudo sem imagens. O jogo *Colossal Cave Adventure* (1976) é identificado como o primeiro jogo de aventura desse tipo, seguido de exemplos mais famosos como *Zork* (1980) e *The Hitchhiker's Guide to the Galaxy* (1984).

Com o advento de computadores pessoais mais poderosos com a capacidade de exibir gráficos (imagens, animações), os jogos de texto começaram a ter imagens ilustrativas acompanhando o texto, popularizando o gênero e evoluindo para interfaces *point-and-click*, dando origem ao formato discorrido aqui e muitos outros.

É difícil definir qual foi o primeiro jogo *point-and-click*, onde o jogador realmente utiliza a interface com um mouse, ao invés de ainda usar comandos escritos, como em *King's Quest* (1984). Entretanto pode-se apontar *Maniac Mansion* (1987) da LucasArts como o jogo que estabeleceu a mecânica devido à uma nova *engine* (chamada SCUMM) criada especificamente para ele, que depois serviu para vários outros jogos *point-and-click* da desenvolvedora, como *The Secret of Monkey Island* (1990); *Sam & Max Hit the Road* (1993), este sendo uma influência direta para *Breakfast Detectives*; e *Full Throttle* (1995).

Os anos 90 foram a “Era de Ouro” dos jogos *point-and-click*, e não somente para a LucasArts, a Sierra e outras desenvolvedoras produziram títulos como a série *Quest for Glory* (1989~1998); *Simon the Sorcerer* (1993); *I Have No Mouth, and I Must Scream* (1995); *Flight of the Amazon Queen* (1995); e *Blade Runner* (1997), este último já apresentando gráficos 3D. O *point-and-click* nessa época era considerado um dos gêneros mais tecnicamente desenvolvidos, apresentando mecânicas como “árvores de diálogo” (conversas com NPCs que se dividem em “ramos” dependendo das escolhas de falas do jogador), inventários complexos, possibilidade de finais diferentes, e com o avanço da tecnologia, gráficos cada vez melhores (chegando ao 3D, como mencionado anteriormente), além de apresentarem contextos imaginativos e ecléticos (abrangendo de fantasia, ficção científica,

comédia, horror e outros) possuindo histórias cativantes lembradas até hoje.

Porém o reinado do gênero foi diminuindo, devido ao aumento de popularidade dos jogos de tiro em primeira pessoa no início dos anos 2000 (SALTER, p.20). Considerado um “gênero de nicho”, ficou difícil projetos de jogos assim encontrarem investimento de publicadoras. Contudo, o interesse em jogos *point-and-click* tem contemplado um ressurgimento pelo sucesso de novos títulos de desenvolvedores independentes, muitos deles possibilitados por financiamento coletivo (*crowdfunding*), e a distribuição digital, facilitando o acesso aos novos jogos tanto quanto aos antigos.

Novas filosofias de *game design*, junto com maior liberdade e capacidade gráfica tornaram os novos jogos do gênero muito mais acessíveis e atraentes ao consumidor moderno, considerando que os jogos antigos muitas vezes possuíam mecânicas desnecessariamente complexas, fazendo com que alguns títulos (da “Era de Ouro”) necessitem mais paciência por parte do jogador acostumado ao estilo dinâmico do *game design* vigente.

Novos títulos que fizeram sucesso foram *Gemini Rue* (2011); *Deponia* (2012); *Broken Age* (2014), este desenvolvido por veteranos da indústria e do gênero, foi totalmente financiado por *crowdfunding*, viabilizando este formato de produção de jogos “de nicho”; e *Paradigm* (2017), notável por ter sido principalmente desenvolvido por uma só pessoa. Esse ressurgimento do gênero também trouxe nomes do passado, seja em novos jogos ou “remasterizações” (processo de formatação de jogos antigos para resoluções maiores de tela, envolvendo a transformação dos gráficos para alta definição), como *The Secret of Monkey Island: Special Edition* (2009), mais um *remake* do que uma remasterização do jogo de 1990, pois além de gráficos melhores, foram refeitas a partir do original a trilha sonora, dublagem e pequenas melhorias no *game design*; *Sam & Max The Devil's Playhouse* (2010); *Grim Fandango Remastered* (2015), remasterização do clássico *cult* de 1998; e *Thimbleweed Park* (2017), desenvolvido pelos mesmos criadores de *Maniac Mansion* e *The Secret of Monkey Island*, concebido como um “sucessor espiritual”, sendo semelhante nos gráficos e na jogabilidade dos jogos da LucasArts da época.

Concluída a pesquisa, se constata que o atual mercado para jogos *point-and-click* se encontra favorável, e tomando como exemplo os já mencionados *Paradigm* e *Thimbleweed Park* do ano passado, ainda há público para jogos do gênero (SALTER, p.136-137), como é a proposta de *Breakfast Detectives*.

4 DESENVOLVIMENTO

4.1 MÉTODO

Tendo em vista as metodologias dos autores mencionados anteriormente, é possível propor um método de desenvolvimento de jogos mais prático e objetivo direcionado ao atual cenário do desenvolvedor brasileiro (ao contrário da indústria de alto orçamento, chamada “AAA”, ou “triple A”, a qual os métodos detalhados dos autores são principalmente destinados). Logo, este método é um apanhado geral e adaptação das metodologias apresentadas, sobretudo o desenvolvimento iterativo descrito pelas autoras Fullerton T.(2008) e Novak J.(2012), e, apesar de ser utilizado como guia no protótipo de *Breakfast Detectives*, ele também se aplica a um projeto inteiro, ou seja, um jogo comercializável.

As adaptações necessárias para o método (e a sua aplicação em *Breakfast Detectives*) consistem em eliminação da etapa de protótipo físico, sendo que é muito difícil fazer um que passe a mesma experiência que o protótipo digital; resumo de itens demasiadamente específicos em itens maiores, como nas etapas Alfa e Beta; agrupamento de itens semelhantes em uma só etapa; retirada de menções à modos *multiplayer* de jogo, já que não é do interesse deste projeto; e itens como “realizar testes em todas as plataformas” foram excluídos ou mantidos como facultativos, já que este jogo (a princípio) seria apenas para computador. Houve também a adição de itens pertinentes à natureza do projeto e do tipo do jogo, como “análise do mercado atual” na primeira etapa deste método, visto que o *point-and-click* é um gênero de nicho atualmente.

Na parte que se refere quanto a prototipação, a máxima é a iteração e ação rápida (com mínima documentação) para obter resultados imediatos e logo ter uma noção concreta das mecânicas do jogo, assim como criar um bom material para o *pitch*, que sendo bem-sucedido permitirá a continuação/início do projeto de maneira efetiva.

A seguir estão descritas as etapas do método, tendo em mente que o desenvolvimento iterativo (como o da Figura 10) e o GDD (documentação) deve estar presente em toda etapa a partir da Pré-produção, e itens marcados com “*” são facultativos:

1) Conceito

- Análise do mercado atual
- Avaliar recursos disponíveis
- *Brainstorming*
- Identificação do público-alvo (ideias geradas pelo *brainstorming* podem ser modificadas nesta parte)
- Visão geral do jogo é concebida
- Documentação básica (conceitos principais, essência do jogo)

2) Pré-produção

- Traçar o plano de produção
- Iniciar GDD a partir da documentação básica (e mantê-lo nesta etapa, retomar na etapa 4)

3) Protótipo *Software*

- Desenvolvimento do protótipo digital com as mecânicas principais do *gameplay*, que demonstrem a essência e o diferencial do jogo
- Criação das artes conceituais iniciais (estabelecimento do estilo gráfico do jogo)
- *Playtesting* até se chegar pelo menos ao mínimo almejado (processo iterativo)
- Documentar separadamente (do GDD) as mudanças realizadas durante o processo iterativo
- *Pitch* com o protótipo contendo as artes iniciais e o *gameplay* básico*
- Avançar para a próxima etapa caso o *pitch* seja aprovado ou o projeto estiver se mostrando promissor

4) Produção

- Retomar GDD
- Intensa aplicação do processo iterativo, desenvolvimento do jogo inteiro (todas as áreas)
- Estágio Alfa:
 - Jogo básico (jogável do início ao fim)
 - Texto na língua principal completo
 - Testada a compatibilidade com diversos *hardware* e *software*
 - Determinados os requisitos mínimos de sistema para suportar o jogo
 - Arte e áudio do jogo *placeholder*

- Estágio Beta:
 - *Assets* finais são integrados
 - Eliminação de *bugs*
 - Produção do jogo encerra
- 5) Controle de Qualidade (*QA*)
 - Testes em todas as plataformas (testes para ver se o jogo funciona sem problemas) *
 - Correção de problemas de performance
- 6) Gold
 - Desenvolvimento concluído
 - Jogo enviado para lançamento (plataforma digital)
- 7) Pós-produção
 - Manutenção (*patches*, suporte ao usuário)
 - Criação de *updates* (volta à etapa 4) *
 - Criação de DLCs (volta à etapa 4) *

Observando as etapas e os itens do método nota-se que não há particularidades referentes à *engine* utilizada no projeto. Apesar da Unreal Engine 4 ser a escolhida para este projeto pela sua gratuidade, *blueprints* e outros, a escolha de manter o método "universal" se deve à intenção de facilitar o trabalho do desenvolvedor independente iniciante, deixando-o livre para escolher uma *engine* adequada ao seu estilo de desenvolvimento e jogo, ao mesmo tempo que utiliza deste método.

Entretanto, considerando que este método foi criado junto com um jogo 2D *point-and-click* na Unreal, a criação de projetos semelhantes nesta plataforma é comprovadamente viável.

4.1.1 Estado Inicial e Desenvolvimento do Protótipo

Visto as etapas do método, conclui-se que se o projeto de *Breakfast Detectives* estivesse sendo produzido como um “jogo inteiro”, a sua produção estaria na etapa 3. Durante o Trabalho de Iniciação Científica do autor no Curso de Graduação de Design de Jogos e Entretenimento Digital da UNIVALI, onde *Breakfast Detectives* começou, foram realizadas as etapas 1 e 2 completamente e parou na etapa 3, mais precisamente no terceiro item da etapa (o *playtesting*), onde o protótipo chegou no seu mínimo almejado.

Logo, é pertinente descrever exatamente qual era o seu estado no fim do TIC (o ponto de partida do PCC) em contraste com seu estado no fim deste PCC, incluindo quais as funcionalidades, aspectos gráficos e

elementos que mudaram ou continuaram os mesmos. Na tabela a seguir (Tabela 1) são comparadas as características do protótipo nesses dois estados, com “X” marcando a presença do elemento e “-” a ausência ou elemento incompleto:

Tabela 1 – Comparação das versões do TIC e do PCC de *Breakfast Detectives*

Elementos	TIC	PCC
Interface inicial com instruções de controle	X	X
Inventário interativo	-	-
Animações de personagens	-	X
Animações de cenário	-	-
Sistema de diálogo (com escolha de resposta)	X	X
Sistema de <i>quest</i>	-	X
Interface <i>in-game</i> para ver quests	-	X
Sistema de NPC	-	-
Sistema de objetos (pistas)	-	-
Sistema para salvar o jogo (estado e progresso)	-	X
<i>Cutscene</i> antes do jogo começar	-	-
Efeitos sonoros	-	-
Música de fundo	-	X
<i>Assets</i> finais (personagens, cenário, objetos)	-	X
Opção na interface <i>in-game</i> de sair do jogo	-	-

Fonte: Desenvolvido pelo autor.

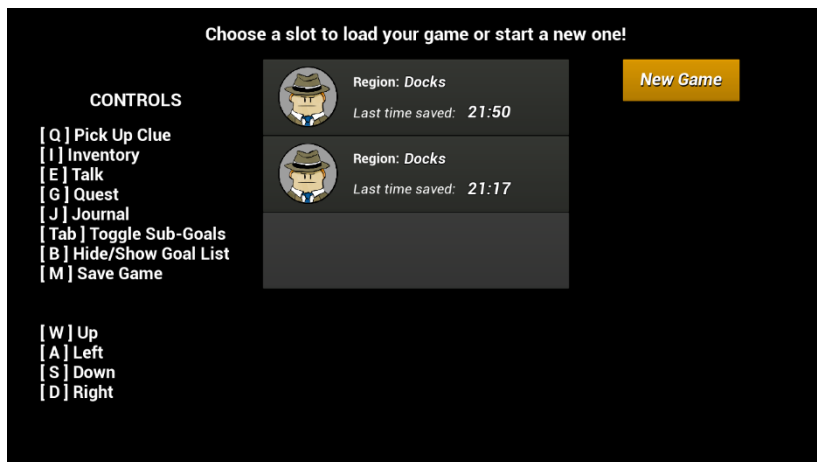
Pode-se ver que a maioria de elementos marcados com “X” na Tabela 1 são da versão PCC de *Breakfast Detectives*, e que a versão TIC só possui dois marcados. Nesses, a interface inicial trocou da com “*Start/Quit*” da versão TIC (Figura 11) para a tela com escolha de *slots* para salvar o jogo da versão PCC (Figura 12).

Figura 11 – Interface inicial do protótipo versão TIC (o “beta” foi colocado para denotar que era a última versão no momento)



Fonte: Desenvolvido pelo autor.

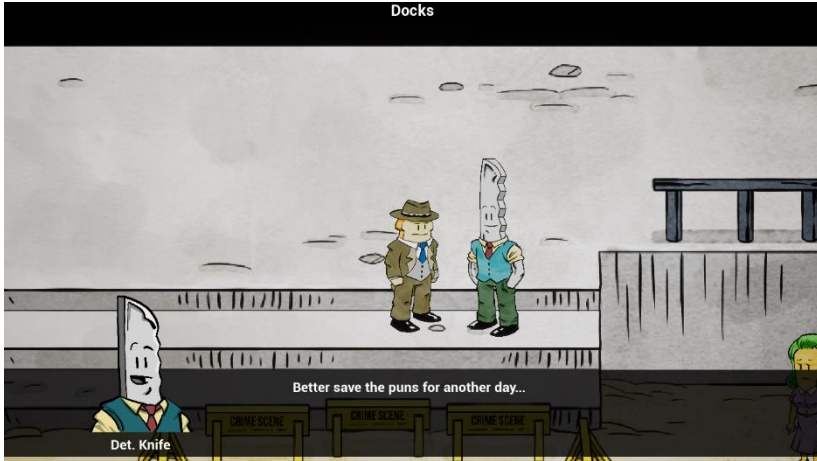
Figura 12 – Interface inicial do protótipo versão PCC



Fonte: Desenvolvido pelo autor.

Já o sistema de diálogo continuou o mesmo, trocando apenas o texto de algumas falas, e é possível conversar somente com o NPC Det. Knife (Figura 13).

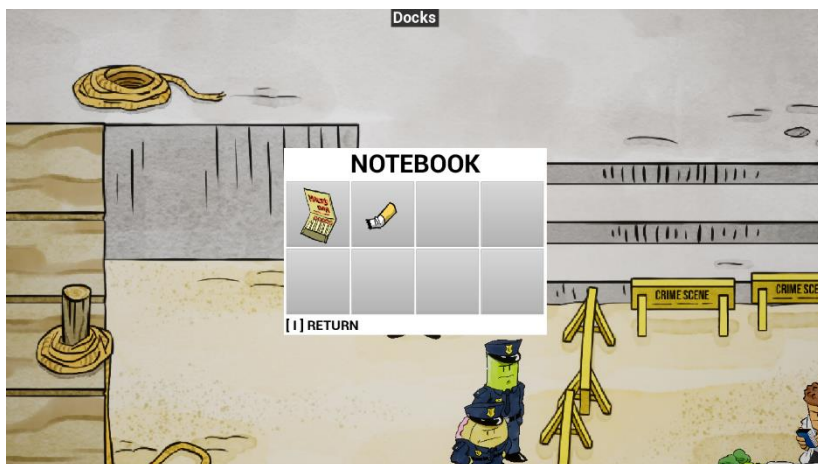
Figura 13 – Jogador em diálogo com o NPC Det. Knife



Fonte: Desenvolvido pelo autor.

O elemento do inventário interativo, onde o jogador poderia ver os objetos (pistas) coletados e detalhes sobre eles continuou sem a parte da interação, logo só se observa os ícones dos objetos coletados no inventário (Figura 14).

Figura 14 – Jogador observando o inventário de objetos coletados



Fonte: Desenvolvido pelo autor.

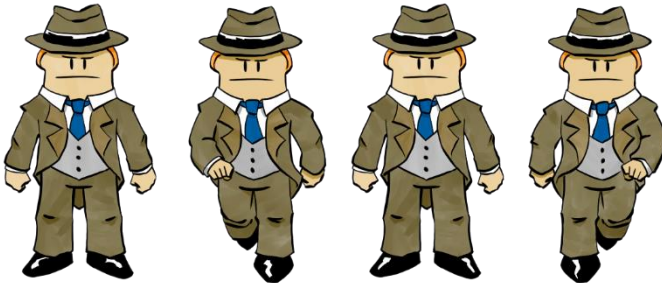
Ausentes na versão TIC, as animações dos personagens foram criadas e implementadas no protótipo atual. Na versão TIC o personagem do jogador podia se deslocar pelo cenário, mas não havia nenhuma animação de caminhada (Figura 15). Agora há as animações de andar para cima, baixo (Figura 16), direita e esquerda, além da *idle* referente a cada posição. As animações *idle* (tanto as do personagem do jogador como as dos NPCs) consistem em o personagem piscar os olhos, com pequenas variações, o NPC Dr. Hazel anotando em seu caderno (Figura 17) e a fumaça do cigarro da NPC Ms. Carrot (Figura 18).

Figura 15 – Captura de tela do *gameplay* de *Breakfast Detectives* versão TIC



Fonte: Desenvolvido pelo autor.

Figura 16 – *Sprites* do *walk cycle* baixo do Det. B. Read, o personagem do jogador



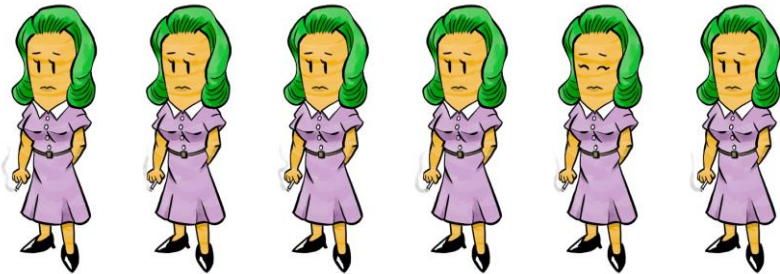
Fonte: Desenvolvido pelo autor.

Figura 17 – *Sprites idle* do NPC Dr. Hazel



Fonte: Desenvolvido pelo autor.

Figura 18 – *Sprites idle* da NPC Ms. Carrot



Fonte: Desenvolvido pelo autor.

Foram planejadas animações mais complexas para os NPCs e para o personagem do jogador, mas devido à divisão de tempo para a realização dos outros elementos, elas foram simplificadas. A animação de cenário para a cena das docas (Figura 19) seria as ondas do mar aparecendo na parte inferior da praia, mas não foi realizada.

Figura 19 – Jogador na parte inferior do cenário, onde apareceriam as ondas do mar

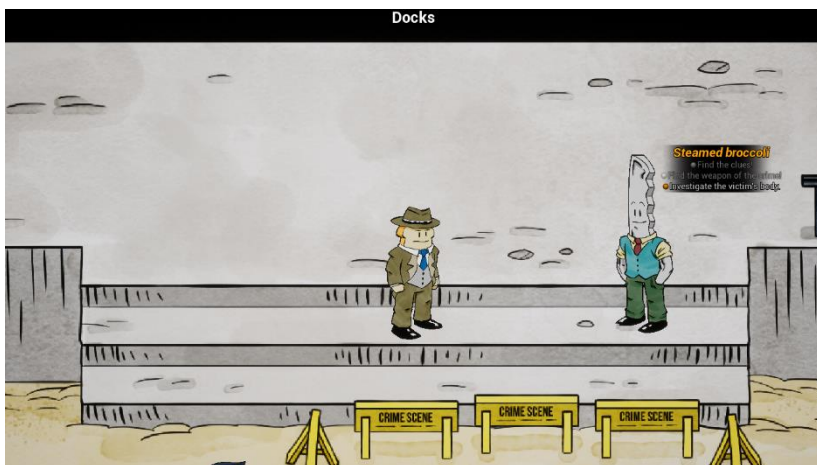


Fonte: Desenvolvido pelo autor.

O sistema de *quest* foi uma das maiores mudanças entre as versões, porque além do próprio sistema, ele envolve a criação da interface *in-game* para ver as *quests*, do sistema de NPC e do sistema de objetos. O resultado foi um tanto rudimentar, pois os sistemas de NPC e de objetos ficaram incompletos, porém já é possível ter uma ideia do que seria em uma versão final.

Na versão do TIC não havia *quest*, o jogador apenas poderia conversar com o NPC Det. Knife (que dava algumas instruções do que fazer), coletar 3 pistas pelo cenário e então aparecia uma mensagem de *feedback* onde logo após encerrava o jogo. A estrutura do *gameplay* continua essencialmente a mesma na versão do PCC, porém quando o jogador interage com o Det. Knife, ele ganha a *quest* referente ao cenário atual, aparecendo os objetivos na tela (Figura 20).

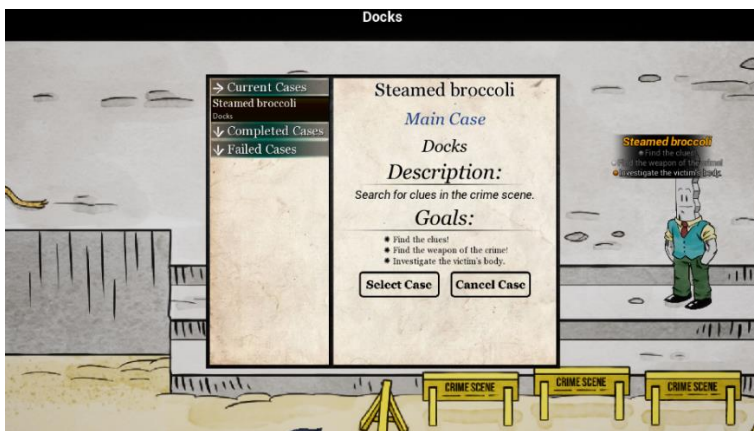
Figura 20 – Jogador acabando de ganhar a *quest* e aparecendo os objetivos no lado direito da tela



Fonte: Desenvolvido pelo autor.

Acessando a interface *in-game*, o jogador pode ver informações mais detalhadas sobre a *quest*, além da opção de selecioná-la ou cancelá-la (Figura 21).

Figura 21 – Jogador acessando a interface *in-game*



Fonte: Desenvolvido pelo autor.

Como foi mencionado anteriormente, só é possível conversar com o Det. Knife, apesar da intenção do sistema de NPC de possibilitar a interação com todos os demais NPCs (Dr. Hazel, Ms. Carrot e os policiais Doughnut e Celery). Conversar com a Ms. Carrot cumpriria um dos objetivos da *quest*, mas devido à incompletude do sistema de NPC, não há progresso nessa parte (Figura 22).

Figura 22 – Jogador só consegue iniciar diálogo com o NPC Det. Knife (o “[E] TALK” não aparece quando o jogador se aproxima de outros NPCs)



Fonte: Desenvolvido pelo autor.

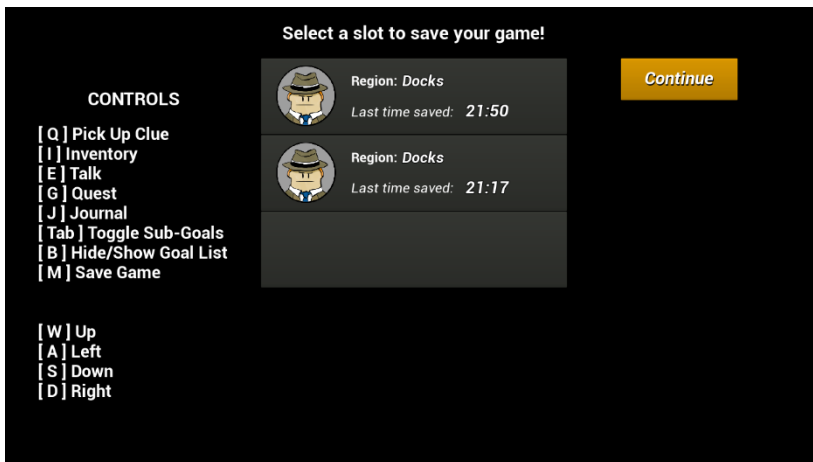
Quanto ao sistema de objetos, a sua particularidade é que ele não possui uma conexão direta com o sistema de *quest*, pois mesmo coletando todas as pistas do cenário, aparece uma mensagem de *feedback* (Figura 23) e se finaliza a sessão de jogo do protótipo, sem completar a *quest*, quase como na versão TIC. Assim foi mantido pela questão de fazer outros elementos do protótipo.

Figura 23 – Mensagem de *feedback* após coletar todas as pistas



Fonte: Desenvolvido pelo autor.

O sistema para salvar o jogo foi um elemento novo no protótipo, que permite ao jogador no meio do *gameplay* abrir uma interface onde ele pode salvar o seu estado atual e progresso no jogo, escolhendo um dos três *slots* para armazenar essas informações. E então a próxima vez em que o jogo é aberto, é possível escolher o *slot* previamente salvo e continuar a sessão de onde havia parado. A interface conta também com instruções básicas de controle e o botão para começar ou continuar a sessão (Figura 24).

Figura 24 – Interface para salvar o jogo aberta durante o *gameplay*

Fonte: Desenvolvido pelo autor.

O caso (criminal) retratado na versão PCC de *Breakfast Detectives* é baseado em um curta-metragem de animação criada pelo autor em 2015, também chamado *Breakfast Detectives* (Figura 25). Um dos elementos opcionais planejados para implementar seria um pequeno trecho da animação que rodaria antes do jogo começar, logo depois do jogador apertar o botão para começar a sessão. Porém, novamente foi dada atenção a outros aspectos do protótipo, e este elemento não foi desenvolvido.

Figura 25 – Captura de tela da cena inicial do curta-metragem de animação *Breakfast Detectives*



Fonte: Desenvolvido pelo autor.

A implementação de elementos como efeitos sonoros (som de caminhada no concreto/areia, som do mar e outros) e a opção de sair do jogo por meio de uma interface *in-game* foram deixados de lado por prioridade a outros elementos.

Apesar da ausência de efeitos sonoros, o protótipo possui uma música de fundo cíclica que começa a tocar no início da sessão, só interrompendo caso o jogador abra a interface de salvar o jogo ou termine a fase. A música é *royalty-free*, chamada “Cool Vibes” e composta por Kevin McLeod. Essa música foi escolhida para realçar a atmosfera de “detetive” do jogo.

Os *assets* implementados na versão PCC são considerados finalizados (coloridos e com sombras), também porque se assemelham mais como o jogo seria em sua versão final (isto é, como um jogo completo). As resoluções das imagens são padronizadas entre todos eles, algo que faltou na versão TIC.

Enfim, nota-se que nos elementos que não foram implementados ou incompletos, a razão dada para isso foi a falta de tempo e/ou prioridade de outros elementos. De fato, na parte de desenvolver o protótipo neste PCC, a prioridade foi focar na parte visual (do cenário e *sprites* principalmente) e complementar as mecânicas do jogo com a programação. Isto se deve à intenção de corrigir certos aspectos

essenciais como a animação que faltaram na versão TIC. No anexo do final do relatório (ANEXO A), há o registro de desenvolvimento do protótipo da versão PCC, onde se vê cada ação e mudança realizada, mostrando o uso do desenvolvimento iterativo para um mesmo elemento.

5 CONCLUSÃO

Depois de criar o método e aplicá-lo no desenvolvimento do protótipo de *Breakfast Detectives* e, tendo em vista o resultado, pode-se afirmar que ela foi validada na questão de ser um método viável para a produção de jogos no contexto nacional. Os recursos das ferramentas escolhidas foram também um fator decisivo, permitindo que o protótipo evoluísse e atingisse a visão do jogo, com as animações, mecânicas de *gameplay* e o estilo visual criando uma unidade que não havia antes no protótipo.

Contudo, durante o desenvolvimento do protótipo de *Breakfast Detectives*, a ausência de conhecimento de programação avançado fez falta, apesar da utilização de meios alternativos como os *blueprints*. O sistema de *quest* ficou incompleto, assim como os sistemas de NPC e de objetos, fazendo que essas mecânicas não interagissem uma com a outra. O método se provou efetivo, mas nem todas as especificidades do projeto foram atingidas, como visto na parte de desenvolvimento do protótipo. Logo se vê que é limitado o que se pode fazer por meio de atalhos em programação, portanto a criação de mecânicas inovadoras ou totalmente adaptadas ao projeto só são possíveis com maior conhecimento na área, que é capaz de criar soluções objetivas aos problemas que surgem durante o processo iterativo, por exemplo.

Apesar disso, o desenvolvedor que utilizar deste método poderá começar a construir um projeto onde por meio da dinâmica da iteração e ferramentas acessíveis conseguirá, ao menos, passar a impressão de qual é a intenção almejada para o *gameplay*. Assim, com um protótipo estável e em andamento, conseguir trazer pessoas ao projeto que supram as necessidades do jogo se torna uma situação simples e que não desvirtuará a ideia original do *game design* do projeto.

REFERÊNCIAS

ARNHEIM, Rudolf. **Arte e Percepção Visual**: Uma Psicologia da Visão Criadora. São Paulo: Cengage Learning, 2014.

AUMONT, Jacques. **A Imagem**. São Paulo: Papirus Editora, 2011.

FULLERTON, Tracy. **Game Design Workshop - A Playcentric Approach To Creating Innovative Games**. 2ª ed. San Francisco: Elsevier, 2008.

ROGERS, Scott. **Level Up! The Guide To Great Video Game Design**. Chichester: Wiley, 2010.

ROUSE III, Richard. **Game Design - Theory & Practice**. 2ª ed. Sudbury: Wordware Publishing, 2005.

NOVAK, Jeannie. **Game Development Essentials**. 3ª ed. Clifton Park: Delmar, Cengage Learning, 2012.

SALTER, Anastasia. **What Is Your Quest?: From Adventure Games to Interactive Books**. Iowa City: University of Iowa Press, 2014.

SCHELL, Jesse. **The Art Of Game Design - A Book Of Lenses**. Amsterdam: Elsevier, 2008.

SCHUYTEMA, Paul. **Design de Games - Uma Abordagem Prática**. São Paulo: Cengage Learning, 2008.

ANEXO A – Registro de Desenvolvimento do protótipo versão PCC

- Programação implementada para rodar animações diferentes de acordo com a direção do personagem (cima, baixo, direita, esquerda)
- Animações *placeholder* implementadas (cima, baixo), versões *walk* e *idle*
- Correção no *blueprint* de *walk up* (era para o *walk down*) e *walk down* (era para o *walk up*)
- Falta implementar animações de *walk right* e *idle right*
- Falta programação para *idle right*
- O jogo começa com o jogador em *idle up* (deveria ser *idle down*)
- Diminuir *hitbox* do personagem do jogador para a área dos pés
- Realizada programação para *idle right*
- Feita e implementada animações *walk right* e *idle right*
- Diminuir velocidade do personagem do jogador
- Ajustada a velocidade do personagem do jogador após vários testes
- Continuação dos ajustes na *hitbox* do personagem do jogador
- *Hitbox* do personagem do jogador ajustada (*hitbox* de formato cápsula reduzida a uma esfera, cobrindo embaixo da virilha até os pés do personagem)
- Posição do personagem do jogador e câmera ajustados (centralizados)
- Jogo começa agora com o personagem do jogador em *idle down* (olhando para a câmera)
- Início de criação de sistema de *quest*
- Interfaces do sistema de *quest* implementadas
- Ajustes ao sistema de *quest*
- Criado sistema de NPC
- Criado sistema de objetos (pistas)
- *Assets* finais de cenário e objetos concluídos
- *Assets* finais implementados no jogo (na *engine*)
- Versões finais dos *sprites* de personagens realizados
- *Sprites* (versões finais) de personagens implementados no jogo (na *engine*)
- Ajustes finais realizados

GLOSSÁRIO DE TERMOS EM INGLÊS

Assets – Conteúdo do jogo como modelos 3D, texturas, arquivos de áudio, tudo do jogo que é feito por artistas e designers.

Blueprint – “Diagrama” ou “desenho técnico”.

Brainstorming – Uma atividade criada para explorar a potencialidade criativa de um indivíduo ou um grupo. Baseia-se na geração espontânea de ideias.

Bugs – Erros de *software* que podem causar problemas drásticos dentro do código do jogo. Um dos principais problemas é que certos tipos de *bugs* podem ser explorados pelo jogador e desvirtuar a experiência idealizada pelo desenvolvedor.

Cutscene – Sequência não-interativa de um jogo, segmentando o *gameplay*. Pode ser usada para mostrar conversas entre personagens, ambientação, introduzir novos elementos de *gameplay*, entre outros. É comum o jogo utilizar animações (vídeos) para tal.

Cult – “Culto”. No contexto de obras/mídia, é quando uma obra possui um “seguimento” *cult*, isto é, fãs devotos e apaixonados pela obra. Muito usado no contexto de filmes com o termo “clássico *cult*”.

Down – “Baixo”. No contexto do *sprite*, é quando o personagem se move para a parte inferior do cenário, com a frente visível ao jogador.

Download – Receber informações de um sistema remoto. Comumente chamado de “baixar”, é possível fazer *download* de arquivos ou até de *softwares* como jogos.

Engine - "Motor", no contexto de jogos é "motor de jogo". Programa de computador que simplifica o desenvolvimento de jogos eletrônicos, possuindo várias funcionalidades para fazê-lo.

Feedback - "Retorno". É o procedimento que consiste no provimento de informação ao usuário sobre o desempenho, conduta, ou ação executada por este, objetivando reorientar ou estimular ações futuras mais adequadas ou diferentes das atuais.

Gameplay – “Jogabilidade”. É o jeito específico em que jogadores interagem com o jogo (digital). É o padrão definido dentro das regras do jogo, a conexão do jogador com o jogo, os desafios e a solução deles, e a conexão do jogador com o contexto.

Game Designer – Designer de Jogos. É a função, dentro de um projeto de jogo, de elaborar os conceitos principais e secundários, determinar como funcionará o *gameplay* e qual é o contexto/ história do jogo, documentar o desenvolvimento do projeto, entre outros. Um Designer de Jogos deve entender sobre todas as áreas envolvidas no projeto, mas ele não é um artista nem um programador.

Hardware – São as partes físicas, tangíveis de um computador (ou de um console de *video game*).

Hitbox – É uma forma invisível utilizada em jogos para detecção de colisão em tempo real. Uma *hitbox* acompanha um objeto ou “ator” no jogo, possibilitando feitos como o personagem do jogador ser acertado por um soco.

Idle – “Ocioso”. No contexto do *sprite*, é quando o personagem está parado e exibe uma animação que demonstra isso.

In-game – “Dentro do jogo”, quando o jogador está realmente jogando, isto é, não está em nenhum menu ou *cutscene*.

Multiplayer – “Multijogador”, um jogo que permite mais de um jogador (em determinadas modalidades, ou na “principal” do jogo), ou a premissa é ser jogado com múltiplas pessoas.

Pitch – A proposta de um jogo, normalmente apresentada para uma produtora de jogos ou investidor, para convencer que é um projeto promissor que vale ser financiado.

Placeholder – No contexto de jogos, é quando é implementado no jogo um *asset* que não é o definitivo, até este tomar o seu lugar.

Playtesting – É um processo no qual um *game designer* testa um jogo no intuito de encontrar *bugs* e falhas no design do jogo antes de seu lançamento. É uma parte estabelecida da prática de controle de qualidade.

Point-and-click - “Apontar e clicar”, título que descreve a mecânica principal de um subgênero dos jogos de aventura.

QA – Sigla de “Quality Assurance”, controle de qualidade.

Quest – É uma jornada em direção a um determinado objetivo, uma missão. No contexto de jogos, é quando o jogador recebe uma missão, consistindo em certas tarefas que deve cumprir.

Remake – É quando um jogo antigo é refeito utilizando melhor tecnologia do que a foi usada originalmente. Pode ser uma reinterpretação, mudando elementos do jogo, ou simplesmente melhorando os gráficos. Há muitas variações do que um *remake* pode ser, e é comum ser confundido com remasterização.

Right – “Direita”. No contexto do *sprite*, é quando o personagem se move para a direita.

Royalty-free – “Livre de *royalties*”, no contexto de música são músicas que podem ser utilizadas em outras mídias sem pagamento de direitos.

Scripting – É um tipo de linguagem de programação que inclui *scripts*, pequenos programas que automatizam a execução de certas tarefas.

Slot – “Espaço”, “abertura”.

Software – Comumente chamado de programa, é um conjunto de informações ou instruções que informa o computador como deve funcionar.

Sprites – É um termo de gráficos de computador que se refere a *bitmaps* (imagens) bidimensionais que integram uma cena maior. Cada frame de uma animação de contida em um jogo 2D é chamada de *sprite*.

Start/Quit – “Começar/Fechar”. Botões que normalmente aparecem em menus principais de jogos, permitindo o jogador começar a jogar ou fechar o programa (jogo).

Triple A – “Triplo A”, termo para designar jogos digitais de alto orçamento.

Up - “Cima”. No contexto do *sprite*, é quando o personagem se move para a parte superior do cenário, com as costas voltadas ao jogador.

Walk – “Caminhar”.

Wiki – É um *site* onde usuários cooperativamente modificam e estruturam conteúdo direto no navegador.