

UNIVERSIDADE FEDERAL DO RIO DE JANEIRO
INSTITUTO DE COMPUTAÇÃO
CURSO DE BACHARELADO EM CIÊNCIA DA COMPUTAÇÃO

PEDRO VÍTOR MARQUES NASCIMENTO

DESENVOLVIMENTO DE UM JOGO DIGITAL EDUCACIONAL PARA O ENSINO
DE PENSAMENTO COMPUTACIONAL CONCORRENTE

RIO DE JANEIRO
2023

PEDRO VÍTOR MARQUES NASCIMENTO

DESENVOLVIMENTO DE UM JOGO DIGITAL EDUCACIONAL PARA O ENSINO
DE PENSAMENTO COMPUTACIONAL CONCORRENTE

Trabalho de conclusão de curso de graduação
apresentado ao Instituto de Computação da
Universidade Federal do Rio de Janeiro como
parte dos requisitos para obtenção do grau de
Bacharel em Ciência da Computação.

Orientador: Prof. Geraldo Bonorino Xexéo
Co-orientador: Prof. Eduardo Freitas Mangeli de Brito

RIO DE JANEIRO

2023

CIP - Catalogação na Publicação

N244d Nascimento, Pedro Vítor Marques
Desenvolvimento de um jogo digital educacional
para o ensino de pensamento computacional
concorrente / Pedro Vítor Marques Nascimento. -- Rio
de Janeiro, 2023.
49 f.

Orientador: Geraldo Bonorino Xexéo.
Coorientador: Eduardo Freitas Mangeli de Brito.
Trabalho de conclusão de curso (graduação) -
Universidade Federal do Rio de Janeiro, Instituto
de Computação, Bacharel em Ciência da Computação,
2023.

1. Desenvolvimento de jogos sérios. 2.
Pensamento computacional. 3. Ensino. I. Xexéo,
Geraldo Bonorino, orient. II. Brito, Eduardo
Freitas Mangeli de, coorient. III. Título.


PEDRO VÍTOR MARQUES NASCIMENTO

DESENVOLVIMENTO DE UM JOGO DIGITAL EDUCACIONAL PARA O ENSINO
DE PENSAMENTO COMPUTACIONAL CONCORRENTE

Trabalho de conclusão de curso de graduação
apresentado ao Instituto de Computação da
Universidade Federal do Rio de Janeiro como
parte dos requisitos para obtenção do grau de
Bacharel em Ciência da Computação.

Aprovado em 1 de setembro de 2023

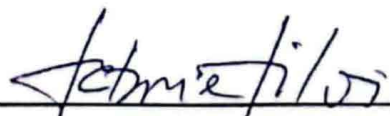
BANCA EXAMINADORA:



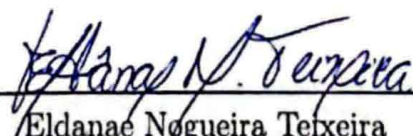
Geraldo Bonorino Xexéo
D.Sc. (COPPE/UFRJ)



Eduardo Freitas Mangeli de Brito
M.Sc. (Centro Universitário IBMEC)



Gabriel Pereira da Silva
D.Sc. (IC/UFRJ)



Eldanae Nogueira Teixeira
D.Sc. (IC/UFRJ)

À Graciete Saraiva, pelo apoio incondicional;
À Milton Alves, pela inspiração e encorajamento profissional;
À Vinicius Alves, pelo interesse e ajuda nos testes;
À Ana, pela motivação e companheirismo; e
À Maria e Bety, pela motivação diária!

AGRADECIMENTOS

Gostaria de agradecer aos participantes do laboratório LUDES em geral e em particular aos professores Geraldo Xexéo e Eduardo Mangeli. Em convivência com vocês, sempre tive grande aprendizado e encorajamento para me desenvolver academicamente. Destaque também para Rafael Studart, pela motivação em estudar Pensamento Computacional e possibilidade de aplicar na prática em sala de aula.

Agradecimentos ao meu querido amigo Vinícius Garcia, egresso do curso de Ciência da Computação, pela elaboração das artes do jogo, que fizeram muito sucesso com o público-alvo. Também ao meu irmão, Vinícius Alves, que compôs e produziu a música do jogo.

Agradecimentos também ao corpo docente do Instituto de Computação da UFRJ, que ao longos dos últimos anos me desenvolveu como profissional da computação e eterno aprendiz.

Por fim, destaco a importância fundamental do CNPq na minha trajetória como bolsista PIBIC.

"I learned a lot, by the end of everything. The past is past, now, but that's . . . you know, that's okay! It's never really gone completely. The future is always built on the past, even if we won't get to see it. Still, it's um, time for something new, now."

Fala do personagem Riebeck no jogo Outer Wilds

RESUMO

Este trabalho apresenta o processo de concepção e desenvolvimento do Mapa do Tesouro, um jogo eletrônico para o ensino de Pensamento Computacional Concorrente, um conceito derivado do Pensamento Paralelo, que por sua vez, seria um componente do Pensamento Computacional apresentado por Wing. O jogo desenvolvido tem como público alvo crianças de idade entre 4 e 10 anos. Os conceitos do PC Concorrente foram incorporados em processos do jogo, com a utilização da teoria da Retórica Procedimental. O jogo foi avaliado por trinta e nove crianças durante sua utilização em aulas de um curso extracurricular de Pensamento Computacional.

Palavras-chave: game design; retórica procedimental; pensamento computacional; jogos educacionais.

ABSTRACT

This work presents the process of conception and development of the Mapa do Tesouro, an electronic game for teaching Concurrent Computational Thinking, a concept derived from Parallel Thinking, which in turn, would be a component of Computational Thinking as presented by Wing. The developed game is targeted towards children aged between 4 and 10 years. The concepts of Concurrent CT were incorporated into the game's processes, using the theory of Procedural Rhetoric. The game was evaluated by thirty-nine children during its use in classes of an extracurricular Computational Thinking course.

Keywords: game design; procedural rethoric; computational thinking; serious games.

LISTA DE ILUSTRAÇÕES

Figura 1 – Diferença entre Concorrência e Paralelismo.	15
Figura 2 – Visão Macro do LUDES GD (MANGELI et al., 2021)	18
Figura 3 – Visão Detalhada (Micro) do LUDESGD (MANGELI et al., 2021)	19
Figura 4 – Screenshot do jogo Lightbot Hour	20
Figura 5 – Screenshot do jogo 7 Billion Humans	21
Figura 6 – Aplicativos relacionados	22
Figura 7 – Endo-GDC (TAUCEI, 2019)	23
Figura 8 – Aplicação do jogo Mapa do Tesouro em aula	26
Figura 9 – <i>Screenshots</i> das principais telas do jogo.	30
Figura 10 – Exemplos de fases do jogo	32

LISTA DE TABELAS

Tabela 1 – Resultados	28
---------------------------------	----

LISTA DE QUADROS

Quadro 1 – Grupos de Teste	27
Quadro 2 – Listagem de Mundos e Mecânicas Apresentadas	31
Quadro 3 – Listagem de Blocos de Comando Disponíveis no Jogo	33
Quadro 4 – Listagem de Blocos de Cenário Existentes no Jogo	34

SUMÁRIO

1	INTRODUÇÃO	12
1.1	OBJETIVO	13
1.2	ESTRUTURA DO TRABALHO	13
2	REFERENCIAL TEÓRICO	14
2.1	PENSAMENTO COMPUTACIONAL	14
2.2	PENSAMENTO COMPUTACIONAL CONCORRENTE	15
2.3	RETÓRICA PROCEDIMENTAL	16
2.4	JOGOS E APRENDIZADO	17
2.5	LUDES GD	18
3	TRABALHOS CORRELATOS	20
4	PROCESSO DE DESENVOLVIMENTO	23
4.1	CONCEPÇÃO	23
4.2	PROJETO	24
4.3	PRODUÇÃO	24
4.4	AValiação INTERNA	25
4.5	AValiação EXTERNA	25
4.6	EMPACOTAMENTO	28
5	MAPA DO TESOURO	29
5.1	VISÃO GERAL	29
5.2	ARTICULANDO ELEMENTOS DO JOGO E O PENSAMENTO COMPUTACIONAL CONCORRENTE	31
5.3	DETALHAMENTO DE ELEMENTOS DO JOGO	33
5.3.1	Comandos	33
5.3.2	Fases e Obstáculos	34
5.4	DIRETRIZES E SUGESTÕES DE USO	35
6	CONCLUSÃO	36
	REFERÊNCIAS	37
	APÊNDICE A – GDD DO JOGO	40

1 INTRODUÇÃO

Este trabalho apresenta os processos de concepção, desenvolvimento e avaliação do jogo educacional Mapa do Tesouro, desenvolvido para auxiliar crianças de 4 a 10 anos no aprendizado de Pensamento Computacional Concorrente (PC Concorrente), uma expansão proposta do conceito Pensamento Paralelo (GROVER; PEA, 2013).

Pensamento Computacional (PC) é uma habilidade que permite utilizar conceitos de Ciência da Computação para solucionar problemas e analisar sistemas. Wing (2006) argumenta que essa é uma competência fundamental para todos, não só para cientistas da Computação (WING, 2006). O desenvolvimento dessa competência foi incluído em currículos escolares de diversos países, com destaque para o Brasil (BRASIL, 2022) e o Reino Unido (United Kingdom, 2019). Na BNCC, se encontram competências como “reconhecer padrão de repetição em sequência de sons, movimentos, desenhos”¹ e “expressar as etapas para a realização de uma tarefa de forma clara e ordenada”².

Jogos são ferramentas populares para ensino (ECK, 2006). No caso do Pensamento Computacional, é possível encontrar diversos jogos com este objetivo (MONCLAR; SILVA; XEXÉO, 2018). Entretanto, como será descrito nas próximas seções, alguns tópicos do PC ainda possuem carência de jogos educacionais.

Foi utilizada a teoria da Retórica Procedimental (BOGOST, 2007) durante a fase de concepção, de modo a articular a mensagem que se pretende transmitir e as mecânicas presentes no jogo. Na etapa de idealização, as definições dos componentes de jogos do framework MDA (HUNICKE; LEBLANC; ZUBEK, 2004) foram utilizados e sustentaram a articulação entre sua mensagem e propósito *vis-à-vis* as mecânicas apresentadas.

Durante a etapa de desenvolvimento, utilizou-se um processo iterativo, com produtos intermediários e avaliações que permitiram correções e modificações no jogo para solucionar problemas e o aprimorar. Este processo segue relatado neste trabalho, bem como o detalhamento dos elementos do jogo e seus comandos, desafios e fases.

Ao final do processo, o jogo foi utilizado por 39 alunos, com idades entre 4 e 10 anos, que expuseram suas impressões a respeito da dificuldade encontrada no jogar, da disposição de repetir a experiência de jogar e de sua impressão sobre o prazer experimentado. As observações dos pesquisadores sobre as particularidades de cada grupo e uma análise qualitativa das respostas dos alunos está estruturada ao final deste trabalho.

¹ Competência EI03CO01

² Competência EI03CO02

1.1 OBJETIVO

O objetivo deste trabalho é apresentar em detalhes as etapas de desenvolvimento de um jogo com propósito, produzido com intuito de auxiliar o ensino de PC Concorrente para crianças entre 4 e 10 anos. Um objetivo secundário é descrever todo o processo como uma instância de aplicação do LUDES-GD e avaliar sua viabilidade como metodologia para desenvolvimento de jogos com propósito.

1.2 ESTRUTURA DO TRABALHO

Este trabalho se estrutura da seguinte forma:

- No Capítulo 2 estão descritos os principais pilares teóricos que embasam a construção do artefato.
- No Capítulo 3 são apresentados os resultados para a busca de jogos e aplicativos com objetivo relacionado.
- No Capítulo 4 o processo de desenvolvimento é detalhado, descrevendo as etapas seguidas e as principais decisões de projeto.
- No Capítulo 5 o artefato em sua versão final é apresentado, descrevendo suas mecânicas e outros elementos.
- Por fim, no Capítulo 6 são feitas as conclusões.

2 REFERENCIAL TEÓRICO

A base teórica deste trabalho se divide em quatro pilares: Pensamento Computacional; Retórica Procedimental; Uso de jogos para ensino; e a LUDES-GD, a Metodologia de desenvolvimento de jogos com propósito seguida.

2.1 PENSAMENTO COMPUTACIONAL

Wing (2006) apresentou o conceito Pensamento Computacional (PC), descrevendo-o como a habilidade que permitiria analisar sistemas, solucionar problemas e entender o comportamento humano utilizando conceitos comuns da Ciência da Computação (WING, 2006).

O conceito de PC foi influenciado significativamente pelo trabalho de Papert (1980), um dos pioneiros no uso de computadores na educação. Papert introduziu a ideia de que crianças poderiam aprender através da interação ativa com a tecnologia. Em seu livro “Mindstorms”, Papert argumenta que as crianças poderiam aprender conceitos complexos e desenvolver habilidades de pensamento crítico através da programação de computadores. Ele desenvolveu a linguagem de programação Logo como uma ferramenta para facilitar essa aprendizagem ativa (PAPERT, 1980).

O interesse da comunidade acadêmica pela proposta de Wing (2006) incitou diversos estudos com novas ideias e possíveis respostas a questões como quais seriam seus principais tópicos, ou como estes deveriam ser ensinados (GROVER; PEA, 2013; VICARI; MOREIRA; MENEZES, 2018). Em 2013, sete anos depois, Grover e Pea (2013) revisaram a literatura sobre PC e listaram os principais conceitos e definições que compõem a habilidade (GROVER; PEA, 2013). Uma das explicações destacadas é atribuída à *Royal Society*: “Pensamento Computacional é o processo de reconhecer aspectos computacionais do mundo em nossa volta e aplicar técnicas da Ciência da Computação para entender e analisar tanto sistemas artificiais quanto naturais”, em tradução livre. Neste mesmo artigo, Grover e Pea (2013) enumeram nove competências do PC:

- abstração e detecção de padrões;
- processamento sistemático de Informação;
- sistemas simbólicos e representações;
- noções algorítmicas de fluxo de controle;
- decomposição de problemas;
- pensamento recursivo, iterativo e paralelo;
- lógica condicional;
- eficiência e desempenho; e
- depuração e detecção de erro.

Kirkpatrick (2017) argumenta sobre a importância do chamado Pensamento Computacional Paralelo, relacionando-o com a habilidade de criar algoritmos que permitam a

execução de várias instruções em simultâneo, de maneira menos centralizada que soluções tradicionais e mais adequada à realidade dos computadores atuais. Kirkpatrick (2017) cita professores da área de computação que destacam os benefícios de desenvolver essa capacidade nos alunos, que passam a criar soluções com múltiplos processos independentes, utilizando melhor os recursos do computador, em vez de um único processo que executa as instruções passo-a-passo e de maneira linear.

2.2 PENSAMENTO COMPUTACIONAL CONCORRENTE

Pensamento Computacional Concorrente (PC Concorrente) é um subtópico do PC. Marques et al. (2021) propõem a seguinte definição:

O PC Concorrente é a habilidade de planejar soluções e entender problemas que apresentam diversos agentes, cada um com seus algoritmos próprios, que são executados de maneira sobreposta no tempo e podem ter dependências no uso de recursos compartilhados entre si.

A contribuição de Kirkpatrick (2017) com a descrição do Pensamento Paralelo, descrita na seção 2.1, possui interseções com a proposta de PC Concorrente, entretanto na área da computação existe uma distinção significativa entre concorrência e paralelismo.

Concorrência, em computação, acontece quando duas ou mais tarefas são executadas se sobrepondo no tempo (HANSEN, 1973), possivelmente no mesmo processador. Se diferencia de paralelismo, onde até mesmo uma tarefa pode ser executada em mais de um processador.

Isso pode acontecer de maneira intercalada, onde um processo começa e pode interromper sua execução para ceder a vez para o próximo; ou de uma maneira simultânea, na qual as tarefas executam suas instruções ao mesmo tempo em núcleos ou processadores diferentes. Paralelismo ocorre apenas quando as tarefas tem suas instruções executadas simultaneamente.

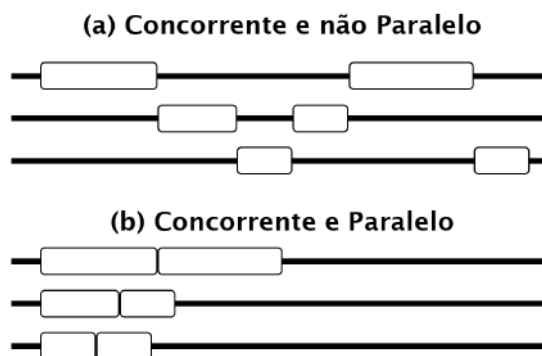


Figura 1 – Diferença entre Concorrência e Paralelismo.

Fonte: do autor

Concorrência, então, é um termo mais abrangente, e que abarca o paralelismo. A Figura 1 exemplifica a diferença: cada retângulo representa o fragmento de um processo sendo executado. As linhas horizontais representam a linha do tempo de execução de um processo específico. No primeiro exemplo da figura, temos três processos que não são executados em paralelo, pois só um fragmento é executado por vez. No segundo exemplo, os mesmos três processos são executados simultaneamente, como se não estivessem sido separados em partes. Nos dois exemplos os processos são concorrentes, pois suas execuções são sobrepostas no tempo.

Vicari, Moreira e Menezes (2018) em revisão bibliográfica sobre o PC concluem que academicamente, “o PC concorrente e não determinístico não é ainda explorado”. Indicam também que em sua busca, “o menor número de artigos trata do PC concorrente, 5 (cinco)”. No trabalho, Vicari, Moreira e Menezes (2018) considerou equivalentes PC concorrente e Pensamento Paralelo, de forma que as afirmações anteriores são em relação aos dois tópicos.

2.3 RETÓRICA PROCEDIMENTAL

A Retórica Procedimental envolve as maneiras pelas quais as regras e mecânicas de um sistema digital interativo podem moldar nossa compreensão de mensagens persuasivas. Em seu núcleo têm a ideia de que os sistemas digitais são, eles próprios, agentes persuasivos, capazes de moldar nossas crenças e atitudes por meio das regras e procedimentos que incorporam (BOGOST, 2008).

Jogos, por seu caráter intrinsecamente interativo, possuem forte potencial para a análise e aplicação deste tipo de retórica. Bogost (2007) considera que videogames possuem a capacidade de descrever (e modelar) o mundo através de processos e, por conta disso, a Retórica Procedimental seria sua maneira prática e efetiva de expressão (BOGOST, 2008).

Bogost (2008) utiliza como exemplo o gênero de jogos de simulação de naves: seus processos apresentam ao jogador uma aproximação de como é a experiência de pilotar uma aeronave. Essa aproximação da realidade consegue produzir no jogador reflexões sobre a complexidade de pilotar. Tornar o jogo mais realista, adicionando mais regras, influencia seus processos e pode incentivar o jogador a ser mais cauteloso (BOGOST, 2008).

A análise e construção de retóricas procedimentais corretas pode potencializar a característica educacional de jogos. Quando utilizada para definição de mecânicas e regras, ela potencialmente promove a aprendizagem por meio da exploração e experimentação dos processos dentro do jogo. Quando utilizada como teoria para análise do que é representado, pode permitir o descobrimento de argumentos contrários ao que se quer persuadir o jogador.

2.4 JOGOS E APRENDIZADO

Jogos são amplamente aceitos como uma importante ferramenta à disposição do educador para criar situações de aprendizagem (FISHMAN et al., 2014). De forma mais audaciosa, Prensky (2003) argumenta que jogos eletrônicos seriam o motor de uma revolução em curso no ensino, a geração de alunos e aprendizes que sempre experienciaram o mundo sob a onipresença da tecnologia acabaria por exigir a mudança das ferramentas utilizadas na educação e a efetiva utilização de jogos eletrônicos como recurso pedagógico.

Há uma clara ligação entre o que se deseja de uma situação de aprendizado e o que acontece com o jogador nos jogos, especialmente nos vídeo games. Entre as facetas positivas podemos citar, por exemplo, a motivação intrínseca e o engajamento gerados, e a imersão por longo tempo em um assunto, característica necessária para o aprendizado profundo (BECKER, 2007). Os jogos eletrônicos seriam a mídia ideal para promover o engajamento de estudantes nativos digitais e atender seu anseio de protagonismo no processo de aprendizagem (PRENSKY, 2003). Eck (2006) afirma que não apenas esses estudantes podem valer-se de jogos eletrônicos como ferramenta de aprendizado, e que a efetividade do processo de aprendizado baseado em jogos eletrônicos, *Digital Game-Based Learning – DGBL*, pode ser empregada no aprendizado de temas diversos para alunos de diferentes níveis de inclusão tecnológica.

Em *A Theory of Fun*, Koster (2013) argumenta que jogos, mesmo de entretenimento, “servem como uma ferramenta fundamental e poderosa de aprendizado” (KOSTER, 2013), sendo essa, na verdade, uma das formas mais importantes de encontrar diversão em jogos.

Já há algum tempo, esse quadro tem feito com que fossem criados jogos eletrônicos educacionais. Possivelmente o jogo eletrônico educacional de maior impacto na indústria, e um dos maiores sucessos comerciais entre jogos eletrônicos educacionais, foi *The Oregon Trail - Brøderbund* de 1985, baseado em um jogo textual com o mesmo nome de 1971, que buscava ensinar as crianças em idade escolar sobre a vida de pioneiros que se dirigiam ao Oregon no ano de 1847 (CAMPBELL, 2013).

Mais recentemente, com a larga distribuição de computadores pessoais, notebooks, *tablets* e *smartphones*, e com a geração nativa digital se impondo como um novo desafio educacional, o interesse por (vídeo) jogos educacionais aumentou muito (ECK, 2006). Entretanto, os desafios de desenvolvê-los repetidamente e com sucesso ainda não foram resolvidos. Este trabalho descreve um jogo desenvolvido dentro de um contexto de estudo sobre como desenvolver sistematicamente jogos com propósito.

2.5 LUDES GD

O LUDES GD é uma metodologia para o desenvolvimento de jogos com propósito construída por membros e colaboradores do LUDES¹, o Laboratório de Ludologia, Engenharia e Simulação do Programa de Engenharia de Sistemas e Computação da Coppe/UFRJ (MANGELI et al., 2021).

A ferramenta foi modelada a partir da detecção de padrões no processo de desenvolvimento de três jogos do LUDES. A análise ocorreu utilizando *logs* e entrevistas com os desenvolvedores (MANGELI et al., 2021).

O Mapa do Tesouro foi um dos jogos utilizados no mapeamento do processo, de forma que ao mesmo tempo que é uma instância de aplicação do LUDES GD, ele também é um dos argumentos para sua validação prática (MANGELI et al., 2021).

O LUDES GD possui uma divisão em cinco grandes etapas: Concepção, Projeto, Produção, Avaliação e Empacotamento. A metodologia segue um paradigma de ciclos iterativos, de forma que é possível percorrer as mesmas etapas várias vezes em momentos diferentes do desenvolvimento. A Figura 2 apresenta a visão Macro do LUDES GD.

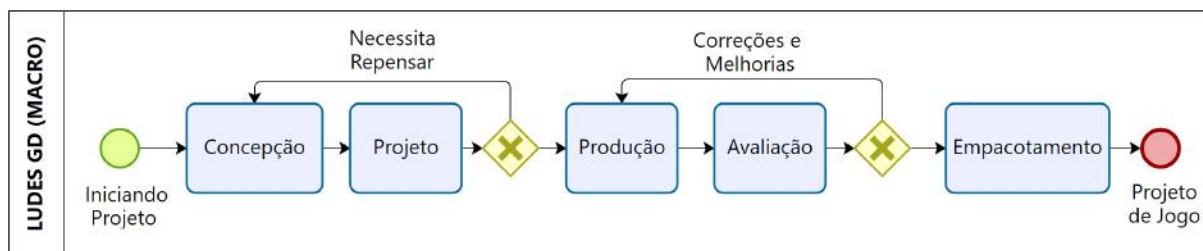


Figura 2 – Visão Macro do LUDES GD (MANGELI et al., 2021)

Cada grande etapa é dividida em etapas menores e mais específicas, algumas das quais são opcionais no processo ou mesmo são flexíveis em termos de ferramentas, deixando para os autores decidirem como executar. A Figura 3 apresenta a visão detalhada do LUDES GD.

A seguir, uma breve descrição das grandes etapas e seus passos internos:

- **Concepção:** Nesta etapa ocorre a definição do propósito do jogo. Partindo disso, é especificado um público-alvo e é feita a concepção dos principais conceitos e mecânicas do jogo.
- **Projeto:** O objetivo principal dessa etapa é preparar a documentação do projeto, que terá os detalhes de implementação e funcionalidades necessárias. O LUDES GD estimula documentação em ferramentas intuitivas, concisas e fáceis de alterar (exemplo um Canvas). Antes de documentar, essa etapa incentiva o uso de teorias e ferramentas para articular os elementos do jogo com os objetivos estabelecidos na etapa anterior (MDA, Retórica Procedimental, Endo-GDC, etc). Ao final, ocorre a

¹ <https://ludes.cos.ufrj.br/>

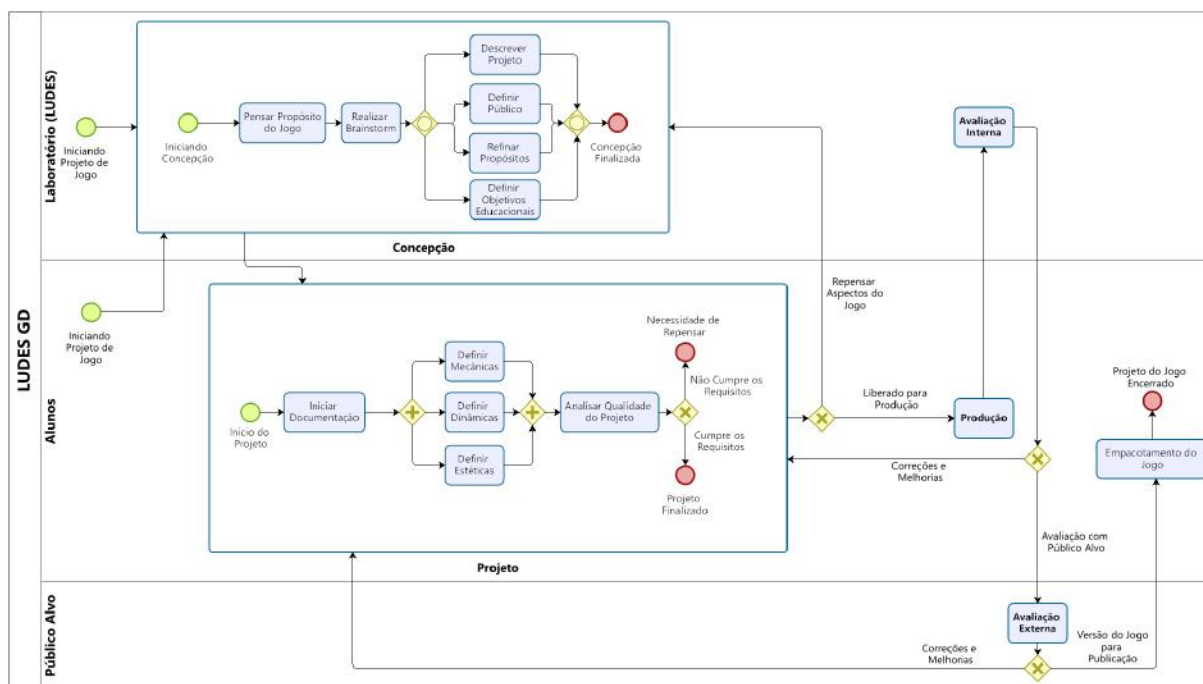


Figura 3 – Visão Detalhada (Micro) do LUDESGD (MANGELI et al., 2021)

avaliação da equipe para definir se os resultados estão adequados. Caso contrário, se recomeça a etapa de Projeto ou mesmo a etapa de Concepção.

- **Produção:** Durante essa etapa ocorre o desenvolvimento do jogo com base na documentação produzida. Por via de metodologias de desenvolvimento ágil, as funcionalidades listadas são transformadas em atividades (o *backlog*) que é consumido regularmente durante as *sprints* de produção. As atividades são priorizadas e separadas em entregas sequenciais. Dessa forma as atividades que formam o núcleo do jogo são executadas primeiro, enquanto detalhamento e polimento ao final.
- **Avaliação:** Nessa etapa o jogo produzido é avaliado em duas partes: internamente (pela equipe e membros do laboratório) e externamente (por uma amostra do público-alvo). Na Avaliação Interna os membros analisam a qualidade de uso, do projeto e da experiência. Na Avaliação Externa se observa, documenta e avalia o efeito do jogo no público-alvo em termos de aprendizado, divertimento e atingimento do propósito. Caso o jogo seja reprovado na etapa interna, a externa não é executada e o processo volta para etapas anteriores.
- **Empacotamento:** Por fim ocorre o lançamento do jogo para o público e a divulgação do conhecimento adquirido por meio de publicações científicas e eventos da área.

3 TRABALHOS CORRELATOS

O objetivo deste capítulo é apresentar os resultados da busca por softwares cuja aplicação seja compatível com o ensino de PC Concorrente. Foram utilizadas como bases de dados para a pesquisa o Google Scholar, a loja de jogos Steam ¹ e o site VideoGameGeek ². As consultas envolveram ocorrência de *game* com pelo menos uma das seguintes palavras-chave em inglês: code, programming, concurrent, concurrency, parallel, parallelism, algorithm e computational. No Google Scholar também utilizamos versões traduzidas das palavras-chave.

Os resultados iniciais das consultas foram analisados rapidamente pelo autor utilizando resumos, no caso de trabalhos científicos, e descrições no caso de jogos. Os resultados com maior compatibilidade passaram por uma leitura mais profunda, que após filtro rendeu poucos resultados que estão discutidos a seguir.

Monclar, Silva e Xexéo (2018) enumeram diversos jogos com propósito cujo objetivo é ensinar ou auxiliar o aprendizado de programação. Muitos dos jogos consistem em uma sequência de fases do tipo “colete todos os objetos da fase”, em que o jogador utiliza blocos de comando para controlar o personagem. O público-alvo são crianças a partir de 5 anos. Deste perfil pode-se destacar alguns jogos: Lightbot Hour (Lightbot Inc., 2019), Cato’s Hike (WAHBA, 1999), Move the Turtle (GREAT, 1999) e RoboZZle (OSTROVSKY, 1999). O objetivo comum é introduzir algoritmos e lógica de programação sequencial, de forma que as mecânicas principais são relacionadas à criação de funções, laços de repetição e alguns também incluem o uso de condições. Conceitos diretamente ligados à PC Concorrente não são abordados.

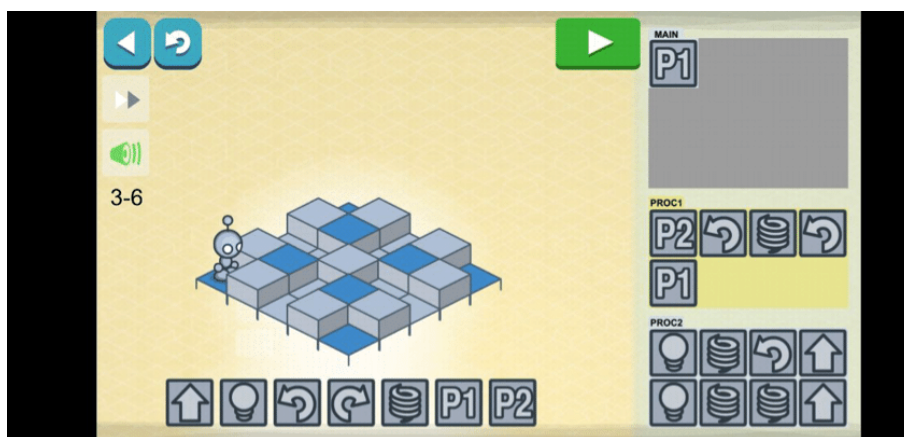


Figura 4 – Screenshot do jogo Lightbot Hour

Na *Steam*, loja online com amplo catálogo de jogos, foram feitas consultas utilizando palavras-chave e análises na categoria “programação”. Foram encontrados jogos com me-

¹ <https://store.steampowered.com/>

² <https://videogamegeek.com/>

cônicas relacionáveis ao PC Concorrente. Parte destes foram desenvolvidos pela *Zachtronics*, empresa com grande catálogo de jogos ligados à computação e engenharia. Seus produtos possuem classificação indicativa *T-rated* (ZACHTRONICS, 2019) (Apropriado para adolescentes de 13 anos ou mais).³ O motivo, fornecido pela própria empresa, é o uso de linguagem inapropriado, referências a drogas, sexo e violência (ZACHTRONICS, 2019), o que os torna inadequados como ferramenta para o ensino de PC Concorrente para crianças de idade entre 4 e 10 anos, o público alvo do jogo desenvolvido nesse trabalho.

Ainda nos resultados da *Steam*, foi encontrado o jogo “7 Billion Humans” (CORPORATION, 2018), que traz uma proposta que une um visual cartunesco e o controle de muitos personagens ao mesmo tempo, com um único código. Mesmo aparentando ter um público-alvo mais jovem, a dificuldade do jogo aumenta rapidamente, fazendo com os problemas alcancem um nível muito alto de complexidade, extrapolando o objetivo educacional para crianças.

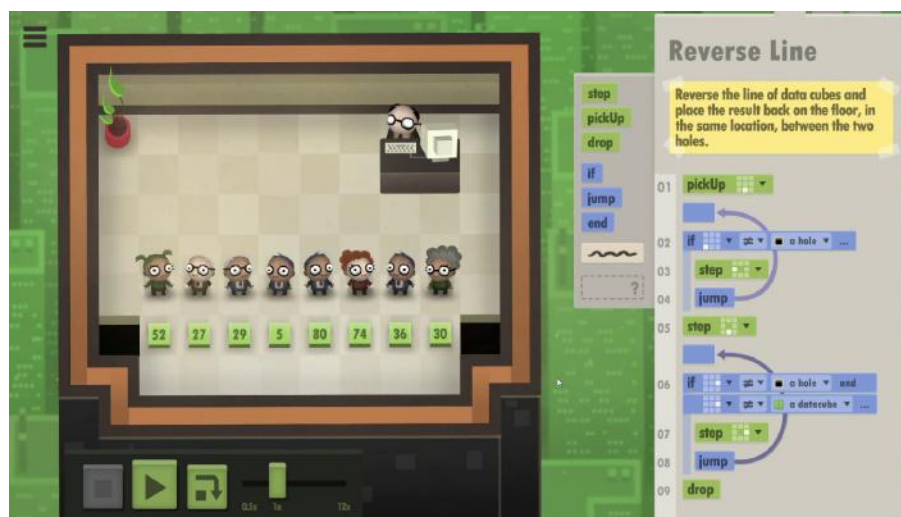
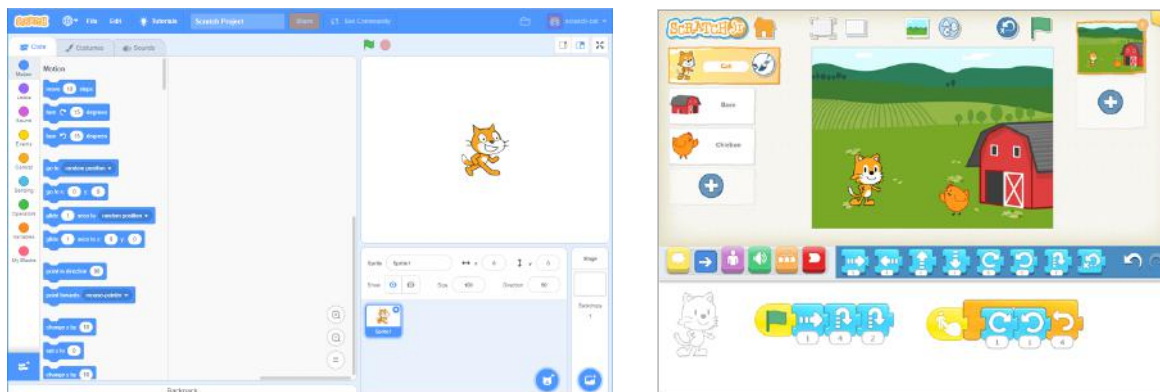


Figura 5 – Screenshot do jogo 7 Billion Humans

Foram encontrados também Aplicativos que visam introduzir jovens ao PC: o Scratch (Scratch Foundation, 2007), direcionado para crianças de idade entre 8 e 10 anos, e o Scratch Jr. (DevTech Reseach Group, 2014), uma versão para crianças entre 5 e 7 anos. Nestes aplicativos, os usuários podem criar animações ou jogos simples utilizando blocos de comando para definir o comportamento dos personagens.

O Scratch e Scratch Jr. são ferramentas consolidadas para a introdução e ensino de programação (FARIAS et al., 2019; FATOUROU et al., 2018). Em relação ao PC Concorrente, por cada personagem possuir uma sequência própria de comandos que é executada de maneira sobreposta, é possível utilizar as duas ferramentas para abordar alguns tópicos. Entretanto o Scratch é menos limitado, pois possibilita interação entre os personagens, permitindo inclusive seu uso para criação de ementas com noções de computação concorrente para crianças, como feito na Grécia por Fatourou et al. (2018).

³ <https://www.esrb.org/ratings-guide/>



(a) Screenshot do Scratch.

(b) Screenshot do Scratch Jr.

Figura 6 – Aplicativos relacionados

Os resultados discutidos indicam que para o público-alvo de criança entre 4 e 10 anos existe uma grande quantidade de jogos que se propõe a ensinar lógica de programação, entretanto uma lacuna de ferramentas que deem suporte ao ensino de PC Concorrente. Existe uma convergência de mecânicas, normalmente envolvendo a criação de sequências de blocos para formar algoritmos. A aceitação do público pode indicar um forte interesse por essa estrutura de jogo e seu potencial para engajamento de usuários e seu ensino. Essa base de mecânicas pode ser expandida para possibilitar suporte ao ensino de PC Concorrente, como será descrito nos próximos capítulos.

4 PROCESSO DE DESENVOLVIMENTO

Neste capítulo o processo de desenvolvimento do Mapa do Tesouro está descrito em detalhes. O texto está estruturado em seções, cada uma relacionada a uma etapa do LUDS-GD em sua visão geral.

4.1 CONCEPÇÃO

A etapa de Concepção foi iniciada pela identificação da carência de ferramentas de ensino que abordassem o ensino de Pensamento Computacional Concorrente para crianças. A partir da identificação deste problema, foi feita uma pesquisa por trabalhos correlatos, que permitiu identificar abordagens já existentes e, ao mesmo tempo, as falhas e oportunidades presentes nelas. A pesquisa de trabalhos teve metodologia e resultados detalhados no Capítulo 3.

Para o escopo de objetivos educacionais se escolheu introdução de lógica de programação, que se relaciona com a competência EI03CO02 da BNCC¹, o planejamento de “soluções e entendimento de problemas que apresentam diversos agente” e a exploração de desafios com “dependências no uso de recursos compartilhados entre si”. Os dois últimos objetivos relacionados diretamente ao PC Concorrente.

A partir disso, foi definido o público-alvo do jogo, que seriam crianças entre 4 e 10 anos. O amplo intervalo de idade representou uma decisão desafiadora, cujo objetivo foi com um único artefato abordar de forma introdutória lógica de programação até o desenvolvimento de conceitos do PC Concorrente.



Figura 7 – Endo-GDC (TAUCEI, 2019)

Com base na ferramenta Endo-GDC (TAUCEI, 2019), um *canvas* que auxilia a idealização de jogos com propósito, foi feito um *brainstorm* em conjunto com membros do

¹ EI03CO02 - “Expressar as etapas para a realização de uma tarefa de forma clara e ordenada”

laboratório LUDES para elencar possibilidades de conceitos para o jogo. Ao final, o *canvas* foi preenchido e utilizado como base em documentações futuras ².

A passagem na etapa de concepção teve fim em 20 dias e ocorreu apenas uma vez no processo.

4.2 PROJETO

A etapa de Projeto foi iniciada com a elaboração de um *Game Design Document* (GDD)³, documento que descreveu de forma explícita e estruturada todos os elementos mecânicos do jogo, suas fases iniciais e sua interface de usuário.

Durante a elaboração do GDD, decisões de projeto foram tomadas sobre o tema do jogo (Caça ao Tesouro), suas mecânicas principais e direção de arte. O Framework MDA (HUNICKE; LEBLANC; ZUBEK, 2004) foi utilizado para identificar dinâmicas e estéticas emergentes das mecânicas. Os elementos de dinâmica foram analisados utilizando a teoria da Retórica Procedimental (BOGOST, 2008). Os resultados da análise são abordados no Capítulo 5

Nesta etapa, as mecânicas do jogo, detalhadas no Capítulo 5, foram definidas para atendimento dos objetivos pedagógicos. O uso de blocos de comando foi escolhido para representar a criação de algoritmos e o planejamento de soluções. A possibilidade de mais de um pirata e sua sincronia com o bloco de Sinalização representam o conceito de múltiplos agentes com dependências de recursos. O número de mecânicas foi limitado para não adicionar grande complexidade ao jogo, visto o público-alvo de crianças.

A primeira passagem na etapa de Projeto teve duração de 13 dias, com a conclusão da documentação do projeto. As passagens seguintes envolveram apenas adaptações simples de definição de Interface de Usuário, de forma que as mecânicas principais do jogo não foram alteradas.

4.3 PRODUÇÃO

A etapa de Produção foi iniciada com a escolha das ferramentas de desenvolvimento. Unity (UNITY, 2019) foi escolhida como *game engine*. Os principais motivos englobaram a familiaridade do autor com a ferramenta e a grande quantidade de plataformas disponíveis para publicar o jogo (Android, WebGL, Windows, etc), dessa forma ampliar o acesso ao jogo não seria um obstáculo. Como versionador o GitHub foi utilizado, além de permitir documentações do projeto em sua Wiki integrada ao repositório.

² Infelizmente, o *canvas* preenchido não pode ser encontrado. Ele foi construído em papel físico, o que dificultou seu armazenamento e recuperação.

³ Disponível nos Apêndices

Ao longo de toda produção a organização seguiu algumas das características da metodologia ágil Scrum (SABBAGH, 2014): ciclos semanais de desenvolvimento, uso de *board* com atividades e reuniões regulares com orientador.

A produção do jogo teve uma forte característica iterativa, de forma que a etapa foi percorrida quatro vezes ao longo do projeto. Foram quatro passagens pela etapa:

1. **Protótipo Inicial:** produção do jogo sem visual elaborado, majoritariamente composto por texto e interação por botões. Duração Total de 7 dias.
2. **MVP:** produção do jogo com visual básico e mecânicas fundamentais. Sem polimento visual, mas com algumas das imagens para compor a atmosfera do jogo. Duração Aproximada de 5 meses.
3. **Versão Candidata:** expansão das funcionalidades do jogo, agora com visual finalizado, interface de usuário adequada e fases para o público-alvo jogar. Esta foi a primeira versão a seguir para Avaliação Externa. Duração Total de 30 dias.
4. **Versão Final:** implementação de melhorias baseadas na avaliação do público-alvo. Duração Aproximada de 2 meses.

4.4 AVALIAÇÃO INTERNA

Ao final de cada passagem pela etapa de Produção, o jogo passou por uma Avaliação Interna, com orientador e membros do laboratório. Cada passagem por esta etapa durou cerca de uma semana, totalizando quatro semanas aproximadamente.

Pela existência de reuniões regulares do laboratório LUDES, do qual o autor é membro, a cada semana o estado do jogo era apresentado aos participantes. Ao longo dessas reuniões, anotações foram tomadas livremente pelo autor, algumas se transformando em atividades no *board* para produção.

Ao final da produção do MVP, uma reunião focada no jogo foi marcada e sete membros jogaram as fases e emitiram suas opiniões sobre o jogo. O principal *feedback* recebido foi sobre uma falha do jogo ao adicionar comandos na lista usando um dispositivo com tela sensível ao toque, que prejudicaria usuários de *tablets* e celulares. Com isso, o jogo voltou para produção, com meta de aperfeiçoar a interface visual do jogo. A inspiração escolhida foi o aplicativo Scratch Jr (DevTech Reseach Group, 2014).

4.5 AVALIAÇÃO EXTERNA

A partir da versão Candidata do jogo, o jogo passou pela primeira vez por Avaliação Externa, cuja função foi analisar o engajamento dos usuários com o artefato em três aspectos de interesse:

- Dificuldade encontrada ao jogar;
- Disposição de repetir a experiência; e
- Impressões sobre a diversão experimentada.

O escopo desta avaliação não envolveu analisar a capacidade educacional do jogo, mas se restringiu a averiguar o interesse dos usuários pela experiência interativa.

Nesta avaliação, 39 crianças fizeram parte de 3 grupos testes no total, um para cada faixa etária: 4 anos, 5 e 6 anos e 7 à 9 anos. O agrupamento dos alunos por idade foi feita utilizando a separação já definida de turmas do colégio. É importante destacar que todos os participantes já possuíam cerca de 15 horas de prática extracurricular em um curso semanal de PC. A ementa do curso era baseada em atividades “desplugadas”⁴ com intuito de desenvolver habilidades do Pensamento Computacional. No Quadro 1 os grupos e quantidade de alunos está destacado.

A passagem por esta etapa durou duas semanas, que corresponderam com a organização e execução das sessões de teste.



Figura 8 – Aplicação do jogo **Mapa do Tesouro** em aula

As sessões de teste tiveram duração entre 30 e 50 minutos, variando de acordo com a disponibilidade de horário da turma. Com exceção das crianças de 4 anos, cada uma jogou individualmente em um *tablet*, sem apresentação do jogo pelo professor. O professor estava disponível para dúvidas, mas evitava interferir na experiência e ficava como observador. O grupo mais novo jogou a maior parte do tempo com apenas um *tablet*, segurado pelo professor enquanto o mesmo direcionava a experiência por meio de perguntas para os alunos.

Durante os testes, o autor observou e documentou os comentários que surgiam e as dificuldades encontradas pelas crianças. As observações dos testes estão separadas por grupos e apresentadas no Quadro 1.

Todas as crianças se mostraram interessadas pela temática pirata, algumas até explicitamente dizendo que acharam o personagem “fofo”. Acreditamos que parte da facilidade

⁴ Do Inglês, *Unplugged*, no sentido de não utilizarem computadores, mas peças e materiais físicos.

Quadro 1 – Grupos de Teste

Grupo	Idade	Qnt.	Observações
A	4 anos	5	A melhor abordagem para esta idade foi manter o professor gerenciando o jogo e pedindo sugestões de movimentos para os alunos enquanto os auxiliava em qualquer dificuldade. As crianças apresentaram dificuldade com a diferença entre os blocos de Andar e Olhar (no jogo é preciso olhar na direção desejada antes de andar). Os alunos apresentaram dificuldade em entender o tutorial do início do jogo, que se revelou muito dependente do texto, foi necessária uma demonstração feita pelo professor. Ficou patente a necessidade de algumas fases iniciais para que o jogador aprenda o funcionamento dos blocos de Andar e Olhar, já adicionadas na última versão do jogo, e uma adaptação do tutorial inicial.
B	5 e 6 anos	9	As crianças já eram alfabetizadas, mesmo assim não se interessaram em ler as mensagens de tutorial. Entretanto, não foi necessário que o professor interviesse para auxiliar no entendimento do funcionamento do jogo, exceto nas fases mais avançadas, nas quais os desafios são maiores.
C	7, 8 e 9 anos	25	

de entender o objetivo do jogo foi consequência do tema escolhido. As crianças assimilaram rápido que o personagem pirata deveria andar para cima dos “X” no mapa e cavar para achar tesouros.

Uma confusão recorrente foi a dificuldade em diferenciar a função do bloco andar, que é sempre para a direção para a qual o pirata olha, com os blocos de olhar em uma direção, que executam apenas uma rotação. Isso foi melhorado na versão final do jogo.

Por conta do tempo e dificuldade do jogo, a maior parte das crianças não chegou até o final, algumas não atingindo as fases que trabalham diretamente o PC Concorrente. Na versão final o jogo teve sua dificuldade diminuída, principalmente nas fases iniciais.

Ao final das sessões de teste três perguntas simples foram feitas para os participantes dos testes, todas de resposta sim ou não, alinhadas com os aspectos de engajamento de interesse.

1. “Gostou do Jogo?”
2. “Você achou difícil?”
3. “Jogaria novamente?”

A escolha binária foi determinada pela idade das crianças no intuito de evitar difi-

culdades de interpretação de uma escala mais complexa como as de *Likert*. A pequena quantidade de perguntas foi uma limitação do tempo disponível na sessão de testes. O resultado pode ser conferido na Tabela 1.

Tabela 1 – Resultados

Pergunta	SIM	NÃO
Gostou do Jogo?	39 (100%)	0 (0%)
Achou difícil?	19 (48,7%)	20 (52,3%)
Jogaria Novamente?	39 (100%)	0 (0%)

As observações indicam a necessidade de uma flexibilização do jogo para lidar com as faixas etárias diferentes e com mecanismos mais visuais ou sonoros para lidar com as crianças. Tutorial e apresentação de mecânicas em forma de animações foram incluídas na versão final do jogo.

4.6 EMPACOTAMENTO

Finalizado o último ciclo produtivo, o jogo foi lançado nas seguintes plataformas:

- Google Play (Sistema Operacional Android) ⁵
- Itch.io (*Browsers*)⁶

O código fonte foi disponibilizado no GitHub ⁷ com acesso aberto e gratuito. Além disso, o processo do jogo foi apresentado em eventos de divulgação da UFRJ como a SIAC 2019 e 2021. Além disso, os resultados parciais do trabalho foram publicados em um simpósio da área de jogos, a SBGames (MARQUES et al., 2021).

⁵ <https://play.google.com/store/apps/details?id=com.LUDES.MapaTesouro&hl=pt>

⁶ <https://tanookihouse.itch.io/mapa-do-tesouro>

⁷ <https://github.com/TanookiVerde/Mapa-do-Tesouro>

5 MAPA DO TESOURO

O jogo Mapa do Tesouro foi concebido com o intuito de ampliar a oferta de ferramentas de ensino de PC Concorrente para crianças entre 4 e 10 anos.

Nesta seção a versão desenvolvida do jogo será descrita em detalhes. As descrições estão divididas em três partes: a introdução ao tema e as mecânicas principais do jogo; análise dos processos utilizando a Retórica Procedural; e por fim detalhamento dos elementos do jogo.

5.1 VISÃO GERAL

A temática escolhida para o artefato foi a pirataria, com o uso de uma paleta de cores viva. A decisão foi tomada com a intenção de tornar o jogo atraente ao público infantil.

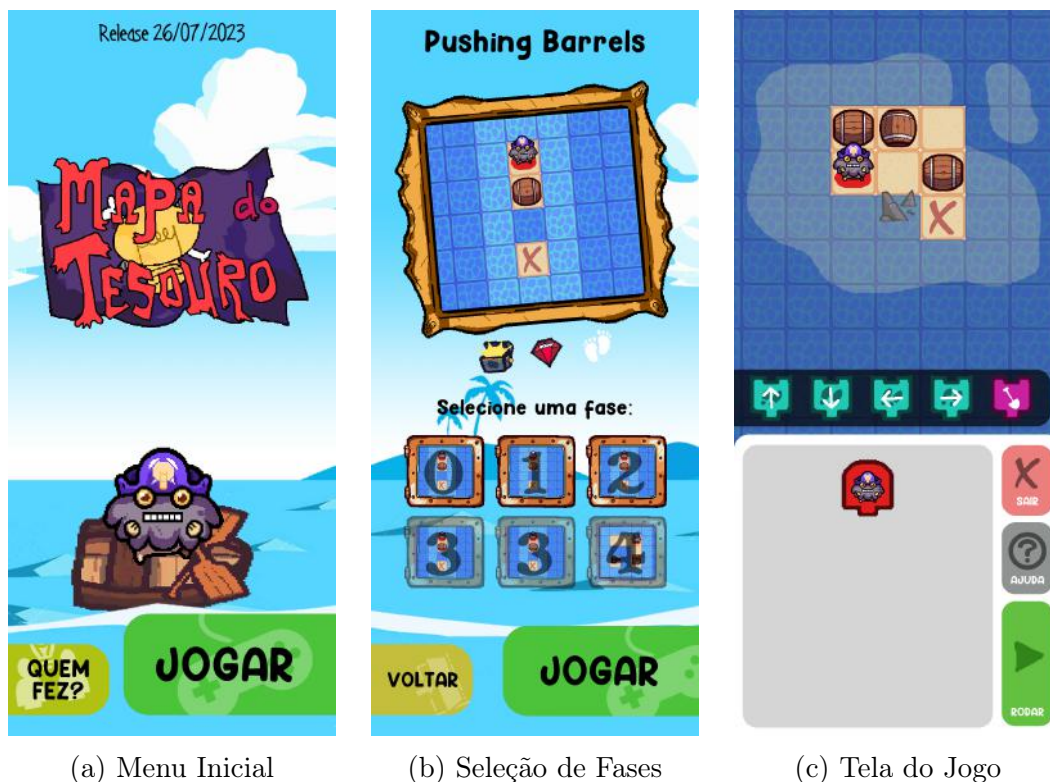
Como forma de ilustrar o progresso no jogo, ele foi dividido em mapas (ou mundos) contendo várias ilhas, cada uma representando uma fase e tem como objetivo encontrar um ou mais baús de tesouro.

Ao longo das fases, o jogador controla um ou dois piratas, limitados a andar em um mapa quadriculado, como podemos ver na Figura 9c. Os piratas são individualmente programados pelo jogador a partir da construção de uma sequência de passos simples, como andar ou cavar. Uma ação de levantar bandeira permite sincronizar os piratas.

As mecânicas do jogo foram escolhidas para permitir a representação de processos relacionados ao PC Concorrente, como a existência de múltiplos agentes, cada um com sua própria sequência de comandos. A sincronização entre piratas permite ao jogador representar a dependência do uso de recursos e a entrada cuidadosa em “zonas críticas”.

As fases do jogo representam desafios rápidos que possuem três aspectos de complexidade:

- **Encontrar Todos os Baús:** a condição obrigatória para completar uma fase é desenterrar todos os baús. Baús enterrados apresentam um “X” grande na areia. Dessa forma, cada fase possui um objetivo claro.
- **Encontrar Todas as Joias:** algumas fases possuem objetos opcionais coletáveis, que podem estar escondidos. Desenterrá-los pode dificultar o caminho e exigir algoritmos maiores. Essa regra motiva o jogador a visitar algumas fases para explorar outros locais e criar um novo algoritmo.
- **Quantidade de Passos Ótima:** todas as fases possuem uma quantidade ótima de passos, que representam a menor quantidade de blocos possível para completar o objetivo obrigatório. Esta regra motiva o jogador a otimizar seus passos usando a menor quantidade de recursos possível, uma atividade típica de computação.



(a) Menu Inicial

(b) Seleção de Fases

(c) Tela do Jogo

Figura 9 – Screenshots das principais telas do jogo.

Para movimentar os personagens, o jogador deve primeiro criar uma sequência de comandos a serem executados. Cada comando é representado como uma peça de um quebra-cabeça, o que indica que devem ser encaixados um a seguir do outro.

Nas fases iniciais, o jogador controla apenas um pirata e vai sendo apresentado a problemas que exigem caminhos, e programas, cada vez mais complexos. Mais tarde, nas fases de programação concorrente, o jogador controla dois piratas, cada um com uma sequência de comandos própria, ou seja, cada um executando uma tarefa própria. Ao clicar em executar, o jogo decide aleatoriamente, a cada iteração, qual pirata terá seu movimento executado. Algumas fases obrigam os dois personagens a se ajudar e a interagir. A apresentação de mecânicas e desafios do jogo é feita de forma progressiva e está organizada no Quadro 2.

Começamos o jogo com fases de execução linear (apenas um personagem) para garantir que qualquer criança, até aquelas que não tenham aprendido conceitos básicos de algoritmos, consigam superar os desafios e atingir as fases que apresentam problemas de natureza concorrente (com mais de um personagem).

As mecânicas do jogo são flexíveis, o que permite soluções criativas, mas ao mesmo tempo aumenta a possibilidade de construir algoritmos com erro. O jogo utiliza dois mecanismos de *feedback* visual para dar dicas aos jogadores:

- **Indicador de Bloco Atual:** o bloco atual sendo executado fica em destaque com cor escura, de forma que o jogador consiga acompanhar em qual comando o código

Quadro 2 – Listagem de Mundos e Mecânicas Apresentadas

ORDEM	MUNDO	DESCRIÇÃO
0	Olá, Mundo!	Introdução das mecânicas básicas do jogo, de criação de sequências de blocos, execução e teste.
1	Barris Caídos	Apresentação da mecânica de empurrar barris caídos para liberar o caminho. As fases exploram as limitações na direção dos barris.
2	Entre Pontes e Segredos	Este mundo explora o potencial do barril de virar uma ponte quando em contato com o mar. As fases demandam que o jogador planeje com antecedência o uso de barris.
3	En(caixa)do	Apresentação dos blocos de Estilingue, que permitem destruir Caixas Velhas a distância. Essa mecânica será essencial para interação dos piratas no próximo mundo.
4	Trabalho em Equipe	Introdução da necessidade de interação e sincronia entre os piratas. O comando de Sinalização fica disponível pela primeira vez.
5	Fases Perdidas	Nenhuma mecânica é apresentada. Este mundo representa diversas fases desenvolvidas, mas que não entraram na campanha principal do jogo.

Fonte: Elaborado pelo Autor.

está quando um erro acontece.

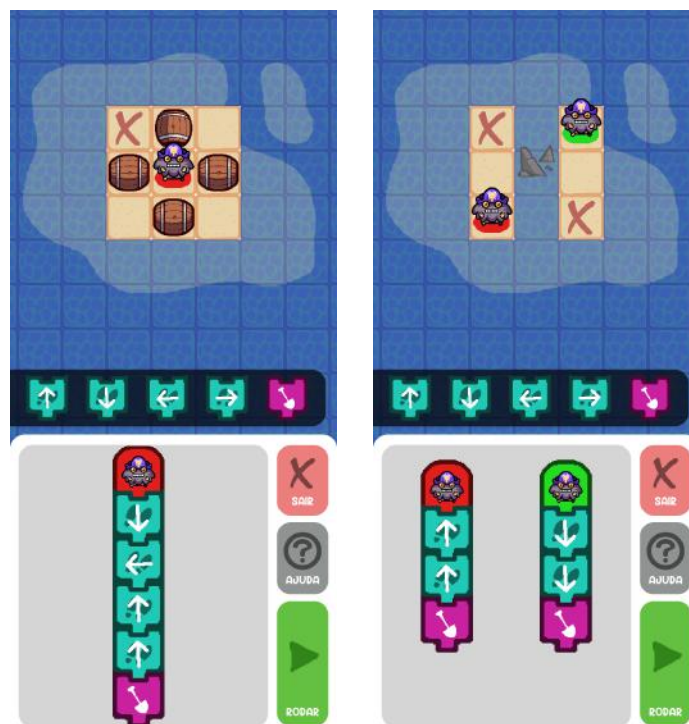
- **Sinalizador de Bug:** quando tenta executar uma ordem impossível, o pirata sinaliza o erro com um barulho e símbolo visual.

Intencionalmente o jogo não possui regras com intuito de punir ou limitar o jogador (perda de corações, limitação de tempo, etc). O objetivo é tornar a experiência acessível e construir um ambiente confortável para experimentação.

Por fim, o jogo foi idealizado e desenvolvido para permitir tanto utilização individual e independente quanto a utilização guiada em sala de aula.

5.2 ARTICULANDO ELEMENTOS DO JOGO E O PENSAMENTO COMPUTACIONAL CONCORRENTE

Na seção seção 2.3 a Retórica Procedimental foi apresentada, junto com sua relevância no contexto de jogos educacionais para dar suporte ao *game design*. Sob sua influência, os elementos e processos do Mapa do Tesouro foram concebidos considerando seu alinhamento com o propósito educacional do jogo.



(a) Fase linear.

(b) Fase concorrente.

Figura 10 – Exemplos de fases do jogo

Ao criar uma sequência de blocos, controlar até dois personagens, e buscar por tesouros o jogador atua sobre o jogo e, ativamente, contribui para seu protagonismo no processo de aprendizagem.

Abaixo estão detalhadas as principais relações entre os processos do jogo com conceitos de PC e PC Concorrente.

Criar sequência de blocos Ao sequenciar blocos, ou seja, criar uma ordem de comandos, o jogador está planejando suas ações. Pensar antes de agir é importante no contexto do Pensamento Computacional (PC).

Controlando um personagem Para dar comandos para um personagem, o jogador deve entender o contexto ao redor, considerando o posicionamento do personagem, obstáculos e o objetivo. Do ponto de vista do PC, também é essencial que se entenda o contexto do que está sendo manipulado, seja um personagem, um banco de dados ou um registrador.

Encontrando tesouros Encontrar os tesouros é o mesmo que solucionar um problema. Nem sempre o acesso pelo mapa é fácil e muitas vezes é preciso de criatividade. Este processo, em conjunto com o uso de comandos, é a aplicação padrão de algoritmos, pois é necessário que o jogador utilize uma sequência de comandos para achar a solução.

Controlar dois personagens Ao controlar dois personagens, o jogador deve entender

dois contextos de posição, obstáculos e objetivos diferentes. Deve pensar se os personagens vão interagir e como vão interagir. Como a ordem de execução deles não é previsível, é definida aleatoriamente, o jogador precisa aprender a sincronizá-los usando o bloco da bandeira. Este é o processo mais relacionado com o PC Concorrente.

5.3 DETALHAMENTO DE ELEMENTOS DO JOGO





Os processos apresentados pelo jogo foram materializados por elementos alinhados a proposta da estética visual e organizados em fases de complexidade e dificuldade crescentes. A Quadro 3 e a Quadro 4 apresentam descrições e representações pictóricas dos comandos e obstáculos, respectivamente.

5.3.1 Comandos

Os comandos que podem ser escolhidos pelo jogador, cada um representando uma ação simples dos personagens, são apresentados na Tabela 3. Foram escolhidos comandos simples e relacionados ao tema do jogo, para que o público-alvo entendesse facilmente.

Uma das decisões foi não utilizar blocos de repetição ou condicionais, pois o objetivo era ampliar o público-alvo para que as fases concorrentes pudessem ser desfrutadas até pelos mais novos. Se fossem adicionados, se acredita que os desafios concorrentes poderiam ficar complexos demais para o público-alvo.

Quadro 3 – Listagem de Blocos de Comando Disponíveis no Jogo

ÍCONE	NOME	DESCRIÇÃO
	Andar	O personagem anda na direção que estiver olhando. Se o personagem esbarrar em obstáculos fixos, ou tentar andar sobre a água, o jogo emite uma mensagem de erro.
	Atirar	O projétil disparo pelo estilingue viaja na direção que o personagem estiver olhando. A munição consegue ativar canhões.
	Cavar	O personagem usa sua pá para cavar o ponto em que está. Se nenhum tesouro for encontrado o jogo emite uma mensagem de erro.
	Sinalizar	Ao levantar a bandeira, o pirata pausa sua execução. Quando seu colega também levanta a bandeira, os piratas abaixam-nas ao mesmo tempo e a execução continua normalmente.

Fonte: Elaborado pelo Autor.






5.3.2 Fases e Obstáculos

Cada fase é composta por um mapa quadriculado de duas dimensões, formado por:

- um conjunto de blocos de chão, em que o personagem pode andar;
- um conjunto de blocos de água, em que o jogador só consegue de mover se estiver em uma ponte; e
- um conjunto de obstáculos, apresentados no Quadro 4.

Os obstáculos foram desenvolvidos para estimular o planejamento. Através de reflexão, o jogador deve entender que é possível manipular o estado dos objetos (Barril pode virar uma ponte, Caixa pode ser destruída, etc). Comandos do algoritmo podem se basear em um estado de fase que não é o inicial, pois o jogador pode manipulá-lo, por exemplo criando uma ponte onde não havia.

Quadro 4 – Listagem de Blocos de Cenário Existentes no Jogo

ÍCONE	NOME	DESCRIÇÃO
	Caixa Velha	A caixa velha é um objeto não ultrapassável e fixo. Pode ser destruída com um tiro de estilingue. Caixas podem esconder a posição de tesouros enterrados.
	Barril	Barril é um objeto que pode ser empurrado se o jogador o fizer no sentido correto. Pode virar uma ponte se empurrado sobre a água. Barris podem esconder a posição de tesouros enterrados.
	Pedras	Objetos fixos no chão que não podem ser destruídos nem ultrapassados. Elas podem impedir a movimentação de barris.
	Baú Enterrado	Localização de um Baú Enterrado. Cavar em cima desta posição permite encontrar o baú.
	Joia Enterrada	Localização de uma joia opcional. Cavar em cima desta posição permite encontrar uma joia.
	Ponte	A jangada representa uma ponte que um pirata pode utilizar para atravessar partes do mar. Este objeto pode ser construído ao empurrar um barril no mar.

Fonte: Elaborado pelo Autor.

5.4 DIRETRIZES E SUGESTÕES DE USO

O Mapa do Tesouro pode ser utilizado como atividade prática em sala de aula ou como material adicional para desenvolver habilidades relacionadas à:

- Competência EI03CO02 da BNCC, que consiste em expressar as etapas para a realização de uma tarefa de forma clara e ordenada;
- Exploração de desafios com dependências no uso de recursos compartilhados entre si; e
- Planejamento de soluções e entendimento de problemas que apresentam diversos agentes.

Baseado nos resultados iniciais da avaliação do jogo, se recomenda a seguinte abordagem de acordo com a idade do público-alvo:

- **Menores que 5 anos:** o professor deve explicar o jogo e ele mesmo controlá-lo. Ao longo das fases, ir pedindo sugestões para os alunos e criando perguntas sugestivas para explorar as mecânicas do jogo.
- **Entre 5 e 6 anos:** o professor pode distribuir o jogo para cada estudante, mas deve apresentar as mecânicas básicas primeiro, principalmente sobre acessar fases, criar sequências de blocos e rodar o programa. O professor deve ficar disponível para dúvidas e direcionamento dos alunos. Se recomenda que ele incentive a construção gradativa do código, sempre executando em sequências intermediárias.
- **Maiores que 6 anos:** o professor precisa intervir pouco, apenas para dúvidas nas últimas fases do jogo, quando o PC Concorrente é aplicado de forma desafiadora.

6 CONCLUSÃO

Este trabalho apresenta a descrição e explicitação do processo de criação de um jogo educacional utilizando a retórica procedimental para construção dos argumentos que sustentam os objetivos de aprendizagem.

O Mapa do Tesouro é um passo inicial no propósito de prover jogos para ensinar PC Concorrente para crianças entre 4 e 10 anos. As avaliações iniciais sobre a usabilidade e engajamento do artefato são positivas e permitiram a criação de sugestões de uso. Contudo, a efetividade do jogo como ferramenta de ensino merece ser avaliada em um trabalho futuro.

Sob aspecto de idealização de um jogo educacional, a utilização da retórica procedimental foi uma ferramenta útil para garantir que os processos apresentados construíssem os argumentos que sustentam os seus objetivos de aprendizagem, além de não impactar negativamente os objetivos pedagógicos.

A formalização do processo em um diagrama (Figura 2) torna explícita a divisão do desenvolvimento em ciclos, que incluíram testes e avaliações. Esta estratégia possibilitou que fossem detectados e solucionados problemas nas etapas iniciais de projeto, além de permitir o aprimoramento a partir das avaliações recebidas.

Por fim, a atuação no projeto foi uma ótima experiência prática e acadêmica para o autor. Houve divulgação e discussão de resultados parciais em simpósio de jogos (SBGames) e eventos da UFRJ, além de disponibilização gratuita para o público e utilização em sala de aula.

REFERÊNCIAS

- BECKER, K. Pedagogy in commercial video games. In: **Games and Simulations in Online Learning: Research and Development Frameworks**. Hershey, PA: Idea Group, 2007. p. 21–47. Citado na página 17.
- BOGOST, I. **Persuasive games: The expressive power of videogames**. Cambridge, Massachusetts.: MIT Press, 2007. Citado 2 vezes nas páginas 12 e 16.
- BOGOST, I. The rhetoric of video games. In: SALEN, K. (Org.). **The ecology of games: Connecting youth, games, and learning**. Cambridge, MA: MIT Press, 2008. p. 117–140. Citado 2 vezes nas páginas 16 e 24.
- BRASIL. **Computação: Complemento à BNCC**. Brasília: Ministério de Educação, 2022. Acessado em 19/7/2023. Disponível em: <http://portal.mec.gov.br/docman/fevereiro-2022-pdf/236791-anexo-ao-parecer-cneceb-n-2-2022-bncc-computacao/file>. Acesso em: 2023-07-19. Citado na página 12.
- CAMPBELL, C. **The Oregon Trail was made in just two weeks**. Polygon, 2013. Disponível em: <https://www.polygon.com/2013/7/31/4575810/the-oregon-trail-was-made-in-just-two-weeks>. Citado na página 17.
- CORPORATION, T. **7 Billion Humans**. 2018. Disponível em: <https://tomorrowcorporation.com/7billionhumans>. Acesso em: 2019-07-09. Citado na página 21.
- DevTech Reseach Group. **Scratch Jr**. 2014. Disponível em: <https://www.scratchjr.org/>. Acesso em: 2019-06-03. Citado 2 vezes nas páginas 21 e 25.
- ECK, R. V. Digital game-based learning: It’s not just the digital natives who are restless. **EDUCAUSE review**, Educause, v. 41, n. 2, p. 16, 2006. Citado 2 vezes nas páginas 12 e 17.
- FARIAS, C. et al. Estimulando o pensamento computacional: uma experiência com scratchjr. In: SBC. **Anais do XXV Workshop de Informática na Escola**. Brasília, 2019. p. 197–206. Citado na página 21.
- FATOUROU, E. et al. Teaching concurrent programming concepts using scratch in primary school: Methodology and evaluation. **International Journal of Engineering Pedagogy (iJEP)**, v. 8, n. 4, p. 89–105, 2018. Citado na página 21.
- FISHMAN, B. et al. **Empowering Educators: Supporting Student Progress in the Classroom with Digital Games**. Ann Arbor: University of Michigan, 2014. Citado na página 17.
- GREAT, N. is. **Move the Turtle**. 1999. Disponível em: <http://movetheturtle.com/>. Acesso em: 2019-06-03. Citado na página 20.
- GROVER, S.; PEA, R. Computational thinking in k–12: A review of the state of the field. **Educational researcher**, Sage Publications Sage CA: Los Angeles, CA, v. 42, n. 1, p. 38–43, 2013. Citado 2 vezes nas páginas 12 e 14.

- HANSEN, P. B. **Operating System Principles**. Upper Saddle River, NJ, USA: Prentice-Hall, Inc., 1973. ISBN 0-13-637843-9. Citado na página 15.
- HUNICKE, R.; LEBLANC, M.; ZUBEK, R. Mda: A formal approach to game design and game research. In: **Proceedings of the AAAI Workshop on Challenges in Game AI**. Menlo Park, CA: The AAAI Press, 2004. v. 4, n. 1, p. 1722. Citado 2 vezes nas páginas 12 e 24.
- KIRKPATRICK, K. Parallel computational thinking. **Communications of the ACM**, v. 60, p. 17–19, 11 2017. Citado 2 vezes nas páginas 14 e 15.
- KOSTER, R. **Theory of fun for game design**. Sebastopol, CA: OReilly Media, Inc., 2013. Citado na página 17.
- Lightbot Inc. **Lightbot Hour**. 2019. Disponível em: <http://lightbot.com/hour-of-code.html>. Acesso em: 2019-06-03. Citado na página 20.
- MANGELI, E. et al. Metodologia para desenvolvimento de jogos com proposito de um laboratorio de ludologia. In: SBC. **Anais Estendidos do XX Simpósio Brasileiro de Jogos e Entretenimento Digital**. Online, 2021. p. 143–151. Citado 3 vezes nas páginas 8, 18 e 19.
- MARQUES, P. et al. Desenvolvimento de um jogo digital educacional para o ensino de pensamento computacional concorrente. In: SBC. **Anais Estendidos do XX Simpósio Brasileiro de Jogos e Entretenimento Digital**. Online, 2021. p. 68–75. Citado 2 vezes nas páginas 15 e 28.
- MONCLAR, R. S.; SILVA, M. A.; XEXÉO, G. Jogos com propósito para o ensino de programação. In: SBC, XVII., 2018, Foz do Iguaçu. **Anais do Simpósio Brasileiro de Games e Entretenimento Digital - SBGames**. Foz do Iguaçu, 2018. p. 1132–1140. Citado 2 vezes nas páginas 12 e 20.
- OSTROVSKY, I. **RoboZZle**. 1999. Disponível em: <http://robozzle.com/>. Acesso em: 2019-06-03. Citado na página 20.
- PAPERT, S. A. **Mindstorms: Children, computers, and powerful ideas**. New York, NY: Basic books, 1980. Citado na página 14.
- PRENSKY, M. Digital game-based learning. **Computers in Entertainment (CIE)**, ACM, v. 1, n. 1, p. 21–21, 2003. Citado na página 17.
- SABBAGH, R. **Scrum: Gestão ágil para projetos de sucesso**. São Paulo: Editora Casa do Código, 2014. Citado na página 25.
- Scratch Foundation. **Scratch**. 2007. Disponível em: <https://scratch.mit.edu/>. Acesso em: 2019-06-03. Citado na página 21.
- TAUCEI, B. B. **ENDO-GDC: Desenvolvimento de um Game Design Canvas para Concepção de Jogos Educativos Endógenos**. Dissertação (Mestrado) — UFRJ/COPPE/Programa de Engenharia de Sistemas e Computação, Rio de Janeiro, 2019. Citado 2 vezes nas páginas 8 e 23.

United Kingdom. **National curriculum in England: computing programmes of study (Statutory Guidance)**. Department for Education, 2019. Disponível em: <https://www.gov.uk/government/publications/national-curriculum-in-england-computing-programmes-of-study>. Acesso em: 2019-07-09. Citado na página 12.

UNITY. **Unity 3D**. 2019. Disponível em: <https://unity.com/pt>. Acesso em: 2019-06-03. Citado na página 24.

VICARI, R. M.; MOREIRA, A. F.; MENEZES, P. F. B. Pensamento computacional: revisão bibliográfica. 2018. Citado 2 vezes nas páginas 14 e 16.

WAHBA, H. **Cato's Hike**. 1999. Disponível em: <http://hwahba.com/catoshike/>. Acesso em: 2019-06-03. Citado na página 20.

WING, J. M. Computational thinking. **Communications of the ACM**, v. 49, n. 3, p. 33–35, 2006. Citado 2 vezes nas páginas 12 e 14.

ZACHTRONICS. **Zachademics**. 2019. Disponível em: <http://www.zachtronics.com/zachademics/>. Acesso em: 2020-08-10. Citado na página 21.

APÊNDICE A – GDD DO JOGO



Mapa do Tesouro

Game Design Document

Sumário

Sumário	2
Resumo	3
Plataformas	3
Objetivos Pedagógicos	3
Público-alvo	3
Mecânicas Básicas	4
Interface de Usuário	4
Fases	5
Fases Concorrentes	6
Elementos de Fase	7
Blocos	8
Fluxo de telas	9

Resumo

O jogo Mapa do Tesouro tem o objetivo de ampliar a oferta de ferramentas de ensino de PC Concorrente para crianças entre 4 e 10 anos. Neste jogo é possível comandar até dois piratas usando blocos de comando e ajudá-los a desenterrar todos os tesouros.

Plataformas

- Navegadores (Chrome, Firefox, ...)
- Android

Objetivos Pedagógicos

1. Introdução de algoritmos
2. Planejamento de soluções e entender problemas que apresentam diversos agente
3. Exploração de desafios com dependências no uso de recursos compartilhados entre si

Público-alvo

- Crianças entre 4 e 10 anos de idade

Mecânicas Básicas

O jogador inicialmente controla um personagem dando para ele uma sequência de comandos. Depois de terminada esta sequência, o usuário pode dar início à execução das instruções quando quiser, igual a um programador quando escreve um script e escolhe quando executar.

Os comandos são responsáveis por movimentar o personagem pelo mapa, fazê-lo andar, atirar, cavar, etc.

O jogo se passa em um tabuleiro retangular de tamanho variável de acordo com as fases. É sempre em uma visão superior. No mapa podem existir diversos tipos de obstáculos: barril, pedras, água, ponte, etc. Cada obstáculo possui uma maneira diferente de interação com o jogador.

O principal objetivo do jogador é encontrar os **grandes tesouros** (ou baú) da fase. Outro objetivo (menor) é encontrar **pequenos tesouros** (ou jóias). Todos os tipos de tesouro ficam escondidos embaixo da areia e o jogador deve identificar sua localização por um "X" para então poder bolar uma solução.

Cada fase pode ser concluída integralmente (Coletando todos os tesouros) ou parcialmente (Coletando pelo menos os **grandes tesouros** e não todos os **pequenos tesouros**). Não existem necessariamente soluções exatas, portanto o jogador pode submeter qualquer tentativa. Ela será correta se encontrar o tesouro final e não tiver erros (por exemplo, encostar na água).

O jogo possui também fases que o usuário controla dois personagens e cada um deles possui uma sequência de instruções própria. A execução desses dois "Scripts" é feita praticamente ao mesmo tempo, instrução à instrução. Outras mecânicas são adicionadas nesses tipos de fase para permitir a sincronia entre os piratas. Elas serão melhor definidas em uma futura sessão.

Interface de Usuário

Para controlar o personagem o jogador deve primeiro criar uma sequência de instruções. A partir de um comando "Rodar" o jogador pode executar os comandos em cadeia.

Para adicionar uma instrução na lista, basta arrastar os blocos usando o dedo (mobile com touchscreen) ou mouse (computadores). É possível também apagar blocos. A usabilidade é semelhante ao Scratch Jr.



Ao lado do cenário, o jogador vê representado a lista dos comandos adicionados na sequência. Ele pode editar livremente, menos quando está executando o código.

Quando o jogador acreditar que a sequência de instruções dadas é suficiente, ele pode clicar em "Rodar" e executar. Existem algumas respostas que o sistema pode dar ao jogador ao receber uma sequência de instruções.

- **Sequência interrompida:** O usuário encostou em algum elemento indevido (tipo fatal). O jogo mantém a sequência enviada e diz exatamente qual linha foi responsável pelo erro.
- **Sequência errada:** Mesmo seguindo o caminho corretamente, o usuário não conseguiu encontrar o baú principal, portanto resposta errada. A sequência se mantém.
- **Sequência correta:** Se ainda faltar algum baú, o sistema pergunta se jogador deseja tentar mais uma vez. Caso contrário só volta para o menu de fases.

Fases

O jogo é separado em fases curtas, cada uma com um baú principal (grande tesouro) e um ou mais baús opcionais (pequenos baús), ambos escondidos debaixo da areia. Para liberar a próxima fase o jogador deve coletar o baú principal da fase anterior a mesma.

Uma fase normalmente é composta por um quadriculado 4x4, 6x6 ou 8x6 que possui elementos organizados (ou obstáculos). Uma descrição completa de cada um destes elementos pode ser encontrada na seção **Elementos das fases**.

Um exemplo de organização de fase (4x4):



O jogador começa em algum ponto do mapa (depende do level). A fase termina quando o baú principal for encontrado pelo jogador.

Fases Concorrentes

Nas fases paralelas, o jogador controla dois personagens, vermelho e verde. Cada um deles possui uma lista independente de comandos que são executados de forma paralela, praticamente ao mesmo tempo.

Um comando adicional é liberado: Sinalizar. Enquanto estiver esperando, a execução de comandos do personagem entra em hiato e só volta a executar quando o outro personagem faz o mesmo. A animação de esperar o aliado é representada pelo personagem levantando uma bandeira.

Elementos de Fase

Em cada fase, uma combinação dos elementos abaixo podem aparecer:

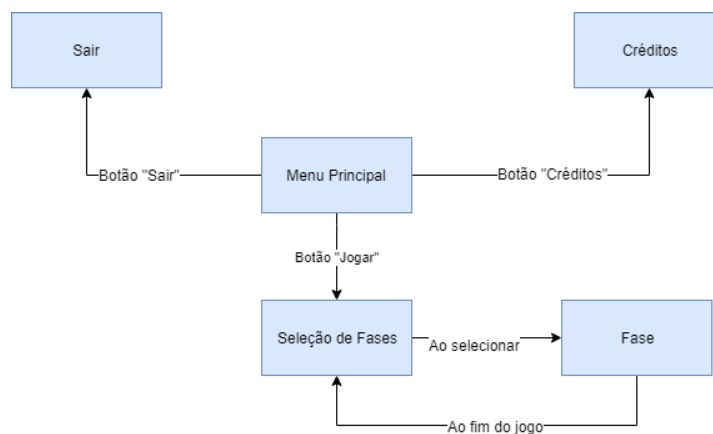
Nome do Obstáculo	Descrição
Barril	Barril é um objeto que pode ser empurrado se o jogador o fizer no sentido correto. Pode virar uma ponte se empurrado sobre a água. Existem dois tipos de barril: vertical e horizontal, a diferença impacta no eixo que pode ser empurrado. Barris podem esconder a posição de tesouros enterrados.
Caixa de Madeira	A caixa velha é um objeto não ultrapassável, fixo no chão e destrutível com tiros de estilingue. Caixas podem esconder a posição de tesouros enterrados.
Pedras	Objetos fixos no chão que não podem ser destruídos nem ultrapassados. Elas podem impedir a movimentação de barris.
Ponte	A jangada representa uma ponte que um pirata pode utilizar para atravessar partes do mar. Este objeto pode ser construído ao empurrar um barril no mar.
Chão	Local seguro para caminhar do pirata.
Mar	Representa a água, elemento que o pirata nunca pode estar sobre diretamente, a não ser que tenha uma ponte.
Baú Enterrado	Localização de um Baú Enterrado. Cavar em cima desta posição permite encontrar o baú.
Joia Enterrada	Localização de uma joia opcional. Cavar em cima desta posição permite encontrar uma joia.

Blocos

Ao longo das fases os seguintes blocos ficam disponíveis:

Nome do Bloco	Descrição
Andar	Variações: Cima, Baixo, Direita, Esquerda O personagem anda na direção que estiver olhando. Se o personagem esbarrar em obstáculos fixos, ou tentar andar sobre a água, o jogo emite uma mensagem de erro.
Atirar	Variações: Cima, Baixo, Direita, Esquerda Uma semente é disparada pelo estilingue que pode destruir caixas.
Cavar	O personagem usa sua pá para cavar o ponto em que está. Se nenhum tesouro for encontrado o jogo emite uma mensagem de erro.
Sinalizar	Ao levantar a bandeira, o pirata espera, sem agir, até que seu colega também levante a bandeira. Quando as duas bandeiras estiverem levantadas, os piratas abaixam-nas ao mesmo tempo e a execução continua normalmente.

Fluxo de telas



Listagem de Assets

A lista inicial de assets necessários para o jogo:

Nome	Tipo
Animação Idle Robô	Arte
Animação Andando Robô	Arte
Animação Sinalizando	Arte
Animação Cavando	Arte
Base de Bloco de Comando	Arte
Ícone: Cavar	Arte
Ícone: Andar	Arte
Ícone: Atirar	Arte
Ícone: Sinalizar	Arte

Nome	Tipo
Obstáculo: Chão	Arte
Obstáculo: Água	Arte
Obstáculo: Ponte	Arte
Obstáculo: Pedra	Arte
Obstáculo: Caixa	Arte
Obstáculo: Barril X	Arte
Obstáculo: Barril Y	Arte
9 Slice básico UI	Arte
Fundo Menu Principal	Arte
Música de Menu	Música
Música de Jogo	Música
Som de Bloco Encaixando	Música
Som de Achar tesouro	Música