



UNIVERSIDADE FEDERAL DE PERNAMBUCO  
CENTRO DE EDUCAÇÃO  
PROGRAMA DE PÓS-GRADUAÇÃO EM EDUCAÇÃO MATEMÁTICA E  
TECNOLÓGICA  
CURSO DE DOUTORADO

TARCISIO ROCHA DOS SANTOS

**A ENGENHARIA DIDÁTICO-INFORMÁTICA E O PROCESSO DE PRODUÇÃO DE  
JOGOS MATEMÁTICOS DE SIMULAÇÃO PARA SITUAÇÕES DIDÁTICAS: um  
estudo da concepção de uma versão digital do jogo Mankala Colhe Três**

Recife

2023

TARCISIO ROCHA DOS SANTOS

**A ENGENHARIA DIDÁTICO-INFORMÁTICA E O PROCESSO DE PRODUÇÃO DE JOGOS MATEMÁTICOS DE SIMULAÇÃO PARA SITUAÇÕES DIDÁTICAS: um estudo da concepção de uma versão digital do jogo Mankala Colhe Três**

Tese apresentada ao Programa de Pós-Graduação em Educação Matemática e Tecnológica da Universidade Federal de Pernambuco, como requisito parcial para a obtenção de título de doutor em Educação Matemática e Tecnológica. Área de concentração: Ensino de Ciências e Matemática.

Orientador: Prof. Dr. Franck Gilbert René Bellemain

Coorientador: Prof. Dr. Péricles de Lima Sobreira

Recife

2023

Catálogo na fonte:  
Bibliotecária: Leonice Cavalcante, CRB4:2198

S237e	<p>Santos, Tarcisio Rocha dos</p> <p>A engenharia didático-informática e o processo de produção de jogos matemáticos de simulação para situações didáticas: um estudo da concepção de uma versão digital do jogo Mankala Colhe Três / Tarcisio Rocha dos Santos. – 2023.</p> <p>174 p. : il.</p> <p>Orientação: Franck Gilbert René Bellemain.</p> <p>Doutorado (Tese) – Universidade Federal de Pernambuco, Centro de Educação. Programa de Pós-graduação em Educação Matemática e Tecnológica. Recife, 2023.</p> <p>Inclui referências.</p> <p>1. Engenharia didático-informática. 2. Desenvolvimento de softwares educativos. 3. Jogo matemático de simulação. 4. Mankala Colhe Três. 5. Situações didáticas. I. Bellemain, Franck Gilbert René (orientador). II. Título.</p> <p>370 CDD (23.ed.)</p>	UFPE (CE2024-046)
-------	--	-------------------

TARCISIO ROCHA DOS SANTOS

**A ENGENHARIA DIDÁTICO-INFORMÁTICA E O PROCESSO DE PRODUÇÃO DE JOGOS MATEMÁTICOS DE SIMULAÇÃO PARA SITUAÇÕES DIDÁTICAS: um estudo da concepção de uma versão digital do jogo mankala colhe três**

Tese apresentada ao Programa de Pós-Graduação em Educação Matemática e Tecnológica da Universidade Federal de Pernambuco, como requisito parcial para a obtenção de título de doutor em Educação Matemática e Tecnológica. Área de concentração: Ensino de Ciências e Matemática.

Aprovado em: 23/08/2023

**BANCA EXAMINADORA**

---

Prof. Dr. Franck Gilbert René Bellemain (Orientador e Presidente)  
Universidade Federal de Pernambuco

---

Profa. Dra. Rosinalda Aurora de Melo Teles (Examinadora Interna)  
Universidade Federal de Pernambuco

---

Prof. Dr. Ricardo Tiburcio dos Santos (Examinador Externo)  
Prefeitura da Cidade do Recife

---

Profa. Dra. Rogéria Gaudêncio do Rego (Examinadora Externa)  
Universidade Federal da Paraíba

---

Prof. Dr. Saddo Ag Almouloud (Examinador Externo)  
Universidade Federal do Pará

Dedico este trabalho primeiramente ao Soberano Deus, “Porque dele, e por meio dele, e para ele são todas as coisas. A ele, pois, a glória eternamente. Amém!” (BÍBLIA, Romanos 11, 36).

À minha esposa, Mirela, e ao meu filho, Samuel.

## AGRADECIMENTOS

Ao Rei eterno, imortal, invisível, Deus único, honra e glória pelos séculos dos séculos. Amém!

À minha linda e amada esposa, Mirela, por seu amor, pelo auxílio verdadeiro durante o período de doutorado e de toda nossa vida. Pelo seu olhar amoroso, precioso e sábio, tornando cada momento vivido mais significativo, leve e bonito. Por ser presente em todos os momentos; nos bons, tornando-os melhores, e nos mais difíceis, me auxiliando a passar por eles sempre confiantes nos planos do Senhor. Por ser a melhor esposa que Deus poderia me oferecer, por já amar tanto nosso filho Samuel, hoje em seu ventre e que enche nossos dias de alegria, esperança e gratidão ao Senhor. Amo vocês.

Aos meus pais, Anselmo e Ana, maiores inspirações de honestidade, amor e dedicação. Pessoas simples e sábias, que sempre priorizaram minha educação em todos os aspectos e nunca cessaram de me amar profundamente, a quem ofereço meu amor e respeito.

Aos meus irmãos, Tiago e Taciana, irmãos incríveis, que sempre me serviram de inspiração e que me enchem de orgulho.

Aos meus sogros, Marcos e Alessandra, e minhas cunhadas, Milena e Melina, pessoas que acompanharam de perto todo esse processo de doutorado, sempre me dando todo apoio possível, a quem tenho muito amor e gratidão.

Ao meu orientador, Franck, pessoa admirável, que me inspira e que permitiu, da melhor forma possível, que a presente pesquisa pudesse ser desenvolvida de forma tão madura e autoral. Maior exemplo de forma de lidar com a pesquisa científica e seus frutos para a sociedade. Agradeço o privilégio da parceria. A quem tenho muita admiração e gratidão.

Ao meu coorientador, Péricles. Inicialmente, pela oportunidade de viabilizar, de prontidão, a realização de um verdadeiro sonho, que foi a experiência do estágio doutoral no exterior. Pelo privilégio da parceria. Por todas as contribuições em torno da Engenharia de Softwares, que possibilitou o desenvolvimento de fundamentos da presente pesquisa, e pela disponibilidade de acompanhar “de perto” o avanço da pesquisa. Pela acolhida no período de estágio doutoral no Canadá, momentos marcantes da minha vida e de minha esposa. Espero um dia poder retribuir tanto carinho a você e a sua linda família. A quem possuo toda gratidão.

À professora Paula Baltar, pelas incontáveis contribuições na minha formação acadêmica e, em especial, nas aulas de Seminários do doutorado, sempre trazendo as melhores contribuições possíveis para a pesquisa, com um olhar apurado e cuidadoso que só ela possui.

Maior exemplo de compromisso com a educação pública e de qualidade, a quem possuo todo o respeito, admiração e gratidão.

Ao colega de grupo de pesquisa e membro das bancas de Qualificação e Defesa da Tese Ricardo Tiburcio, pesquisador admirável, que me inspirou na escolha do tema da presente pesquisa e que hoje tenho o privilégio de chamar de colega de pesquisa. Pela sua excelente contribuição na formalização da Engenharia Didático-Informática, que me permitiu dar continuidade às suas investigações. A quem tenho muita admiração e gratidão.

Aos demais colegas do grupo de pesquisa Atelier Digitas, por tantas contribuições ao longo da jornada de desenvolvimento desta tese de doutorado. Grupo que tenho orgulho de compor e que tenho toda gratidão.

Aos professores Rogéria Gaudêncio, Rosinalda Teles e Saddo Almouloud pelo privilégio de tê-los nas bancas de Qualificação e Defesa da Tese, apresentando contribuições excelentes para a presente pesquisa, a quem possuo toda admiração, respeito e gratidão.

À CAPES, pela bolsa no Programa de Doutorado Sanduíche no Exterior, sem a qual não seria possível viver a valorosa experiência em Gatineau (Québec-CA), através da Université du Québec en Outaouais (UQO), e por meio da qual pude conhecer pessoas excepcionais, como o professor Karim El Guemhioui, o qual também possuo grande gratidão pela acolhida carinhosa e pela viabilização dos estudos no âmbito da UQO.

Aos grandes amigos Dona Adelaide, Gilles e Dominique, que tive o privilégio de conhecer durante a estadia em Gatineau. Pessoas que marcaram minha vida por meio de um acolhimento verdadeiro e de uma amizade preciosa. A quem tenho muito carinho e gratidão.

À toda equipe de técnicos do EDUMATEC, em especial, Clara, por toda assistência prestada ao longo desse período de doutorado, sempre de forma providencial, cuidadosa e afetiva; e a todos os professores do EDUMATEC, em especial, aos que me ministraram disciplinas, em que tive o privilégio de aprender mais sobre educação.

Aos colegas do Colégio de Aplicação, professores José Carlos, Lúcia Durão, Marcos Melo e Rogério Ignácio, que colaboraram da melhor forma possível no período do desenvolvimento desta pesquisa, em que minha carga horária de aulas esteve reduzida e, em especial, durante o período de realização de doutorado sanduíche no exterior, em que estive integralmente afastado das atividades do colégio. Toda minha admiração, respeito e gratidão.

## RESUMO

A problemática em torno da qualidade de softwares educativos e seus processos de produção vem sendo amplamente discutida no ambiente da Educação Matemática e, partindo em busca de contribuir com esse debate, propomos dar continuidade às questões trazidas por pesquisas que formalizaram a Engenharia Didático-Informática (EDI) como uma metodologia de produção de software educativo. A EDI foi concebida para dar suporte à Engenharia de Softwares Educativos por meio da articulação dos princípios metodológicos da Engenharia de Softwares Educativos com os da Engenharia Didática e apresenta, como produto, seu modelo de processo de software educativo. Buscando trazer novas contribuições no que se refere à problemática da qualidade de softwares educativos e, em particular, à Engenharia Didático-Informática, propomos, na presente pesquisa, dar seguimento à discussão trazida por pesquisadores que se debruçaram sobre a formalização da EDI ao buscar responder questões que envolvem as primeiras impressões de utilização em sua versão mais recente e a análise de particularidades de seu uso na concepção de um jogo matemático de simulação para situações didáticas, que representa um software de tipologia diferente da que a EDI tem sido utilizada (ambientes de simulação/micromundos). Partimos da hipótese que a concepção de uma versão digital do jogo Mankala Colhe Três, baseada nos princípios teórico-metodológicos da Engenharia Didático-Informática, fornecerá elementos suficientes para a proposição de um modelo de processo de software da EDI destinado à produção de jogos matemáticos de simulação para uso em situações didáticas, que representa nosso objetivo geral. Buscando atender esse objetivo, a metodologia da pesquisa será conduzida a partir da concepção de uma versão digital do jogo Mankala Colhe Três seguindo a realização das fases presentes no modelo de processo da EDI, de modo a obter informações sobre particularidades desse processo de produção e, com isso, propor um modelo de processo específico para a produção de softwares educativos dessa tipologia. Após a análise da realização das etapas iniciais da Engenharia Didático-Informática na concepção do jogo, nos deparamos com a necessidade de um detalhamento maior sobre a realização da sua etapa de Desenvolvimento, que viabilize ainda mais o uso dessa metodologia de produção de software educativo para os educadores matemáticos interessados em produzir jogos matemáticos digitais. Essa nova problemática nos remeteu a um novo objetivo de pesquisa, integrado ao objetivo inicial, e que se remete à construção de uma metodologia para a etapa de Desenvolvimento da EDI que forneça um percurso para transformação dos requisitos em código a ser implementado no desenvolvimento de jogos matemáticos de simulação para situações didáticas. A partir de então, construímos uma

metodologia de auxílio à etapa de Desenvolvimento da EDI para a produção de jogos matemáticos de simulação para situações didáticas, baseada em princípios da Engenharia de Software voltados à modelagem de sistemas. Após a apresentação e exemplificação de uso dessa metodologia, apresentamos detalhadamente um modelo de processo de software da EDI que considere as contribuições da análise da concepção do Mankala Colhe Três e da metodologia construída para auxiliar a fase de Desenvolvimento da EDI.

**Palavras-chave:** Engenharia Didático-Informática. Desenvolvimento de Softwares Educativos. Jogo Matemático de Simulação. Mankala Colhe Três. Situações Didáticas.

## RÉSUMÉ

La question de la qualité des logiciels éducatifs et de ses processus de production a été largement débattue dans le milieu de l'Enseignement des Mathématiques et, cherchant à contribuer à ce débat, nous proposons de poursuivre les enjeux soulevés par les recherches qui ont formalisé l'Ingénierie Didactique-Informatique (EDI) comme un Méthodologie de production de logiciels éducatifs. L'EDI a été conçu pour soutenir l'Ingénierie de Logiciel Éducatif à travers l'articulation des principes méthodologiques de l'Ingénierie de Logiciel Éducatif avec ceux de l'Ingénierie Didactique et présente, en tant que produit, son modèle de processus logiciel éducatif. Cherchant à apporter de nouvelles contributions au regard de la problématique de la qualité des logiciels éducatifs et, en particulier, pour l'Ingénierie Didactique-Informatique, nous proposons, dans la présente recherche, de donner suite à la discussion apportée par des chercheurs qui se sont focalisés sur la formalisation de l'EDI en cherchant des réponses aux questions qui impliquent les premières impressions d'utilisation de sa version la plus récente et l'analyse des particularités de son utilisation dans la conception d'un jeu de simulation mathématique pour des situations didactiques, qui représente un logiciel d'une typologie différente de celle EDI travaille sur (environnements de simulation/micromondes). Nous partons de l'hypothèse que la conception d'une version numérique du jeu Mankala Colhe Três, basée sur les principes théorico-méthodologiques de l'Ingénierie Didactique-Informatique, fournira suffisamment d'éléments pour la proposition d'un modèle de processus logiciel de l'EDI destiné à la production de jeux mathématiques de simulation à utiliser dans situations didactiques, ce qui représente notre objectif général. Cherchant à répondre à cet objectif, la méthodologie de recherche sera menée à partir de la conception d'une version numérique du jeu Mankala Colhe Três suite à la réalisation des phases présentes dans le modèle de processus de l'EDI, afin d'obtenir des informations sur les particularités de ce processus de production et, avec cela, de proposer un modèle de processus spécifique pour la production de logiciels éducatifs de ce type. Après avoir analysé les performances des premières étapes de l'Ingénierie Didactique-Informatique dans la conception du jeu, nous sommes confrontés à la nécessité de plus de détails sur la réalisation de son étape de Développement, ce qui permet en outre l'utilisation de cette méthodologie pour la production des logiciels éducatifs pour les professeurs de mathématiques intéressés par la production de jeux mathématiques numériques. Ce nouveau problème nous a conduit à un nouvel objectif de recherche, intégré à l'objectif initial, et qui fait référence à la construction d'une méthodologie pour l'étape de Développement de l'EDI qui fournit un chemin pour transformer les exigences

en code à implémenter dans le développement de jeux modèles mathématiques de simulation pour des situations didactiques. Dès lors, nous avons construit une méthodologie d'aide à l'étape de Développement de l'EDI pour la production de jeux de simulation mathématique pour des situations didactiques, basée sur les principes du Génie Logiciel axé sur la modélisation du système. Après avoir présenté et illustré l'utilisation de cette méthodologie, nous présentons en détail un modèle de processus logiciel de l'EDI qui tient compte des apports de l'analyse de la conception Mankala Colhe Três et de la méthodologie construite pour aider la phase de développement de l'EDI.

**Mots-clés:** Ingénierie Didactique-Informatique. Développement de Logiciels Éducatifs. Jeu Mathématique de Simulation. Mankala Colhe Três. Situations Didactiques.

## LISTA DE FIGURAS

Figura 1 – Encadeamento das pesquisas que conduziram à atual investigação .....	23
Figura 2 – Exemplo de cartela vencedora no Bingo dos Racionais .....	29
Figura 3 – Articulação das fases das Engenharias de Software e Didática .....	38
Figura 4 – Modelo de processo de software da Engenharia Didático-Informática .....	43
Figura 5 – Imagem de tabuleiros de Mankala Colhe Três confeccionados com sucata e materiais de baixo custo .....	60
Figura 6 – Configuração de uma jogada de Mankala Colhe Três .....	63
Figura 7 – Fotos de diferentes mankalas sendo jogados .....	65
Figura 8 – Esquema dos procedimentos metodológicos da pesquisa de Santos (2014).....	74
Figura 9 – Tipos de jogadas no Mankala Colhe Três .....	75
Figura 10 – Exemplo de configuração do tabuleiro do Mankala Colhe Três .....	77
Figura 11 – Hierarquia de jogadas no Mankala Colhe Três .....	79
Figura 12 – Etapas do Dispositivo Experimental Central da pesquisa de Santos (2014).....	80
Figura 13 – Integração do Computador no Sistema Didático .....	84
Figura 14 – Diagrama de Casos de Uso do Mankala Colhe Três Digital .....	96
Figura 15 – Ficha de regras do Mankala Colhe Três .....	99
Figura 16 – Representação de posição de jogo do Mankala Colhe Três Digital no relatório de jogadas.....	114
Figura 17 – Exemplo da atividade 1 da ficha de atividades .....	117
Figura 18 – Exemplo da atividade 2 da ficha de atividades .....	118
Figura 19 – Exemplo da atividade 3 da ficha de atividades .....	119
Figura 20 – Exemplo da atividade 4 da ficha de atividades .....	120
Figura 21 – Representação do segundo ciclo do modelo de processo da EDI .....	125
Figura 22 – Representação gráfica do relacionamento de dependência .....	135
Figura 23 – Estrutura do processo – duas dimensões.....	138
Figura 24 – Trabalhadores, atividades e artefatos .....	139
Figura 25 – Exemplo de pessoas e trabalhadores no RUP .....	141
Figura 26 – Fluxo de Análise e Design.....	146
Figura 27 – Diagrama de atividade do requisito “Realizar Jogada” .....	152
Figura 28 – Diagrama de caso de uso para o requisito “Realizar Jogada” .....	153
Figura 29 – Classes de análise do caso de uso “Realizar Jogada”, segundo o padrão ECF ...	153
Figura 30 – Diagrama de sequência para a realização de uma jogada hipotética .....	154

Figura 31 – Diagrama de Classes (de análise) para o caso de uso “Realizar Jogada” .....	155
Figura 32 – Diagrama de classes para o caso de uso “Realizar jogada” .....	155
Figura 33 – Modelo de Processo da EDI destinado a jogos matemáticos de simulação para situações didáticas .....	157
Figura 34 – Fase teórica (analítico-hipotética) .....	158
Figura 35 – Fase construtiva (desenvolvimento iterativo e incremental).....	160
Figura 36 – Fase experimental (validativa) .....	161

## LISTA DE QUADROS

Quadro 1 – Variáveis didáticas da versão digital do Bingo dos Racionais.....	34
Quadro 2 – Direcionamentos para as análises prévias da EDI .....	44
Quadro 3 – Sugestão de documentação de requisitos da EDI .....	45
Quadro 4 – Documento de requisitos educativos do Mankala Colhe Três Digital.....	90
Quadro 5 – Histórico de alterações do Modelo de Casos de Uso do Mankala Colhe Três Digital .....	94
Quadro 6 – Caso de uso consultar informações do jogo .....	98
Quadro 7 – Caso de uso iniciar jogo online.....	100
Quadro 8 – Caso de uso realizar jogada .....	102
Quadro 9 – Caso de uso efetuar <i>login</i> .....	106
Quadro 10 – Caso de uso iniciar jogo <i>online</i> .....	108
Quadro 11 – Caso de uso resolver ficha de atividades .....	109
Quadro 12 – Caso de uso configurar modelo de jogo <i>online</i> .....	110
Quadro 13 – Caso de uso arranjar alunos nas partidas online .....	112
Quadro 14 – Caso de uso realizar relatório de jogadas .....	113
Quadro 15 – Configurar ficha de atividades.....	115
Quadro 16 – Acessar respostas das atividades.....	120
Quadro 17 – Requisito não funcional: necessidade de outros sistemas .....	122
Quadro 18 – Requisito não funcional: requisitos de processo.....	123
Quadro 19 – Requisito não funcional: requisitos de produto .....	124

## SUMÁRIO

<b>1</b>	<b>INTRODUÇÃO</b> .....	17
1.1	OBJETIVOS.....	20
1.1.1	<b>Objetivo Geral</b> .....	20
1.1.2	<b>Objetivos específicos</b> .....	20
1.2	ESTRUTURA DO TEXTO.....	20
<b>2</b>	<b>A ENGENHARIA DIDÁTICO-INFORMÁTICA E A METODOLOGIA DE PESQUISA</b> .....	22
2.1	AS ENGENHARIAS DE SOFTWARE E DIDÁTICA.....	23
2.1.1	<b>Considerações sobre a Engenharia de Software</b> .....	24
2.1.2	<b>Considerações sobre a Engenharia Didática</b> .....	25
2.2	AS ORIGENS DA ENGENHARIA DIDÁTICO-INFORMÁTICA .....	27
2.3	FORMALIZAÇÃO DA ENGENHARIA DIDÁTICO-INFORMÁTICA.....	35
2.4	VERSÃO ATUAL DA ENGENHARIA DIDÁTICO-INFORMÁTICA .....	39
2.4.1	<b>Modelo de Processo de software da Engenharia Didático-Informática</b> .....	42
2.5	METODOLOGIA DA PESQUISA.....	48
<b>3</b>	<b>REALIZAÇÃO DO CICLO ANALÍTICO-HIPOTÉTICO NA CONCEPÇÃO DO MANKALA COLHE TRÊS DIGITAL</b> .....	49
3.1	ESPECIFICAÇÃO .....	49
3.2	ANÁLISES PRÉVIAS .....	55
3.2.1	<b>Os Jogos Matemáticos na Educação Matemática</b> .....	56
3.2.2	<b>O Mankala Colhe Três</b> .....	59
3.2.3	<b>A Teoria das Situações Didáticas e a análise do potencial educativo do Mankala Colhe Três</b> .....	67
3.2.3.1	Elementos da Teoria das Situações Didáticas .....	68
3.2.3.2	Análise do potencial educativo do Mankala Colhe Três .....	74
3.2.4	<b>A Transposição Didático-Informática do Mankala Colhe Três</b> .....	83
3.3	LEVANTAMENTO DE REQUISITOS .....	87
<b>4</b>	<b>REALIZAÇÃO DO CICLO HIPOTÉTICO-EXPERIMENTAL NA CONCEPÇÃO DO MANKALA COLHE TRÊS DIGITAL</b> .....	92
4.1	CONCEPÇÃO E ANÁLISE A <i>PRIORI</i> .....	92
4.1.1.	<b>Requisitos funcionais do Mankala Colhe Três Digital</b> .....	97

4.1.1.1 Consultar informações do jogo .....	98
4.1.1.2 Iniciar jogo <i>offline</i> .....	100
4.1.1.3 Realizar jogada.....	101
4.1.1.4 Efetuar <i>login</i> .....	106
4.1.1.5 Iniciar jogo <i>online</i> .....	107
4.1.1.6 Resolver ficha de atividades.....	109
4.1.1.7 Configurar modelo de jogo <i>online</i> .....	110
4.1.1.8 Arranjar alunos nas partidas <i>online</i> .....	111
4.1.1.9 Acessar relatório de jogadas.....	113
4.1.1.10 Configurar ficha de atividades .....	115
4.1.1.11 Acessar respostas das atividades .....	120
<b>4.1.2. Requisitos não funcionais do Mankala Colhe Três Digital.....</b>	<b>121</b>
4.2.1.1 Necessidade de outros sistemas .....	122
4.2.1.2 Requisitos de processo .....	122
4.2.1.3 Requisitos de produto.....	123
4.2 <b>PROTOTIPAÇÃO</b> .....	125
4.3 <b>DESENVOLVIMENTO</b> .....	127
<b>5    <b>DOS REQUISITOS À IMPLEMENTAÇÃO: CONSTRUÇÃO DE UMA METODOLOGIA DE AUXÍLIO À ETAPA DE DESENVOLVIMENTO DE JOGOS MATEMÁTICOS DE SIMULAÇÃO PARA SITUAÇÕES DIDÁTICAS</b></b>	<b>132</b>
5.1 <b>APORTES DA LINGUAGEM DE MODELAGEM UNIFICADA (UML) E DO PROCESSO UNIFICADO DA RATIONAL (RUP) NA ENGENHARIA DIDÁTICO- INFORMÁTICA</b> .....	<b>132</b>
<b>5.1.1    Considerações gerais sobre a UML</b> .....	<b>133</b>
<b>5.1.2    Considerações gerais sobre o RUP</b> .....	<b>136</b>
<b>5.1.3    Um olhar para a disciplina <i>Análise e Design</i> do RUP</b> .....	<b>142</b>
5.2 <b>METODOLOGIA PARA A ETAPA DE DESENVOLVIMENTO DA EDI NA PRODUÇÃO DE JOGOS MATEMÁTICOS DE SIMULAÇÃO PARA SITUAÇÕES DIDÁTICAS</b> .....	<b>145</b>
<b>5.2.1    Cumprimento das atividades dos detalhes de fluxo “Definir uma arquitetura candidata” e “Refinar a arquitetura”</b> .....	<b>148</b>
<b>5.2.2    Cumprimento das atividades dos detalhes de fluxo “Analisar o comportamento”, “Projetar componentes” e “Projetar o banco de dados”</b> .....	<b>150</b>

<b>5.2.3</b>	<b>Considerações sobre a Metodologia.....</b>	<b>150</b>
<b>5.3</b>	<b>UTILIZAÇÃO DA METODOLOGIA PARA A ETAPA DE DESENVOLVIMENTO DA EDI NA MODELAGEM DO MANKALA COLHE TRÊS DIGITAL .....</b>	<b>151</b>
<b>5.3.1</b>	<b>Modelagem da relação Ator-Caso de Uso por meio do Diagrama de caso de uso</b>	<b>151</b>
<b>5.3.2</b>	<b>Descoberta das Classes de análise (Entidade, Controle, Fronteira) .....</b>	<b>153</b>
<b>5.3.3</b>	<b>Produção das classes de design.....</b>	<b>155</b>
<b>6</b>	<b>MODELO DE PROCESSO DA ENGENHARIA DIDÁTICO-INFORMÁTICA DESTINADO À PRODUÇÃO DE JOGOS MATEMÁTICOS DE SIMULAÇÃO PARA SITUAÇÕES DIDÁTICAS .....</b>	<b>156</b>
<b>6.1</b>	<b>FASE TEÓRICA (ANALÍCO-HIPOTÉTICA) .....</b>	<b>158</b>
<b>6.2</b>	<b>FASE CONSTRUTIVA (DESENVOLVIMENTO).....</b>	<b>160</b>
<b>6.3</b>	<b>FASE EXPERIMENTAL (VALIDATIVA) E ETAPA DE EVOLUÇÃO .....</b>	<b>161</b>
<b>7</b>	<b>CONSIDERAÇÕES FINAIS.....</b>	<b>163</b>
	<b>REFERÊNCIAS .....</b>	<b>170</b>

## 1 INTRODUÇÃO

Muitos pesquisadores discutem sobre a problemática em torno da qualidade de softwares educativos, relacionada ao seu processo de produção, pelo fato de ainda não haver uma sistematização sólida da Engenharia de Softwares Educativos (BELLEMAIN, 2000; SANTOS, 2009; BENNETTI *et al.*, 2005).

Buscamos, por meio da presente pesquisa, contribuir com o debate acerca da produção de softwares educativos de qualidade a partir da posição epistemológica citada por Bellemain (2000), que considera necessária uma ampla reflexão sobre a especificidade da Engenharia de Softwares Educativos, devendo criar seus próprios métodos e conceitos a partir do compartilhamento e da integração dos métodos e conceitos de diversas áreas, como Educação, Didática, Psicologia Cognitiva, Ciência da Computação etc.

A presente investigação surge como uma contribuição para o que vem sendo estudado no Grupo de Pesquisa “Atelier Digitas: concepção e desenvolvimento de uma plataforma-suporte à produção de material didático com recursos ativos”, que propõe a concepção e desenvolvimento de ferramentas informáticas para a organização e gestão de situações de ensino. Dentre as investigações e desenvolvimentos do Atelier Digitas destaca-se o papel central dado aos micromundos, simulações e jogos como suporte ao ensino e a aprendizagem, como artefatos em que o aluno realiza parte da atividade matemática em ambiente computacional.

Uma das produções teórico-metodológicas estudadas atualmente no Atelier Digitas é a Engenharia Didático-Informática (EDI), uma metodologia para a produção de software educativo fundamentada nas contribuições do campo de estudos e investigações da Educação Matemática articulada aos indicativos e orientações da Engenharia de Software (TIBURCIO *et al.*, 2021).

A EDI foi concebida para dar suporte à Engenharia de Softwares Educativos, tendo em vista que essa área de conhecimento, que se propõe a estabelecer princípios teóricos e metodológicos para a concepção, desenvolvimento e validação de um software, necessita de uma articulação com conhecimentos da educação, da informática, de conteúdos disciplinares e de didática desses conteúdos na concepção de softwares educativos (BELLEMAIN, 2000; BELLEMAIN *et al.*, 2015).

Ao articular os princípios metodológicos da Engenharia de Softwares Educativos com os da Engenharia Didática (ARTIGUE, 1996), foi sistematizado (TIBURCIO, 2016) e

aprimorado (TIBURCIO, 2020) o modelo de processo de software da EDI. Atualmente, o modelo de processo da EDI é composto por quatro fases (analítica, hipotética, experimental e operacional) e quatro ciclos (1º ciclo: analítico-hipotético; 2º ciclo: hipotético-experimental; 3º ciclo: experimental-operacional; e 4º ciclo: operacional-analítico), constituídos pela integração das fases.

Nas considerações finais da pesquisa de Tiburcio (2020), o autor aponta três encaminhamentos futuros de seu trabalho, organizados nas seções: “Pôr em prática a nova EDI”; “Tipologia de software”; e “Verificar como softwares educativos são desenvolvidos”, respectivamente.

O encaminhamento “Pôr em prática a nova EDI” propõe a utilização da EDI recém aperfeiçoada, já que ela ainda não havia sido posta em funcionamento, com o intuito de verificar como ela poderá contribuir para o desenvolvimento dos produtos, além de perceber suas limitações (TIBURCIO, 2020).

O segundo encaminhamento futuro apresentado, “Tipologia de software”, se refere a possibilidade de que futuros estudos e projetos de desenvolvimento de software educativo possam utilizar a EDI na criação de outros tipos de software além dos ambientes de simulação/micromundos, tendo em vista que as pesquisas que se utilizaram da EDI anteriormente investigaram a produção de softwares apenas dessa tipologia (SILVA, 2016; SILVA, 2019; SIQUEIRA, 2019).

Ainda sobre o segundo encaminhamento, o pesquisador destaca que a utilização da EDI para a produção de diferentes tipologias de software, como jogos digitais, tutoriais em páginas *online*, aplicativos para *smartphones*, além de permitir a verificação das contribuições e limitações da EDI em tais construções trará uma diversidade de pontos de análise para o aperfeiçoamento dessa metodologia (TIBURCIO, 2020).

Por fim, sobre o encaminhamento “Verificar como softwares educativos são desenvolvidos”, Tiburcio (2020) propõe uma expansão da investigação realizada em sua pesquisa acerca das orientações metodológicas utilizadas para a produção de softwares educativos existentes, em uma perspectiva de revisão sistemática ou outra forma, para que seja apresentada uma amostra mais ampla do modo como esses produtos são desenvolvidos.

Buscando trazer novas contribuições no que se refere à problemática da qualidade de softwares educativos e, em particular, para a Engenharia Didático-Informática, propomos, na presente pesquisa, dar seguimento aos dois primeiros encaminhamentos futuros elencados por Tiburcio (2020), por meio da análise da concepção de uma versão digital do jogo matemático Mankala Colhe Três fundamentada nos princípios metodológicos da EDI.

O jogo matemático Mankala Colhe Três foi criado na UFPE, entre os anos de 2010 e 2011, por meio do projeto “Formação docente: interdisciplinaridade e ação docente – Projeto Rede”, no subprojeto que tinha como elemento central a criação de jogos matemáticos que pudessem ser construídos com material de sucata ou baixo custo (GITIRANA *et al.*, 2013).

O Mankala Colhe Três foi bastante estudado e utilizado em formações de professores no âmbito do Projeto Rede, visando o trabalho com múltiplos e divisores de um número e permitindo a mobilização noções como: desenvolver estratégias de quantificar mentalmente; resolver problemas de situações mistas (aditivas e multiplicativas); dividir por cálculo mental; mapear as possibilidades; reconhecer divisores de um determinado número; identificar múltiplos de um número; reconhecer números primos.

O jogo foi objeto de uma pesquisa de mestrado (SANTOS, 2014), que, fundamentada na Teoria das Situações Didáticas (BROUSSEAU, 1997), buscou investigar seu potencial didático. Tal pesquisa obteve resultados positivos envolvendo análises das interações dos sujeitos, do equilíbrio entre a dimensão lúdica e a intencionalidade didática nas situações de jogo e da mobilização de conhecimentos matemáticos de distintas naturezas.

Nas considerações finais de sua pesquisa, Santos (2014) apresentou algumas indicações para pesquisas futuras, com o objetivo de ampliar o potencial educativo do Mankala Colhe Três, tanto por meio de propostas de modificações das características do jogo, quanto através de sugestões metodológicas de uso do jogo em situações didáticas. Ao acreditarmos que as indicações apresentadas pelo autor possam ser atendidas por meio da proposição de uma versão digital do Mankala Colhe Três, decidimos pela opção da concepção deste jogo na nossa investigação.

Nossa escolha de utilizar a Engenharia Didático-Informática na concepção de uma versão digital do jogo Mankala Colhe Três se deu, portanto, por permitir que pudéssemos, por um lado, verificar as contribuições e limitações dessa metodologia de desenvolvimento de software educativo recém desenvolvida (TIBURCIO, 2020) e, por outro lado, analisar as particularidades inerentes da utilização da EDI no processo de produção de um jogo matemático de simulação a ser utilizado em situações didáticas, de modo a obter elementos para o aperfeiçoamento da Engenharia Didático-Informática.

Partindo do que foi apresentado acima, podemos apresentar as seguintes questões que direcionam nossa investigação: “quais as primeiras impressões da utilização dos fundamentos da EDI em sua nova versão (TIBURCIO, 2020)?”; “há sugestões de atualizações?”; “quais elementos do modelo de processo de software da EDI (TIBURCIO, 2020) precisarão ser

reconsiderados para o caso da produção de jogos matemáticos de simulação para situações didáticas?”.

Acreditamos que a concepção de uma versão digital do jogo Mankala Colhe Três, baseada nos princípios metodológicos da Engenharia Didático-Informática, fornecerá elementos suficientes para a proposição de um modelo de processo de software da EDI destinado à produção de jogos matemáticos de simulação para uso em situações didáticas.

## 1.1 OBJETIVOS

### 1.1.1 Objetivo Geral

A presente pesquisa tem o objetivo geral de propor um modelo de processo de software da Engenharia Didático-Informática destinado à produção de jogos matemáticos de simulação a serem utilizados como instrumentos centrais na vivência de situações didáticas.

Buscando sistematizar de maneira mais detalhada os resultados que pretendemos alcançar com a pesquisa, na seção seguinte apresentamos três objetivos específicos.

### 1.1.2 Objetivos específicos

- Realizar um estudo do Mankala Colhe Três e das noções que podem ser mobilizadas por meio do jogo, considerando aspectos de natureza epistemológica, cognitiva, didática e informática, de modo a estabelecer requisitos educativos para uma versão digital do jogo;
- Identificar as particularidades da utilização dos princípios da Engenharia Didático-Informática na concepção de uma versão digital do Mankala Colhe Três;
- Apresentar contribuições metodológicas para a Engenharia Didático-Informática.

## 1.2 ESTRUTURA DO TEXTO

O presente texto da Tese contém sete capítulos, organizados conforme descrição que segue.

O primeiro capítulo é a introdução do trabalho, que inclui a problemática, os objetivos (geral e específicos) de pesquisa e a visão geral da estrutura do texto da Tese.

O segundo capítulo é destinado à apresentação da Engenharia Didático-Informática (EDI) como metodologia para o desenvolvimento de software educativo e o modo como essa metodologia, por vezes, se mescla com a metodologia da pesquisa. Nesse capítulo, são trazidos aspectos da criação da EDI, de sua utilização em projetos de softwares educativos e o seu modelo de processo, que orienta e especifica as etapas de criação de software.

A realização do ciclo Analítico-hipotético da EDI na concepção de uma versão digital do jogo Mankala Colhe Três é abordada no terceiro capítulo, que compreendem as etapas de Especificação, de Análises prévias e de Levantamento de requisitos.

No Capítulo 4, apresentamos nossas considerações sobre a realização das etapas relacionadas ao ciclo Hipotético-experimental. Iniciamos pela realização da Concepção e Análise *a priori*, caracterizada pela produção do Modelo de Casos de Uso do Mankala Colhe Três Digital, e seguimos com a etapa de Prototipação. Por fim apresentamos uma discussão particular sobre a etapa de Desenvolvimento da EDI, que resultou no surgimento de um novo objetivo de pesquisa, relacionado à necessidade de construção de uma metodologia que dê suporte a essa fase do modelo de processo da EDI para o caso de jogos matemáticos de simulação para situações didáticas.

Ao buscar atender o novo objetivo da pesquisa, trazemos, no Capítulo 5, a construção de uma metodologia de desenvolvimento de jogos matemáticos de simulação para situações didáticas, focada na proposição de um percurso que auxilie a transformação de requisitos educativos em código a ser implementado.

No sexto capítulo, apresentamos o modelo de processo de software da EDI destinado à produção de jogos matemáticos de simulação para situações didáticas. Esse modelo de processo possui as contribuições advindas da análise da concepção do Mankala Colhe Três e da metodologia construída para auxiliar a fase de Desenvolvimento da EDI.

Finalizando a Tese, o Capítulo 7 traz as considerações finais da pesquisa, incluindo os encaminhamentos para pesquisas futuras e, por fim, são apresentadas as referências bibliográficas do trabalho.

## 2 A ENGENHARIA DIDÁTICO-INFORMÁTICA E A METODOLOGIA DE PESQUISA

Neste capítulo, pretendemos aproximar o leitor à construção coletiva realizada pelo grupo de pesquisa Atelier Digitas – CAC/UFPE/CNPq, especialmente na linha de pesquisa direcionada à engenharia de artefatos para o ensino, a inclusão e a sustentabilidade. É importante destacar que esse grupo de pesquisa, formado em 2019, traz contribuições advindas das discussões e investigações anteriores do grupo de pesquisa “Laboratório de Ensino da Matemática e Tecnologia (LEMATEC)”, atualmente desativado.

Iniciaremos o capítulo por meio de breves considerações acerca da Engenharia de Software e da Engenharia Didática, tendo em vista que a EDI foi formulada a partir da busca por uma articulação entre essas engenharias.

O início da abordagem sobre a elaboração da EDI está presente na seção denominada “As origens da Engenharia Didático-Informática”, que representa uma síntese da pesquisa de mestrado de Carolina Ramos (RAMOS, 2014).

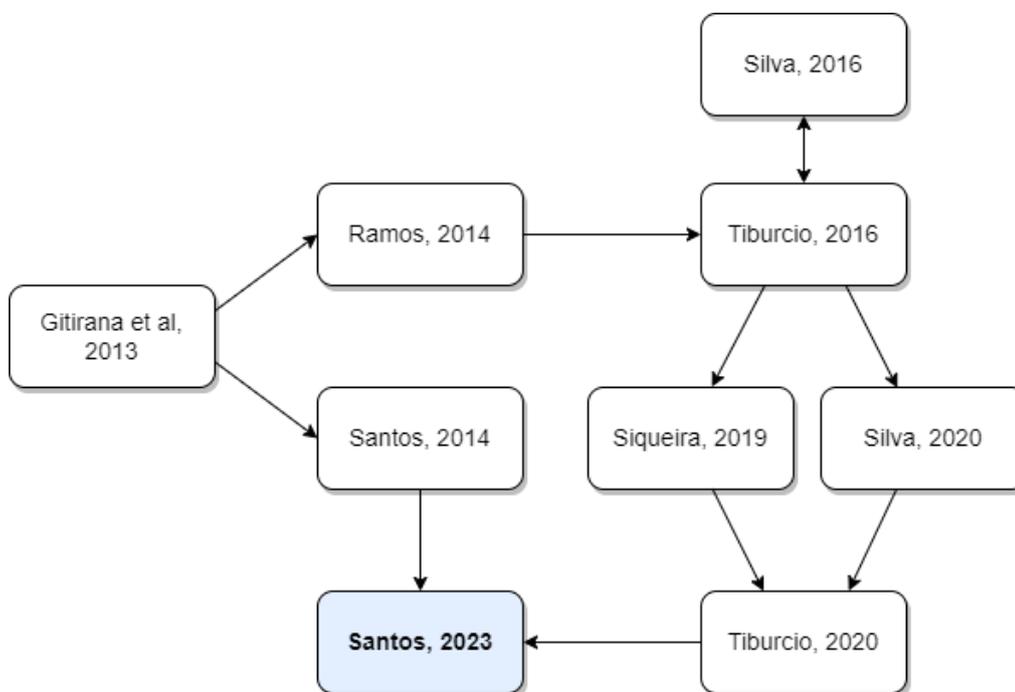
Partindo da problemática em torno da qualidade de softwares educativos e dos métodos de produção que interferem nessa questão, Ramos (2014) deu início às primeiras discussões sobre a necessidade de proposição de uma metodologia de construção de softwares educativos que se utilizasse de princípios da Engenharia Didática e da Engenharia de Softwares simultaneamente. Nessa pesquisa a nomenclatura da EDI foi apresentada, mas ainda como “Engenharia Didática Informática” (RAMOS, 2014).

Na seção seguinte, intitulada como “Formalização da Engenharia Didático-Informática”, realizaremos uma síntese da pesquisa de Tiburcio (2016), que, em colaboração com a pesquisa de Silva (2016), formalizou mais detalhadamente a EDI, por meio de uma metodologia composta pela concepção, criação e análise da Engenharia Didático-Informática.

Para finalizar o resgate da elaboração da EDI, traremos a seção “Versão atual da EDI”, em que serão apresentadas as contribuições que conduziram Tiburcio (2020) à elaboração da Engenharia Didático-Informática como conhecemos hoje, contendo o seu atual modelo de processo de software educativo. Vale ressaltar que, na metodologia de sua pesquisa, o autor buscou tanto investigar as engenharias aplicadas ao desenvolvimento de softwares educativos que possuíam pesquisas consolidadas, quanto realizar um estudo dos trabalhos que utilizaram a EDI, em sua versão de Tiburcio (2016), como fundamentação teórico-metodológica para a proposição de softwares educativos (SILVA, 2016; SIQUEIRA, 2019; SILVA, 2019).

Com a intenção de auxiliar a leitura das próximas seções e de enfatizar a característica colaborativa e incremental das pesquisas relacionadas à utilização, elaboração e aprimoramento da Engenharia Didático-Informática, apresentamos, a seguir, um diagrama que ilustra o encadeamento das pesquisas recentes que conduziram à atual investigação.

Figura 1 – Encadeamento das pesquisas que conduziram à atual investigação



Fonte: elaborada pelo autor.

Optamos por apresentar as sínteses das principais pesquisas recentes que ajudaram a elaborar formalmente a Engenharia Didático-Informática com o intuito de que o leitor tenha a oportunidade de conhecer os principais aportes à EDI advindos de cada uma das pesquisas no decorrer da leitura. Deste modo, como propomos dar seguimento a esse encadeamento de investigações, assumimos como parte da nossa fundamentação os aportes apresentados no presente capítulo.

Ainda neste capítulo, após as considerações acerca da Engenharia Didático-Informática, explicitaremos a metodologia de nossa pesquisa, em que será enfatizada a forma como, por vezes, ela se mescla com a metodologia de produção de software educativo proposto pela EDI.

## 2.1 AS ENGENHARIAS DE SOFTWARE E DIDÁTICA

### 2.1.1 Considerações sobre a Engenharia de Software

A Engenharia de Software, conceituada por Sommerville como “uma disciplina da engenharia que se preocupa com todos os aspectos de produção de software” (SOMMERVILLE, 2011, p. 4), possui como principais atividades a especificação de software, o desenvolvimento de software, a validação de software e a evolução de software. Deste modo, o autor endossa que o foco da Engenharia de software “está em todos os aspectos da produção de software, desde os estágios iniciais da especificação do sistema até sua manutenção, quando o sistema já está sendo usado” (SOMMERVILLE, 2011, p. 5).

Discorrendo um pouco mais sobre as características da Engenharia de Software, Sommerville fala das particularidades que deve haver nas utilizações de notações, métodos e técnicas quando se trata de diferentes tipos de software, pois, como afirma o autor,

Não faz sentido procurar notações, métodos ou técnicas universais para a engenharia de software, porque diferentes tipos de software exigem abordagens diferentes. Desenvolver um sistema de informações corporativo é totalmente diferente de desenvolver um controlador para um instrumento científico. Nenhum desses sistemas tem muito em comum com um jogo computacional com gráficos intensos. Todas essas aplicações precisam de engenharia de software, embora não necessitem das mesmas técnicas (SOMMERVILLE, 2011, p. 2).

Da mesma forma que não faz sentido universalizar as notações, os métodos e as técnicas na Engenharia de Software para os diferentes tipos de softwares, já que cada software pode ter características bastante diversas, também não se pode dizer que um método é melhor do que outro, já que, como afirma Sommerville, “enquanto todos os projetos de software devem ser gerenciados e desenvolvidos profissionalmente, técnicas diferentes são adequadas para tipos de sistemas diferentes” (SOMMERVILLE, 2011, p. 4).

Uma definição importante no campo da Engenharia de Software, que se remete à abordagem sistemática usada, é a de *processo de software*. Segundo Sommerville, processo de software “é uma sequência de atividades que leva à produção de um produto de software” (SOMMERVILLE, 2011, p.5). Ainda segundo o autor, todos os processos de software possuem quatro atividades fundamentais:

1. Especificação de software, em que clientes e engenheiros definem o software a ser produzido e as restrições de sua operação;
2. Desenvolvimento de software, em que o software é projetado e programado;
3. Validação de software, em que o software é verificado para garantir que é o que o cliente quer;
4. Evolução de software, em que o software é modificado para refletir a mudança de requisitos do cliente e do mercado (SOMMERVILLE, 2011, p. 5-6).

Na mesma linha do que é trazido por Sommerville (2011), Magela identifica a Engenharia de Software como um

Conjunto de técnicas, métodos, ferramentas e processos utilizados na especificação, construção, implementação e manutenção de um software que visa a garantir a gerência, o controle e a qualidade dos artefatos gerados através de recursos humanos (MAGELA, 2006, p. 24).

Dentre os aspectos trazidos por Magela (2006), presentes na citação acima, um deles merece destaque em nossa pesquisa, que é a problemática da qualidade de softwares. Ao longo do presente capítulo, discutiremos a respeito desse aspecto, tendo em vista que foi a problemática da qualidade de softwares, no ambiente educativo, que motivou pesquisadores a formalizarem e aperfeiçoarem a Engenharia Didático-Informática, hoje caracterizada como uma metodologia para o desenvolvimento de software educativo, com um modelo de processo que orienta e especifica as etapas de criação de software (TIBURCIO, 2020).

### **2.1.2 Considerações sobre a Engenharia Didática**

Os estudos da Engenharia Didática tiveram início por volta das décadas de 1970 e 1980 na França, com buscas por uma metodologia que fosse capaz de auxiliar a elaboração nas sequências didáticas voltadas para o ensino de matemática. Em 1990 Michèlle Artigue sistematizou os resultados de diversas investigações realizadas em um artigo, trazendo um enfoque à Engenharia Didática como uma metodologia de pesquisa, apresentando suas características, singularidades e fases (BITTAR, 2017).

A Engenharia Didática como metodologia de pesquisa busca justamente estudar condições que possam favorecer a aprendizagem de um tema ou conteúdo no sistema didático. Para isso, utiliza um esquema experimental que engloba a concepção, a realização, a observação e a análise das sequências de ensino. A utilização do termo “engenharia” vem do fato de que esse trabalho didático se assemelha ao trabalho de um engenheiro, que,

para realizar um projeto preciso, se apoia nos conhecimentos científicos do seu domínio, aceita submeter-se a um controle de tipo científico, mas ao mesmo tempo, se encontra obrigado a trabalhar sobre objetos muito mais complexos do que os objetos depurados da ciência, e, portanto, a estudar de uma forma prática, com todos os meios ao seu alcance, problemas de que a ciência não quer ou ainda não é capaz de se encarregar (ARTIGUE, 1996, p.193, tradução nossa).

A Engenharia Didática é constituída de quatro fases: as análises prévias; a concepção e a análise *a priori* das situações didáticas da engenharia; a experimentação; e, por fim, a análise *a posteriori* e a validação (ARTIGUE, 1996). Há uma particularidade especial na Engenharia Didática, que é o modo como a validação se dá, estritamente interna, como pode ser observado na citação abaixo, em que Almouloud e Coutinho falam que a Engenharia Didática, vista como metodologia de pesquisa,

Caracteriza-se também como pesquisa experimental pelo registro em que se situa e pelos modos de validação que lhe são associados: a comparação entre análise *a priori* e análise *a posteriori*. Tal tipo de validação é uma das singularidades dessa metodologia, por ser feita internamente, sem a necessidade de aplicação de um pré-teste ou de um pós-teste (ALMOULOU E COUTINHO, 2008, p. 66).

Na Engenharia Didática, a fase de concepção precisa se apoiar tanto em um quadro teórico didático geral e em conhecimentos didáticos já adquiridos no domínio estudado, quanto em análises preliminares que são, em sua maior parte:

- A análise epistemológica dos conteúdos visados pelo ensino;
- A análise do ensino habitual e dos seus efeitos;
- A análise das concepções dos alunos, das dificuldades e obstáculos que marcam a sua evolução;
- A análise do campo de restrições no qual virá a situar-se a realização didática efetiva;
- E, naturalmente, tendo em conta os objetivos específicos da investigação. (ARTIGUE, 1996, p. 198).

Ao considerar os pontos apresentados, Artigue (1996) organiza a fase de análises prévias em três dimensões: a dimensão *epistemológica*, associada às características do saber em jogo; a dimensão *cognitiva*, associada às características cognitivas do público ao qual se dirige o ensino; e a dimensão *didática*, associada às características do funcionamento do sistema de ensino. Um importante aspecto trazido por Artigue sobre essas diferentes componentes de análise é que,

Nos trabalhos publicados, a maioria das vezes, nem todas as diferentes componentes de análise mencionadas atrás intervêm de forma explícita. Poderá ser um excelente exercício de didática identificar, a propósito de trabalhos precisos, as dimensões privilegiadas e tentar procurar, *a posteriori*, o seu significado didático (ARTIGUE, 1996, p. 199).

A fase de concepção e análise *a priori* é caracterizada pela atuação do investigador sobre um determinado número de variáveis de comando, que ele propõe serem variáveis pertinentes

para o problema estudado (ARTIGUE, 1996). Tais variáveis podem ser diferenciadas em dois tipos:

- as variáveis *macro-didáticas ou globais*, que dizem respeito à organização global da engenharia;
- e as *variáveis micro-didáticas ou locais*, que dizem respeito à organização local da engenharia, isto é, à organização de uma sessão ou de uma fase, podendo umas e outras ser, por sua vez, variáveis de ordem geral ou variáveis dependentes do conteúdo didático cujo ensino é visado (ARTIGUE, 1996, p. 202-203).

Na análise *a priori*, busca-se determinar de que forma as escolhas efetuadas permitem controlar os comportamentos dos alunos e o sentido desses comportamentos (ARTIGUE, 1996).

Na fase de experimentação da Engenharia Didática, é posto em prática o dispositivo preparado, de modo que, após sua realização, possa ser realizada a fase de análise *a posteriori*, subsidiada pelo conjunto de dados recolhidos a partir da experimentação, como observações realizadas nas sessões de ensino, produções dos alunos em sala de aula ou fora dela, além de possíveis instrumentos complementares, oriundos de metodologias externas, como questionários e testes (ARTIGUE, 1996).

Como dito anteriormente, é por meio do confronto entre as análises *a priori* e *a posteriori* que a validação das hipóteses levantadas na investigação ocorre.

## 2.2 AS ORIGENS DA ENGENHARIA DIDÁTICO-INFORMÁTICA

A Engenharia Didático-Informática surge como uma proposta teórico-metodológica às questões em torno da qualidade de softwares educativos para o ensino e a aprendizagem da Matemática e dos métodos de desenvolvimento desses softwares, que busquem considerar aspectos teóricos e práticos sobre o desenvolvimento de recursos com essa finalidade (RAMOS, 2014).

Questões sobre o desenvolvimento de softwares educativos estão presentes desde a produção dos primeiros softwares destinados ao ensino até os dias de hoje. Em sua pesquisa, Ramos (2014) historia um pouco desse percurso, iniciando pelas primeiras produções de softwares educativos, em torno dos anos 1950, no ensino programado. Ainda sobre as raízes do uso do computador para o ensino, Rézeau afirma que

Todos os autores do domínio concordam em dizer que o ensino pelo computador encontra suas raízes no ensino programado dos anos 1950-1970. O nível de adequação entre as teorias behavioristas de Skinner e as possibilidades técnicas do computador fazia dele a "máquina de ensinar" ideal. (RÉZEAU, 2001, p. 243 - tradução nossa).

Há, segundo Ramos (2014), uma distinção, entre as primeiras produções de softwares educativos e a origem da Engenharia de Software Educativo (ESE) como domínio científico, presente em comunicações científicas, que emerge no início dos anos 1990 com as publicações de Galvis (1992).

Sobre a emergência da ESE, Ramos (2014) fornece um resgate histórico que se inicia com as reflexões iniciais da informática educativa, como aparecimento e introdução dos primeiros microcomputadores nas escolas, que provocou uma importante mudança de paradigma e levantou novas questões.

Segundo a autora, foi nesse contexto em que se conceberam vários softwares considerados de qualidade destinados ao ensino de matemática. Sobre a qualidade desses softwares educativos, a autora afirma que

Os *software* citados: *LOGO*, *Derive*, *Cabri-géomètre* podem ser considerados como experiências bem sucedidas da informática educativa, pela quantidade de usuários, quantidade de publicações em torno da concepção ou do uso, existência de eventos nacionais ou internacionais sobre esses *software*, longevidade, quantidade de *software* desenvolvidos retomando as ideias inovadoras criadas (RAMOS, 2014, p. 27).

Ainda com foco na argumentação sobre a qualidade dos softwares educativos, Ramos (2014) traz um elemento crucial sobre o processo de produção de tais recursos, que foi a busca por respostas originais e apropriadas por meio da integração entre conhecimentos relativos aos conteúdos específicos, à educação e à informática. Detalhando um pouco mais sobre a integração de áreas no desenvolvimento de projetos de softwares educativos bem-sucedidos, a autora enfatiza que

É importante destacar que esses *software* bem sucedidos foram geralmente inventados por pesquisadores-desenvolvedores que tinham no mínimo uma dupla competência de engenharia de *software* e de um outro domínio, como, por exemplo, a matemática (David Stoutmeyer para *Derive*), a psicologia (Seymour Papert para *LOGO*), a didática da matemática (Baulac, Bellemain e Laborde para *Cabri-géomètre*). Nessa leitura da concepção de um software educativo, entendemos que ela não é resultado de uma simples justaposição de competências, mas de uma integração das mesmas (RAMOS, 2014, p. 27-28).

Considerando a problemática da Engenharia de Software Educativo, Ramos (2014) desenvolveu uma pesquisa de mestrado cujo objetivo foi de integrar os princípios teórico-metodológicos da Engenharia Didática (ARTIGUE, 1996) à Engenharia de Softwares

Educativos no contexto da concepção e do desenvolvimento de uma versão digital do jogo matemático Bingo dos Racionais (RAMOS, 2014; BELLEMAIN *et al.*, 2015).

O Bingo dos Racionais foi criado no âmbito do Projeto Formação Docente: interdisciplinaridade e ação docente – Projeto Rede (GITIRANA *et al.*, 2013), assim como o Mankala Colhe Três, objeto de estudo da presente pesquisa. Ambos os jogos foram desenvolvidos com a proposta de permitirem que os próprios alunos e professores pudessem construí-los com material de sucata ou baixo custo.

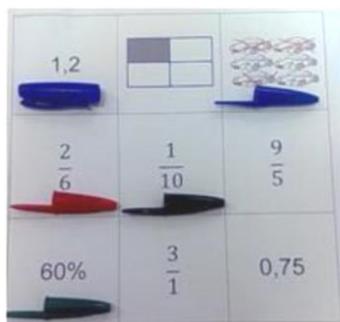
Na pesquisa de Ramos (2014), a autora enfatiza que o jogo foi concebido com o objetivo de trabalhar com as diferentes representações dos números racionais (fração, decimal, porcentagem, figural, língua natural etc.) ao indicar uma situação em que a noção de jogo é explorada como viés para criar um ambiente *ad hoc* (RAMOS, 2014).

Como os tradicionais jogos de bingo, o Bingo dos Racionais possui cartelas para serem marcadas pelos jogadores a partir de valores anunciados pelo “chamador”. No caso do Bingo dos Racionais, as cartelas são compostas por nove quadrados, organizados em uma configuração quadrada “3 por 3”, e os elementos a serem marcados são representações de números racionais.

O “chamador” do jogo (que pode ser o professor ou um estudante) anuncia o número selecionado na língua natural e os jogadores, que jogam em grupos de, no máximo, três estudantes, marcam na cartela uma possível representação desse número, caso a cartela a contenha. Vence o jogo aquele jogador que conseguir marcar corretamente as representações que estejam em uma mesma linha, coluna ou diagonal de sua cartela.

O jogo possui possibilidades de variações para torná-lo mais complexo, por meio da manipulação simultânea de diferentes representações de um mesmo número (MELO *et al.*, 2013). A seguir, será apresentada, como exemplo, uma cartela vitoriosa do jogo, em que se utilizou tampinhas de caneta para marcar as possíveis representações anunciadas:

Figura 2 – Exemplo de cartela vencedora no Bingo dos Racionais



Fonte: Melo *et al.* (2013).

A partir do objetivo principal de exploração das diferentes representações de números racionais, os criadores do Bingo dos Racionais apresentaram algumas finalidades educacionais do jogo. Segundo esses pesquisadores, espera-se que o jogo favoreça nos alunos o desenvolvimento das capacidades de:

- Produzir escritas numéricas apropriadas para números racionais, a partir de expressões correspondentes em linguagem natural;
- Reconhecer que um número racional pode ser representado de diversas maneiras;
- Associar vários representantes (expressões em linguagem natural, representações figurativas de quantidades contínuas e discretas e representações simbólicas numéricas em forma de fração, número decimal e porcentagem) a um mesmo representado (número racional);
- Articular diferentes representações de um mesmo número (MELO *et al.*, 2013, p. 127).

Uma primeira ideia para o início da articulação entre os aportes da Engenharia Didática e da Engenharia de Softwares na produção de uma versão digital do Bingo dos Racionais veio do fato de que a criação do jogo considerou princípios teórico-metodológicos da Engenharia Didática, em que, particularmente, foi efetuada uma análise *a priori* detalhada das situações de jogo. Deste modo, o estudo de Ramos (2014) focou em “retomar essa análise para investigar os aportes da engenharia didática à engenharia de software educativo para a concepção de situações didáticas no contexto computacional” (RAMOS, 2014, p. 21).

Essa aproximação entre a Engenharia Didática e a Engenharia de Softwares Educativos também é discutida por Tchounikine, em que o autor afirma que a Engenharia Didática pode fornecer um quadro metodológico para a criação de softwares educativos, como pode ser visto na citação a seguir:

No trabalho focado no processo didático e na epistemologia, o ponto de entrada é a noção de *conhecimento em jogo*<sup>1</sup> e a modelação da situação que permite sua aquisição, com a teoria das situações didáticas (BROUSSEAU, 1998) fornecendo um quadro geral de análise e os trabalhos de engenharia didática um quadro geral para as dimensões ligadas à concepção (TCHOUNIKINE, 2004, p. 6, tradução nossa. Grifos nossos).

Sobre as características disciplinares da equipe de desenvolvimento de softwares educativos, Tchounikine (2009) afirma que a Engenharia de Softwares Educativos é de natureza transdisciplinar, considerando uma distinção que ele próprio realiza, entre pluridisciplinaridade, interdisciplinaridade e transdisciplinaridade:

---

<sup>1</sup> Proposta de tradução da expressão francesa “*enjeu de connaissance*”.

- Pluridisciplinaridade: enriquecimento de uma reflexão disciplinar pela contribuição de outras disciplinas (o problema é abordado pelo suporte de várias disciplinas, mas num esquema de justaposição).
- Interdisciplinaridade: transferência e adaptação de métodos de uma disciplina a outra (as respectivas contribuições e os cruzamentos podendo levar as disciplinas a evoluir).
- Transdisciplinaridade: integração de diferentes abordagens científicas num quadro próprio, ultrapassando os quadros disciplinares (elaboração de conceitos ou métodos próprios etc.) (TCHOUNIKINE, 2009, p.16, tradução nossa).

Em sua pesquisa, Ramos (2014) apresenta uma importante característica da Engenharia Didático-Informática, que se mantém até sua versão atual, que é o olhar transdisciplinar dado à Transposição Didática (CHEVALLARD, 1991), resultando na Transposição Informática<sup>2</sup>, conceito introduzido por Balacheff (a) e considerado como fundamental na EDI.

Para melhor explicar o parágrafo anterior, apresentaremos, a seguir, uma citação de Chevallard em que é trazido o conceito de Transposição Didática:

Um conteúdo de saber que foi designado como saber a ensinar, sofre, a partir de então, um conjunto de transformações adaptativas que vão torná-lo apto para ocupar um lugar entre os *objetos de ensino*. O “trabalho” que transforma um objeto de saber a ensinar em objeto de ensino é denominado *transposição didática* (CHEVALLARD, 1991, p. 45, tradução nossa. Grifos do autor).

Sobre a Transposição Informática, Bellemain (2000) afirma que, da mesma forma em que a transposição didática traz o olhar para os fenômenos de transformação do saber de referência em saber a ensinar, na engenharia de softwares educativos, ao ser introduzida a dimensão informática nesse processo, o olhar informático precisa ser considerado, não apenas focando nos aspectos da integração das novas tecnologias no ensino, aspecto que já vem sendo estudado, cujo foco é o reflexo sobre o saber a ensinar, mas, principalmente, do ponto de vista das transformações do saber de referência ocasionado pela presença do computador desde o início do processo de produção dessas tecnologias. Para Balacheff,

O que normalmente é chamado de informatização não é uma simples transliteração; ambientes de aprendizagem informatizados são o resultado de uma construção que é o local de novas transformações de objetos de ensino. Chamamos, portanto, esse processo de *transposição informática* (BALACHEFF, 1994, p. 1, tradução nossa).

Ao considerar as questões relacionadas à transposição informática, Ramos (2014) propôs um processo de produção de software educativo baseado em uma reformulação,

---

<sup>2</sup> Também chamada de “Transposição Didática Informática” ou “Transposição Didático-Informática” ao longo do texto, devido ao fato de estar sempre atrelada ao conhecimento que se pretende ser ensinado no final do processo.

incorporando a dimensão informática na Engenharia Didática realizada para a criação do jogo físico, sugerindo, deste modo, uma Engenharia Didática Informática<sup>3</sup>.

Ramos (2014) escolheu incorporar a discussão da dimensão informática não como uma quarta dimensão das análises prévias da Engenharia Didática, mas em cada uma das outras três dimensões. Essa escolha feita pela autora se mostrou pertinente em seu trabalho, principalmente pelo fato de que a construção do Bingo dos Racionais já possuía uma discussão razoavelmente madura em cada uma das três dimensões da Engenharia Didática, fazendo com que a dimensão informática trouxesse um novo olhar para a proposta do jogo digital, complementando os aspectos epistemológicos, cognitivos e didáticos a partir de um viés informático.

Na análise da dimensão epistemológica, Ramos (2014) insere a dimensão informática por meio da discussão sobre a limitação e as contribuições computacionais existentes na maneira de representar os números racionais. Segundo a autora, o número, nesse contexto, é sempre discreto e finito, mas, por outro lado, é dinâmico (RAMOS, 2014).

De fato, ao analisar as características dos números racionais digitalmente, concordamos com Bellemain *et al.*, quando afirmam que

a representação digital dos racionais elaborada deve ser satisfatória em relação com as situações abordadas no contexto do bingo dos racionais, e pode ter aportes por trazer novos registros (digitais e dinâmicos) de representação dos racionais (BELLEMAIN *et al.*, 2015, p. 7).

Na análise da dimensão cognitiva, devido a existência de interfaces entre o aluno e o jogo, a dimensão informática discutiu novas questões:

- como da inserção de respostas pelos alunos no jogo, considerando inclusive a possibilidade de inserção de respostas erradas (que são manifestações de conhecimentos do aluno).
- como é feito o tratamento das respostas do aluno como e quando são dados os *feedbacks*. Se trata, em particular, de determinar a natureza desses *feedbacks* (sem *feedback*, do tipo certo-errado, pontuação, *feedback* calculados e evoluindo em função do histórico do aluno, explicação, proposta de outra atividade parecida) (BELLEMAIN *et al.*, 2015, p. 7).

Em relação à análise da dimensão informática na dimensão didática das análises prévias, também são trazidas novas questões para o jogo, relacionadas à determinação do papel do computador no gerenciamento das fases do jogo, das regras, da avaliação etc.:

---

<sup>3</sup> Em pesquisas posteriores a de Ramos (2014), a EDI teve sua nomenclatura atualizada para *Engenharia Didático-Informática*, o que justifica a utilização dessa versão atualizada no presente trabalho.

Quais são as mudanças realizadas nos cenários no contexto computacional? Construir novas situações, novos cenários? Novos papéis do aluno, do professor? O bingo sendo um jogo coletivo, como é considerada essa dimensão coletiva no contexto computacional? Quais são os suportes à orquestração instrumental do jogo para sua integração no ensino? (BELLEMAIN *et al.*, 2015, p. 8).

Retomando os elementos das análises prévias, deu-se início à análise *a priori*, de modo a elaborar situações e estabelecer as variáveis didáticas que subsidiarão a elaboração de sequências ou cenários para o favorecimento de aprendizagens relacionadas aos números racionais.

Essa análise foi realizada na elaboração do jogo físico, que concebeu: o bingo; as variáveis; os números racionais utilizados etc. Nessa análise, puderam ser distinguidos dois tipos de variáveis: “as que dizem respeito às escolhas de números e representações nas cartelas e fichas e as que caracterizam as regras do jogo (papel dos alunos, do professor, conferência do jogo, natureza e gerenciamento dos *feedbacks* etc.)” (BELLEMAIN *et al.*, 2015, p. 8).

Em relação ao jogo digital, as variáveis têm a função de elementos de configuração do jogo e permitem ampliar o repertório de valores dessas variáveis. No entanto, como destacam Bellemain *et al.*, “se trata de rever as variáveis (acrescentando novas variáveis pertinentes no contexto computacional, eliminando outras não pertinentes nesse mesmo contexto) e de rever os valores possíveis (acrescentando/eliminando)” (BELLEMAIN *et al.*, 2015, p. 8).

A Engenharia Didático-Informática pôde ser operacionalizada por meio da abordagem da concepção da versão digital do Bingo dos Racionais. O trabalho de Ramos (2014) não chegou a fornecer um protótipo do jogo digital, devido ao limite de tempo de realização de uma pesquisa de mestrado, no entanto, a autora apresentou diversas variáveis para a fase de concepção e análise *a priori*, considerando os aspectos obtidos a partir das análises prévias. Na ilustração a seguir serão apresentadas algumas das variáveis levantadas por Ramos (2014) e sistematizadas por Bellemain *et al.* (2015):

Quadro 1 – Variáveis didáticas da versão digital do Bingo dos Racionais

Fichas
<ul style="list-style-type: none"> <li>• <b>Chamada</b> - ela pode ser feita pelo próprio computador, com função de voz para uma chamada equivalente à versão papel, mas se pode utilizar outras mídias.</li> <li>• <b>Fichas</b> - ela pode ser feita pelo próprio computador, com função de voz para uma chamada equivalente à versão papel, mas se pode utilizar outras mídias.</li> <li>• <b>Representação das Fichas</b> - as fichas, dependendo da chamada, podem ter outras representações além da língua natural.</li> </ul>
Cartelas
<ul style="list-style-type: none"> <li>• <b>Tamanho das cartelas</b> - com o bingo digital, as cartelas podem ter um número qualquer de linha e colunas.</li> <li>• <b>Registro das chamadas</b> - esse registro pode ser feito pelo próprio computador.</li> <li>• <b>Dinamismo</b> - possibilidade de apresentar o jogo de maneira mais dinâmica como, por exemplo, numa representação figurativa e dinâmica de números racionais.</li> </ul>
Jogadores
<ul style="list-style-type: none"> <li>• <b>Número de jogadores por cartela</b> - esse número fica a critério da proposta didática escolhida pelo professor.</li> <li>• <b>Papel do aluno</b> - com a digitalização do Bingo, ele pode continuar com o papel de chamador no jogo, marcador das peças, autor de fichas e cartelas.</li> <li>• <b>Papel do professor</b> - o professor continua desempenhando a função mediadora no jogo, tendo a possibilidade de ser autor de fichas e cartelas.</li> <li>• <b>Elaborador de cartelas</b> - o usuário (o professor ou o aluno), tem a possibilidade de criar um jogo, montar a cartela com diferentes representações.</li> </ul>
Jogo
<ul style="list-style-type: none"> <li>• <b>Fim do jogo</b> - manter a lógica do bingo, para quem complete uma linha, coluna.</li> <li>• <b>Conferência do jogo</b> - o computador pode fazer automaticamente a cada resposta, no final do jogo.</li> <li>• <b>Tempo</b> - a inclusão de algumas variáveis envolvendo o tempo como, por exemplo, a delimitação do tempo para uma partida, ou para a marcação da cartela. Ou ainda, em um jogo colaborativo, (onde pode haver um jogo em rede, com vários usuários online) validar apenas o jogador que marcar a cartela primeiro.</li> <li>• <b>Nível do jogo</b> - o aumento do número de fichas e cartelas facilita a inclusão de novos níveis e desafios.</li> </ul>
Conteúdo
<ul style="list-style-type: none"> <li>• <b>Conteúdo</b> - expansão para outros assuntos matemáticos, como por exemplo, assuntos de geometria, números irracionais etc.</li> <li>• <b>Representação dos números</b> - pode haver um aumento do número das diversas representações apresentadas no jogo. O computador vai possibilitar uma maior variedade dessas representações, incluindo representações dinâmicas.</li> </ul>

Fonte: adaptado de Bellemain *et al.* (2015).

A pesquisa de Ramos (2014) trouxe uma grande contribuição para o processo de Engenharia de Softwares Educativos ao integrar os princípios da Engenharia Didática à Engenharia de Software Educativo e deu início a uma série de novas formalizações da EDI, que serão apresentadas nas seções seguintes do presente capítulo.

### 2.3 FORMALIZAÇÃO DA ENGENHARIA DIDÁTICO-INFORMÁTICA

Motivado pela problemática da concepção de recursos tecnológicos digitais de qualidade para auxiliar o ensino e a aprendizagem e inspirado na primeira tentativa de proposta de uma Engenharia Didática Informática, realizada por Ramos (2014), Tiburcio deu início a uma pesquisa de mestrado, que visava a criação de um processo de desenvolvimento de software educativo do tipo micromundo para a aprendizagem da Matemática que levasse em consideração uma perspectiva transdisciplinar de exploração das potencialidades teóricas e tecnológicas das engenharias de Software e Didática, como pode ser visto em seu objetivo geral:

Construir, analisar e validar um Processo de Desenvolvimento de micromundo para a aprendizagem da Matemática, integrando os métodos da engenharia de requisitos e Ágeis, como elementos da Engenharia de Softwares, integrados com princípios da Engenharia Didática numa perspectiva transdisciplinar de exploração das potencialidades teóricas e tecnológicas dessas engenharias (TIBURCIO, 2016, p. 17).

Nessa pesquisa, Tiburcio iniciou a primeira formalização mais detalhada da Engenharia Didático-Informática, por meio de um trabalho que buscava conceber esse processo de desenvolvimento de micromundo levando em conta uma articulação, percebida como primordial pelo autor, entre diversas áreas de conhecimento envolvidas na concepção desses softwares – Cognição, *Design*, Epistemologia etc. – além de uma conexão entre a Engenharia de Software e as contribuições teóricas sobre o ensino e aprendizagem (TIBURCIO, 2016).

O autor retoma a discussão iniciada por Ramos (2014) enfatizando que a necessidade de articular a Engenharia de Software e a Engenharia Didática vem do fato de que ambas apresentam limitações para a produção de software educativo se consideradas separadamente. De fato, como afirma Tiburcio,

As limitações da Engenharia Didática foram verificadas no momento em que tal metodologia não contempla, em suas contribuições, a totalidade das exigências para a concepção de SE<sup>4</sup>. A Engenharia de Software, por sua vez, não reúne especificidades que os software educativos necessitam, quando se observa que os modelos padronizados de desenvolvimento foram criados para produtos comerciais, bancários, domésticos, etc. (TIBURCIO, 2016, p. 26).

Ao fazer um estudo sobre a forma com que os softwares educativos são geralmente desenvolvidos, Tiburcio (2016) apresenta um quadro geral composto por três categorias: as

---

<sup>4</sup> Abreviação de “Software Educativo”.

*metodologias padronizadas* provenientes da Engenharia de Softwares (não educativos); a *adaptação de metodologias padronizadas*; e a *integração de áreas*.

Explanando um pouco sobre cada um desses tipos de desenvolvimento, o autor enfatiza que, quando os interessados no projeto se utilizam de *metodologias padronizadas* para conceber recursos, eles não realizam qualquer tipo de modificação na metodologia escolhida. Dentre as metodologias padronizadas de desenvolvimento de softwares, incluem-se: o método de Cascata, o Desenvolvimento Iterativo e Incremental, o método de Prototipagem, o método de Espiral, as Metodologias Ágeis, entre outras (TIBURCIO, 2016).

Segundo Tiburcio (2016), em relação às *adaptações de metodologias padronizadas*, os interessados concluem que, ao escolherem uma metodologia para o desenvolvimento do software educativo desejado, são necessárias adaptações do método, por reconhecerem as especificidades dos produtos tecnológicos para fins educativos em comparação com softwares para fins diversos.

Por fim, na *integração de áreas*, o autor traz que as contribuições teóricas sobre o ensino e a aprendizagem, que envolvem o que se discute na academia, nas orientações de documentos oficiais etc., são observadas e busca-se conectar os elementos oriundos dessa análise às possibilidades tecnológicas atuais (TIBURCIO, 2016).

Em suas considerações, Ramos (2014) afirma que a proposta de EDI apresentada no seu trabalho permite evitar que os softwares educativos sejam construídos sem o olhar das contribuições teóricas da pesquisa em Educação Matemática e, por outro lado, consideram as contribuições dos estudos da Engenharia de Software.

Mesmo reconhecendo toda a contribuição da investigação de Ramos (2014) sobre a EDI para a concepção de softwares educativos, Tiburcio (2016) percebeu que a proposta trazida pela autora continha algumas limitações, como fases não explícitas e outras características que dificultavam a utilização da EDI na criação de outros softwares educativos, como pode ser observado no trecho a seguir:

A intenção de relacionar a ED com a ESE não apresentou os elementos comuns e também os complementares em ambas as engenharias, além disso, o desenvolvimento do software foi claramente pautado na Engenharia Didática, onde não ficou compreensível quais são as fases do esquema proposto, bem como os processos (etapas) de criação do Bingo dos Racionais nos resultados da pesquisa. Isso que nos motivou a dar continuidade ao que fora sugerido por Ramos (2014) (TIBURCIO, 2016, p. 30).

Como apresentado anteriormente, a pesquisa de Tiburcio (2016) visou conceber, criar e analisar a Engenharia Didático-Informática. Para isso, o autor decidiu realizar um percurso

metodológico composto inicialmente por uma revisão de literatura, contendo um panorama geral a respeito do desenvolvimento de software educativo, seguido da concepção da Engenharia Didático-Informática e do desenvolvimento do processo de software e, por fim, pela realização de um estudo de caso que se utilizou do processo de software ainda em desenvolvimento, para que a EDI pudesse ser analisada em uso.

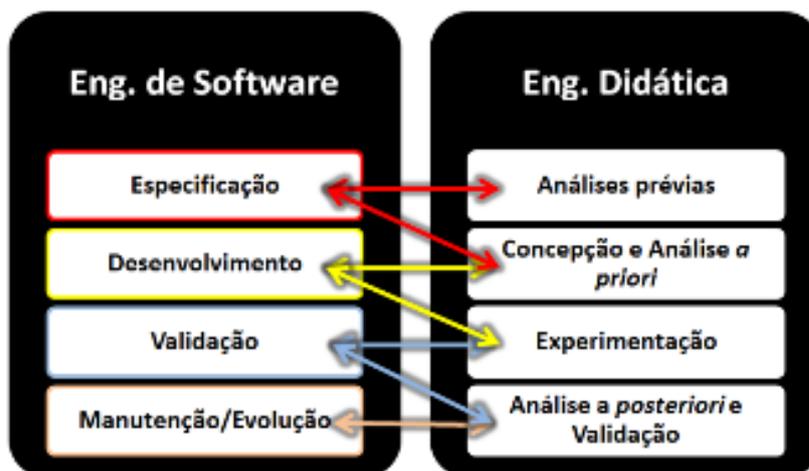
É importante destacar que esse estudo de caso foi realizado por meio de uma colaboração com outra pesquisa em desenvolvimento, que visava a criação de um recurso tecnológico para contribuir para o ensino e a aprendizagem de funções matemáticas (SILVA, 2016), permitindo, deste modo, colocar em prática o processo de desenvolvimento de software da EDI enquanto o próprio processo ainda estava sendo construído.

Segue uma citação de Tiburcio (2020) falando sobre esse trabalho colaborativo realizado entre sua pesquisa de mestrado e a pesquisa de Silva (2016):

É válido ressaltar o caráter colaborativo do estudo de caso realizado, visto que foi possível colocar em prática nossa estrutura de elaboração de software ainda quando a mesma estava sendo criada. Em parceria com Silva (2016), que possuía foco no estudo do desenvolvimento de um recurso tecnológico para o ensino e aprendizagem de funções matemáticas, criamos situações de colaboração em que uma pesquisa utilizava dos resultados imediatos da outra, funcionando como uma retroalimentação. [...] Ainda segundo os autores, esse processo observa os métodos da engenharia de requisitos integrados com uma Engenharia Didática, sendo assim definida como Engenharia Didático-Informática, na qual são contempladas as potencialidades teóricas (do ensino e aprendizagem) e tecnológicas (da computação) (TIBURCIO, 2020, p. 35).

A pesquisa de Tiburcio (2016) mostrou, como resultado, que os produtos desenvolvidos com a perspectiva de articulação entre elementos teóricos sobre o ensino e a aprendizagem da Matemática e as potencialidades tecnológicas atuais podem contribuir para a especificação e a implementação dos requisitos educativos. Partindo dessa consideração, o autor apresenta um diagrama que ilustra uma proposta de articulação entre a Engenharia Didática (ARTIGUE, 1996) e a Engenharia de Software (SOMMERVILLE, 2011):

Figura 3 – Articulação das fases das Engenharias de Software e Didática



Fonte: Tiburcio (2016).

A proposta de integração entre as engenharias inicialmente relaciona a fase de Especificação do software às fases de Análises Prévias e Concepção e Análise *a priori* da Engenharia Didática, de modo a fornecer elementos para a fase seguinte da Engenharia de Softwares: o Desenvolvimento; que, por sua vez, utiliza-se de elementos da Concepção e Análise *a priori* para dar início ao processos de Experimentação do software, no sentido de pôr em funcionamento o recurso em situações de uso, com o objetivo de verificar se foram atendidos os objetivos estipulados (TIBURCIO, 2016).

O processo de Validação da Engenharia de Software é relacionado às fases de Experimentação e Análise *a posteriori* e Validação da Engenharia Didática, considerando que a validação das sequências didáticas, na Engenharia Didática, se dá pela confrontação entre as hipóteses levantadas anteriormente e a experimentação da sequência.

Considerando também que a validação se dá por uma análise teórica e outra experimental, há outro tipo de confrontação: as situações de uso, funcionalidades e objetivos do software com o que se conseguiu desenvolver na experimentação, permitindo, com isso, a implementação do software ou a conclusão do desenvolvimento.

Por fim, as etapas de Manutenção/Evolução e Análise *a posteriori* e Validação são relacionadas tendo em vista que os elementos oriundos da última fase da Engenharia Didática fomentam o aperfeiçoamento e evolução do software desenvolvido.

A fase de validação do software é composta por testes, pela criação de situações de utilização, por observações, buscando verificar o pleno funcionamento do software, por correção dos erros encontrados e pela conferência da performance de todas as características idealizadas a construção (TIBURCIO, 2016).

A pesquisa de Tiburcio (2016) apresentou como resultado a primeira estrutura da Engenharia Didático-Informática, contendo uma versão do modelo de processo de software. A versão inicial da EDI foi validada por meio da criação do software *Function Studium* (Silva, 2016) e dos resultados de sua utilização.

Mesmo alcançando todos os objetivos estipulados para a sua pesquisa, Tiburcio (2016) aponta como encaminhamentos futuros de seu trabalho uma melhor explanação de cada fase da EDI, além da necessidade de delimitações mais explícitas de como cada etapa deve ser considerada.

O autor ainda apresenta como encaminhamento para futuras pesquisas a necessidade de uma atualização do referencial teórico utilizado, incluindo investigações em torno da Engenharia Didática de Segunda Geração e um novo olhar para os referenciais da Engenharia de Softwares, para atender a demandas tecnológicas atuais, o que justifica a continuidade das investigações acerca da Engenharia Didático-Informática.

#### 2.4 VERSÃO ATUAL DA ENGENHARIA DIDÁTICO-INFORMÁTICA

Buscando dar seguimento às indicações para futuras pesquisas levantadas em Tiburcio (2016), relacionadas à necessidade de aperfeiçoamento da EDI, e com a motivação de analisar o modo como a EDI estava sendo utilizada em pesquisas posteriores, tendo em vista que os softwares *Function Studium* (SILVA, 2014), *Conics Studium 3D* (SIQUEIRA, 2019) e *Magnitude Studium* (SILVA, 2019) foram desenvolvidos a partir da utilização da EDI como fundamentação para a produção de softwares educativos, o autor deu início a sua pesquisa de doutorado (TIBURCIO, 2020).

Além da análise dos usos da EDI nos trabalhos citados no parágrafo anterior, Tiburcio percebeu que seria importante realizar uma abordagem histórica para resgatar as engenharias de alguns softwares com pesquisas consolidadas na Educação Matemática, de modo a responder algumas questões que surgiram, como as apresentadas na citação a seguir:

“se não existe uma engenharia específica para o desenvolvimento de software educativo, como esses software foram desenvolvidos?”; “Como foram idealizados?”; “Consideravam questões teóricas sobre o ensinar e o aprender?”; “Eram fundamentados em teorias didáticas, cognitivas, epistemológicas?”; e ainda “Tentavam sanar dificuldades observadas com os potenciais da tecnologia?” (TIBURCIO, 2020, p. 20).

Deste modo, o autor selecionou os softwares *Casyopée* (LAGRANGE, 2005), *Function Probe* (CONFREY, 1992) e *Modellus* (TEODORO, 2002) para realizar um estudo histórico de resgate das engenharias utilizadas nos seus desenvolvimentos. A escolha dos softwares se deu por se tratar de recursos digitais da mesma tipologia que os softwares criados por meio de uma EDI, os ambientes de simulação/micromundos.

A partir do contexto apresentado, Tiburcio levantou as seguintes hipóteses para a sua nova pesquisa em torno da EDI:

1. Compreender metodologias de desenvolvimento de software educativo colabora para a criação de modelizações de processos; 2. A realização de um estudo histórico traz subsídios para o aprimoramento da Engenharia Didático-Informática; 3. A análise das utilizações da EDI, na sua primeira versão, produz elementos para aprimorar a modelização do processo de software (TIBURCIO, 2020, p. 21).

Atrelado às hipóteses levantadas, Tiburcio traz, como objetivo geral de sua pesquisa, “aperfeiçoar a Engenharia Didático-Informática, bem como o modelo de processo de desenvolvimento de software dessa metodologia, realizando uma abordagem histórica e analítica” (TIBURCIO, 2020, p. 21). Cada uma das hipóteses levantadas pelo autor se articula com seus objetivos específicos:

- Analisar a utilização da EDI em projetos de desenvolvimento de software educativos como meio de obter fundamentação para aprimorar o processo em evolução;
- Investigar modelizações de desenvolvimento de software educativo em uma abordagem histórica para obter informações de modo a contribuir com a nova versão da EDI;
- Observar as contribuições dos estudos analisados para lançar nova versão do modelo de processo da EDI (TIBURCIO, 2020, p. 21).

Para propor uma metodologia de pesquisa que possibilitasse o alcance de seus objetivos, Tiburcio (2020) dividiu sua investigação em duas etapas: um *resgate histórico das metodologias em softwares* e a *análise dos projetos que utilizaram a EDI*. Após a realização das duas etapas, foi feito um levantamento de aplicações de implementações para a proposição de uma nova versão do modelo de processo de software da EDI.

O *resgate histórico das metodologias em softwares* foi realizado por meio de entrevistas *online* com criadores dos softwares educativos escolhidos e estudos relacionados aos softwares em questão. Em relação à *análise dos projetos que utilizaram a EDI*, Tiburcio (2020) analisou as potencialidades e limitações do uso da EDI nos projetos de desenvolvimentos de softwares educativos.

Como resultados do resgate histórico, Tiburcio (2020) conseguiu compreender um padrão de modelização das engenharias analisada, confirmando, portanto, a primeira hipótese de sua pesquisa. Além disso, ainda em relação ao resgate histórico, a segunda hipótese de pesquisa também conseguiu ser confirmada, já que os resultados dessa análise indicaram que “existe a atenção por parte dos desenvolvedores em considerar teorias sobre o ensino, a aprendizagem, o uso de tecnologias, as concepções, entre outras contribuições, para desenvolver recursos digitais educativos” (TIBURCIO, 2020, p. 184).

Segundo o autor, essa fase de resgate histórico se mostrou muito útil para o aperfeiçoamento da EDI, por serem observadas semelhanças entre as engenharias utilizadas e critérios similares de levantamento de requisito desses softwares (TIBURCIO, 2020).

Em relação aos resultados obtidos a partir da análise dos projetos que utilizaram a EDI, Tiburcio (2020) enfatiza que os autores desses projetos de pesquisa apontaram para a necessidade de aprimoramento dessa metodologia no que se refere ao detalhamento de cada parte do processo de desenvolvimento. Além disso, percebeu-se a necessidade de atualização dos referenciais teóricos e metodológicos. Ao considerar as questões presentes nos resultados dessa etapa da pesquisa para a formalização da versão atual da EDI, a terceira hipótese da pesquisa de Tiburcio (2020) foi confirmada.

Ao sistematizar os resultados obtidos em sua pesquisa, Tiburcio (2020) destaca que conseguiu alcançar os objetivos propostos ao aperfeiçoar a EDI como uma proposta de metodologia de desenvolvimento de software educativo, que leva em consideração aspectos sobre o ensino, a aprendizagem, a epistemologia e as tecnologias digitais, além de características relevantes de outras naturezas.

O autor ainda apresenta detalhadamente as implementações realizadas na nova versão da EDI, divididas em cinco blocos: 1. Caracterização da Engenharia Didático-Informática como uma metodologia para o desenvolvimento de software educativo; 2. Atualização dos fundamentos teóricos e metodológicos das engenharias Didática e de Software; 3. Articulação teórica explícita entre as engenharias Didática e de Software; 4. Apresentação esclarecedora de cada fase/ciclo do modelo de processo de software; 5. Flexibilização do modelo de processo considerando um ciclo, removendo etapas rígidas e lineares (TIBURCIO, 2020).

Assim como foi apresentado na introdução da presente pesquisa, Tiburcio (2020) apresenta, como encaminhamentos futuros de sua investigação, três questões vistas como relevantes, principalmente pelo fato de reconhecer a constante evolução das tecnologias digitais e das pesquisas sobre o ensino e aprendizagem com o auxílio desses recursos.

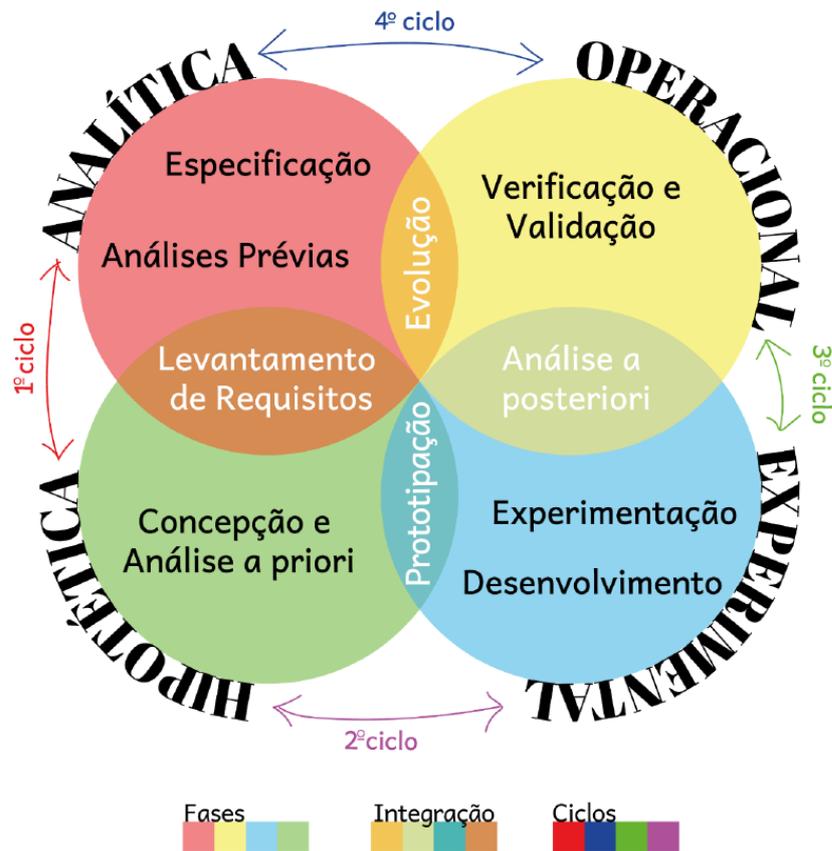
Relembramos, a seguir, os encaminhamentos futuros de Tiburcio (2020): *pôr em prática a nova EDI*, tendo em vista que essa metodologia ainda não foi colocada em funcionamento desde que foi atualizada; *tipologia de software*, caracterizada pela possibilidade de utilização da EDI para o desenvolvimento de softwares educativos de tipologias diferentes dos ambientes de simulação/micromundos; e *verificar como softwares educativos são desenvolvidos*, sugerindo que essa ampliação de pesquisa, numa perspectiva de revisão sistemática, por exemplo, poderá auxiliar na compreensão das percepções de professores e pesquisadores quanto à utilização e ao desenvolvimento de software educativo.

Na seção seguinte será apresentado o principal produto da atual versão da EDI: o novo modelo de processo de software educativo da Engenharia Didática-Informática (TIBURCIO, 2020), a ser utilizado na presente pesquisa para a concepção de uma versão digital do jogo Mankala Colhe Três.

#### **2.4.1 Modelo de Processo de software da Engenharia Didático-Informática**

O atual Modelo de Processo da Engenharia Didático-Informática, proposto por Tiburcio (2020) possui quatro fases (analítica, hipotética, experimental e operacional), organizadas em quatro ciclos (analítico-hipotético, hipotético-experimental, experimental-operacional e operacional-analítico), que representam integrações entre as fases. Segue o esquema do atual Modelo de Processo da EDI:

Figura 4 – Modelo de processo de software da Engenharia Didático-Informática



Fonte: adaptado de Tiburcio (2020).

Uma particularidade do modelo de processo de software proposto pela EDI é que, diferentemente dos modelos de processo de software geralmente utilizados para a criação de produtos (não necessariamente educativos), em que o processo apresenta rigidez entre etapas e procedimentos, na EDI, é possível revisitar etapas já realizadas.

Para exemplificar essa dinâmica, Tiburcio (2020) aponta que, a partir do que foi levantado nas análises preliminares, as especificações podem ser revisitadas e até modificadas, para que o protótipo possa ser iniciado posteriormente, idealizando situações de uso que possam contribuir para o ensino e aprendizagem do que foi delimitado.

Falando um pouco mais sobre cada etapa do modelo de processo de software educativo da EDI, na etapa de Especificação, são delimitados os problemas existentes, relacionados ao ensino e a aprendizagem, que o software a ser desenvolvido ajudará a solucionar, em que são definidos quais saberes serão abordados e como as tecnologias podem contribuir na compreensão deles.

Além disso, é na Especificação que são definidos os potenciais usuários do software a ser desenvolvido e a composição da equipe de desenvolvimento. A EDI considera a composição

de equipes transdisciplinares, garantindo que os saberes das áreas envolvidas sejam integrados numa perspectiva de união para que novos saberes sejam criados (TIBURCIO, 2020).

A outra etapa que compõe a fase analítica da EDI é a de Análises Prévias. Essa etapa é caracterizada pela realização de um levantamento analítico dos temas que podem fornecer encaminhamentos didáticos, epistemológicos, cognitivos e informáticos acerca do conhecimento delimitado na especificação, de modo a subsidiar o levantamento de requisitos iniciais do software (TIBURCIO, 2020).

Essa etapa deve ser orientada pelos questionamentos apresentados no quadro a seguir:

Quadro 2 – Direcionamentos para as análises prévias da EDI

<i>DIMENSÕES</i>	<i>QUESTIONAMENTOS</i>
<i>Cognitiva</i>	<i>Existem indicações na literatura de como o estudante aprende? Quais dificuldades de aprendizado são identificadas? Quais etapas são elencadas para a construção do conhecimento?</i>
<i>Didática</i>	<i>Qual é o estado atual do ensino do conhecimento? Quais são as consequências desse ensino? Quais são as dificuldades em ensinar esse conhecimento?</i>
<i>Epistemológica</i>	<i>Quais intervenções são realizadas para adaptar o saber matemático ao saber a ser ensinado? Quais são os aspectos do conhecimento que podem dificultar e/ou facilitar a aprendizagem?</i>
<i>Informática</i>	<i>Quais são as contribuições tecnológicas que o software deve conter para auxiliar na compreensão e ensino dos conhecimentos?  Em que aspectos as tecnologias digitais influenciam no currículo e nas mudanças das práticas docente e discente?</i>

Fonte: Tiburcio (2020).

A etapa Levantamento de Requisitos, conforme ilustrado na Figura 4, representa uma das etapas de conexão entre fases da EDI. O estudo de requisitos do software é iniciado a partir dos encaminhamentos obtidos na fase *analítica* da EDI, que, por sua vez, precisam ser estruturados de forma detalhada e clara, contendo as características que o software deve conter para alcançar os objetivos educativos esperados. O autor reforça que “os requisitos podem ser modificados considerando as hipóteses e a experimentação” (TIBURCIO, 2020, p. 172), por isso, o autor enfatiza a importância de que o documento de requisitos seja de fácil compreensão e permita modificações.

Tiburcio (2020) ressalta que a etapa de levantamento de requisitos engloba a obtenção, a análise, a especificação e a validação dos requisitos, além de que, pelo fato de a EDI propor um modelo de processo de software cíclico, é recomendável que o gerenciamento de requisitos seja realizado durante todo o ciclo de vida do software.

O autor divide a obtenção dos requisitos em quatro etapas:

1. Questionamentos iniciais: *Como o ensino e a aprendizagem podem ser favorecidos? Como a compreensão dos saberes é auxiliada com o uso do software? Quais recursos e situações o software propõe para ajudar o usuário a compreender os conhecimentos?*
2. Análise externa: *quais funcionalidades existem em produtos da área? Quais são os possíveis diferenciais do software que se pretende desenvolver? O que o software trará de novo referente ao que já existe?*
3. Documento de requisitos: sugestão de produção de quadro de documentação de requisitos, categorizados por dimensão, conforme ilustrado abaixo:

Quadro 3 – Sugestão de documentação de requisitos da EDI

DIDÁTICOS	COGNITIVOS	EPISTEMOLÓGICOS	INFORMÁTICOS	OUTRAS NATUREZAS

Fonte: Tiburcio (2020)

4. Verificação dos requisitos quanto à pertinência, consistência e integridade. Pode haver necessidade de modificação de requisitos após essa verificação (TIBURCIO, 2020).

O segundo ciclo da EDI, chamado *hipotético-experimental*, é composto pela fase *hipotética*, representada pela etapa de Concepção e Análise *a priori*, pela Prototipação como etapa de integração entre as fases *hipotética* e *experimental* e pelo desenvolvimento do protótipo, para que os testes possam ser iniciados na etapa seguinte (TIBURCIO, 2020).

A Concepção e Análise *a priori* da EDI, possui como característica o desenvolvimento de situações de utilização do software, envolvendo, inclusive, detalhes sobre distintas possibilidades de interações dos usuários, considerando as exigências advindas da etapa de Levantamento de Requisitos.

Em relação à etapa de Prototipação do software, Tiburcio (2020) ressalta que ela precisa ocorrer simultaneamente ao desenvolvimento das situações de utilização do software da Análise *a priori*, já que a prototipação tem a função de permitir que os profissionais de design e

arquitetura de software compreendam e execute os requisitos e funcionalidades do software. Complementando o que foi dito sobre as etapas de Concepção e Análise *a priori* e Prototipação, Tiburcio coloca que

As situações de utilização, bem como o protótipo, devem ser construídas objetivando superar os problemas de ensino e aprendizagem (e de outras naturezas) descritos pela equipe considerando o levantamento teórico nas dimensões da EDI e, principalmente, analisando o documento de requisitos. Esse procedimento serve para fundamentar o software na base teórica e metodológica em pesquisas com resultados consolidados que foram analisadas pela equipe (TIBURCIO, 2020, p. 174-175).

Ainda em relação à Prototipação, o autor indica que seja feita “em telas”:

os líderes da equipe, de posse dos requisitos, podem utilizar editores de textos ou imagens para simular como será o ambiente a ser desenvolvido, com as funcionalidades, botões, menus, etc. Assim exibindo as telas iniciais do produto e como algumas de suas funções a serem executadas (TIBURCIO, 2020, p. 175).

Continuando sobre o ciclo hipotético-experimental, Tiburcio (2020) enfatiza que as situações que foram idealizadas são implementadas no software (ou não) ao passo que a experimentação acontece. Além disso, simultaneamente à elaboração do produto, acontecem os testes iniciais, o *feedback* da utilização, as percepções dos membros da equipe de desenvolvimento, as compreensões e incompreensões das funcionalidades.

O Desenvolvimento, caracterizado pelo *design* e a arquitetura do software, é realizado a partir da fundamentação das ideias do protótipo, resultando em componentes internos e interfaces. O autor também enfatiza que acredita que a etapa de desenvolvimento ocorre simultaneamente à experimentação, em termos de utilização do protótipo pela equipe, que traz subsídios para guiar características necessárias ao produto. Além disso, “o protótipo desenvolvido precisa ser utilizado pelos possíveis usuários e essas experiências devem ser registradas a fim de obter elementos de análise e implementação do software” (TIBURCIO, 2020, p. 175).

Com o desenvolvimento realizado, indica-se que a equipe crie um manual do usuário, ou algum tipo de documento instrutivo, com as funcionalidades básicas para utilização na fase de Experimentação.

O ciclo *experimental-operacional*, terceiro da EDI, é iniciado pela etapa de Experimentação e contém também elementos da Análise *a posteriori*.

É na etapa de Experimentação que o software é colocado em situações de uso, tanto pela equipe de desenvolvedores, para que o software possa ter suas funcionalidades testadas, quanto

por usuários em potencial do produto, possibilitando a testagem da validade das situações propostas. Essas testagens já representam elementos da etapa de Análise *a posteriori*, como pode ser observado na citação a seguir:

Uma das finalidades da experimentação, já diretamente relacionada com a *análise a posteriori*, é a testagem da validade das situações propostas. A equipe deve observar se o software auxilia os usuários a construir os conhecimentos que são esperados. Dessa maneira, à medida em que a experimentação ocorre, informações importantes são levantadas. Por um lado, realiza-se a análise dos requisitos levantados: se eles atendem as expectativas, se auxiliam o ensino e a aprendizagem, se ajudam na compreensão dos saberes, se os recursos tecnológicos são eficazes para atender o que foi planejado; por outro lado são verificadas as falhas, sugestões de implementações por parte dos usuários e pela análise da equipe (interface, comandos, botões, menus, etc), bugs e outras inconsistências no uso, ou seja, as etapas de análise *a posteriori* e validação tem início ainda na experimentação, acontecem de forma simultânea (TIBURCIO, 2020, p. 177-178).

A etapa de Análise *a posteriori* também se relaciona diretamente à etapa de Validação do software. De fato, a partir do confronto entre as hipóteses iniciais com o que pôde ser verificado na análise da Experimentação e das implementações realizadas pela equipe, o atendimento dos objetivos do software pode ser analisado (TIBURCIO, 2020).

O quarto ciclo da EDI, denominado *operacional-analítico*, contém as etapas de Verificação e Validação e de Evolução do software. A Verificação e a Validação do software estão atreladas tanto à confirmação do atendimento dos objetivos esperados do conjunto teórico-hipotético quanto à contribuição do software para o ensino e a aprendizagem dos conhecimentos especificados (TIBURCIO, 2020).

Tiburcio (2020) indica que a validação seja realizada de duas formas: “1. Teórica – consiste em verificar se as teorias e hipóteses, com as situações e funcionalidades do software, devem ser refutadas ou aprovadas; 2. experimental – se a utilização do software apresenta contribuições efetivas para os problemas elencados na fase de sua concepção” (TIBURCIO, 2020, p. 179).

Como o modelo de processo de software da EDI é cíclico, a fase *operacional* se relaciona a uma nova fase *analítica*, por meio da Evolução do software. A Evolução do software se baseia nos encaminhamentos de alterações resultantes da etapa de Validação e, assim que as indicações de ajustes são elencadas, dá-se início a uma nova Especificação do software, considerando a possibilidade de criação novos objetivos para o software e, deste modo, recomeçando o ciclo da Engenharia Didático-Informática.

Ainda sobre a evolução do software, Tiburcio (2020) enfatiza que uma de suas premissas é a manutenção do software. Deste modo, ele propõe que a equipe de desenvolvedores considere

quatro tipos de manutenção do software desenvolvido: corretiva; adaptativa; perfectiva; e preventiva.

Finalizamos esse resgate da elaboração da Engenharia Didático-Informática ressaltando ao leitor que todo detalhamento trazido neste capítulo foi idealizado para que pudéssemos não apenas oportunizar a difusão das discussões de investigações anteriores à presente pesquisa, em termos de reflexões sobre a Engenharia de Softwares Educativos e proposição de uma metodologia de desenvolvimento de softwares educativos integrados aos princípios teórico-metodológicos da Engenharia Didática, mas também para embasar as discussões e propostas de atualizações e adaptações da Engenharia Didático-Informática que virão mais adiante neste trabalho.

## 2.5 METODOLOGIA DA PESQUISA

Como apresentamos anteriormente, a presente pesquisa possui o objetivo geral de propor um modelo de processo de software da Engenharia Didático-Informática destinado à produção de jogos matemáticos de simulação a serem utilizados como instrumentos centrais na vivência de situações didáticas.

Retomando também os objetivos específicos da pesquisa, buscamos:

- Realizar um estudo do Mankala Colhe Três e das noções que podem ser mobilizadas por meio do jogo, considerando aspectos de natureza epistemológica, cognitiva, didática e informática, de modo a estabelecer requisitos educativos para uma versão digital do jogo;
- Identificar as particularidades da utilização dos princípios da Engenharia Didático-Informática na modelagem de uma versão digital do Mankala Colhe Três;
- Apresentar contribuições metodológicas para a Engenharia Didático-Informática.

Visando alcançar os objetivos estipulados para a pesquisa, a metodologia será conduzida a partir da concepção de uma versão digital do jogo Mankala Colhe Três seguindo a realização das fases presentes no modelo de processo da Engenharia Didático-Informática, proposto por Tiburcio (2020) e apresentado na seção 2.4 deste capítulo, de modo a obter informações sobre particularidades do processo de produção de jogos matemáticos de simulação para situações didáticas e, com isso, propor um modelo de processo específico para a produção de softwares educativos dessa tipologia.

### 3 REALIZAÇÃO DO CICLO ANALÍTICO-HIPOTÉTICO NA CONCEPÇÃO DO MANKALA COLHE TRÊS DIGITAL

Neste capítulo realizaremos as etapas do ciclo Analítico-hipotético do modelo de processo da Engenharia Didático-Informática para a concepção de uma versão digital do jogo Mankala Colhe Três, que também chamaremos de Mankala Colhe Três Digital.

Como detalhado na seção referente à metodologia de nossa pesquisa, nossa proposta de condução da investigação guia-se pela realização de cada uma das etapas do modelo de processo da EDI, de modo a obter informações sobre particularidades do processo de produção de jogos matemáticos de simulação para situações didáticas.

Sendo assim, apresentaremos neste capítulo a realização das seguintes etapas do modelo de processo da EDI para a concepção do Mankala Colhe Três Digital: Especificação; Análises Prévias; e Levantamento de requisitos.

#### 3.1 ESPECIFICAÇÃO

Como vimos no capítulo anterior, a Especificação da EDI é caracterizada pela etapa em que se definem os conhecimentos matemáticos abordados com o software, os conhecimentos relacionados que também devem ser trabalhados, a composição da equipe de produção e o público-alvo a quem se destina o software.

Para direcionar a fase de Especificação, Tiburcio (2020) apresenta alguns questionamentos, apresentados na citação que segue:

“Quais são os problemas percebidos que o software poderá se apresentar como solução?  
Quais conhecimentos se pretende abordar na utilização do software?  
Considerando as relações entre os saberes delimitados, quais conceitos e definições devem estar presentes?  
Qual será o diferencial da utilização desse software comparado a um ambiente papel e lápis? (TIBURCIO, 2020, p. 170).

Ao buscar refletir sobre os questionamentos acima, precisamos nos remeter às pesquisas envolvendo o jogo Mankala Colhe Três, que, como já apresentamos anteriormente, foi desenvolvido no âmbito do Projeto “Formação docente: interdisciplinaridade e ação docente – Projeto Rede (GITIRANA *et al.*, 2013) e tinha como uma das intenções didáticas permitir o trabalho com os divisores de um número.

Os criadores do jogo, após investigações e experiências práticas com o Mankala Colhe Três, elencaram as seguintes finalidades educacionais:

- Desenvolver estratégias de quantificar mentalmente;
- Resolver problemas com situações mistas: aditivas e multiplicativas;
- Dividir por cálculo mental;
- Mapear as possibilidades;
- Explorar as possibilidades de distribuição em partes iguais a partir das quantias existentes nas covas:
  - Reconhecer os divisores de um determinado número;
  - Identificar múltiplos de um número;
  - Reconhecer números primos e compostos (ANDRADE *et al.*, 2013, p. 29).

Como foi descrito anteriormente, Santos (2014) desenvolveu uma pesquisa de mestrado fundamentada na Teoria das Situações Didáticas (BROUSSEAU, 2008), cujo objetivo geral era investigar as contribuições do jogo Mankala Colhe Três para a aprendizagem de conhecimentos matemáticos por alunos de 6º ano do Ensino Fundamental.

Dentre os resultados de sua pesquisa, o autor avaliou positivamente alguns aspectos relacionados a escolhas metodológicas do uso do jogo em sua experimentação, ao enfatizar que o caráter adidático das situações de jogo com o Mankala Colhe Três pôde ser fortalecido a partir de um processo de devolução bem conduzido e que as diferentes modalidades de interação entre os sujeitos da pesquisa permitiram um bom equilíbrio entre a dimensão lúdica e a intencionalidade didática, além de fortalecerem a evolução das estratégias durante as partidas (SANTOS, 2014).

Em termos de mobilização dos conhecimentos esperados, Santos (2014) afirma que a utilização do jogo permitiu o trabalho com divisores de um número de modo lúdico e com a matemática sendo mobilizada implicitamente, por meio das jogadas, o que favoreceu o levantamento de hipóteses e a descoberta de importantes propriedades dos números primos, dos números compostos, da quantidade de divisores de um número, entre outras.

Um aspecto importante que podemos inicialmente ressaltar, relacionado ao Mankala Colhe Três como recurso para o ensino e a aprendizagem de Matemática, é que, assim como muitos jogos matemáticos, ele permite a mobilização de: noções da matemática que não são necessariamente institucionalizadas como conteúdos matemáticos escolares; e da dimensão lúdica, que deve permear as situações de uso do jogo.

Questões envolvendo as características dos saberes, incluindo essas distinções entre noções que são, ou não, consideradas como objetos de saber da Matemática, são profundamente discutidas por Yves Chevallard (1991), ao tratar da Transposição Didática.

Chevallard (1991) afirma que, para que um *objeto de saber* possa ser considerado como tal, no campo dos agentes do sistema de ensino, é necessário que sua inserção no sistema dos “objetos a ensinar” seja apresentada como útil para a economia do sistema didático.

Conforme apresentamos na seção referente às origens da Engenharia Didático-Informática, a Transposição Didática se interessa pelo conjunto de processos que transforma o saber sábio (objeto de saber) em saber a ensinar (objeto de ensino) (CHEVALLARD, 1991).

Esses processos de transformações adaptativas são definidos por Chevallard como: a epistemologia do regime didático do saber; a noosfera; as criações didáticas; a vigilância epistemológica; a desincretização do saber; a despersonalização do saber; a programabilidade do saber; a publicidade do saber; o controle social das aprendizagens; a dialética antigo/novo; a obsolescência externa e interna; a cronogênese; e a topogênese (CHEVALLARD, 1991).

Ao discutir a epistemologia do regime didático do saber, Chevallard (1991) diferencia as *noções matemáticas*, *paramatemáticas* e *protomatemáticas*. Para o teórico, as *noções matemáticas* são os conteúdos de saber. Já as *noções paramatemáticas* não podem ser consideradas como objetos de ensino, mas como objetos de saber “auxiliares”, que são vistos como necessários para o ensino e a aprendizagem dos próprios objetos matemáticos e que devem ser “aprendidos” (ou melhor, “conhecidos”), mas não são “ensinados”, como, por exemplo a noção de parâmetro, a noção de equação, a noção de demonstração.

Em relação às *noções protomatemáticas*, Chevallard afirma que se trata de uma camada “mais profunda” de noções, que são mobilizadas implicitamente pelo contrato didático<sup>5</sup>. Essas noções podem ser entendidas como as capacidades desenvolvidas, como criar e testar hipóteses, análise de dados etc. (CHEVALLARD, 1991).

Deste modo, em relação aos conhecimentos que se pretende abordar na utilização do Mankala Colhe Três Digital, precisamos destacar que nossa compreensão é que, no caso do jogo, este aspecto possui particularidades se comparado aos demais softwares educativos que se utilizaram de versões anteriores da EDI para a produção de ambientes de simulação/micromundos.

É possível perceber que, no caso da nossa proposta, ao invés de procurarmos conceber um software a partir de análises do conhecimento matemático delimitado, focaremos nossas análises para o jogo como recurso já existente em uma versão física e que, por sua vez, já carrega consigo um leque de noções importantes para a aprendizagem de matemática que

---

<sup>5</sup> O Contrato Didático é um conceito central da Teoria das Situações Didáticas (BROUSSEAU, 2008) e pode ser compreendido como um sistema de responsabilidades recíprocas – predominantemente implícitas – entre o professor e o aluno em uma situação didática, na qual existe um conhecimento matemático visado.

podem ser mobilizadas, além de outros aspectos relacionados ao potencial educativo como um todo e que não se referem, necessariamente, aos conhecimentos matemáticos, como, por exemplo, as importantes atitudes de convivência que podem ser propiciadas por meio de situações de jogo no ambiente pedagógico.

Sendo assim, partiremos da distinção apresentada por Chevallard, das *noções matemáticas*, *paramatemáticas* e *protomatemáticas*, para elencar as noções que puderam ser identificadas como mobilizadas em situações de uso do jogo em estudos anteriores, principalmente em Santos (2014), e que pretendemos que continuem constituindo o foco principal do jogo em sua versão digital.

Como *noções matemáticas*, podemos destacar a *divisibilidade* como principal conhecimento que se pretende abordar, que se relaciona aos conceitos de *múltiplos de um número*, *números primos* e *números compostos*. Como *noções paramatemáticas*, podemos elencar as capacidades de: quantificar mentalmente; dividir por cálculo mental; mapear possibilidades mentalmente; reconhecer divisores e múltiplos de um número; reconhecer números primos e números compostos. Em relação às noções *protomatemáticas*, podemos elencar as capacidades de reconhecimento de padrões, de formulação e de testes de estratégias.

Refletindo ainda sobre os questionamentos norteadores da etapa de Especificação apresentados por Tiburcio (2020), em termos de analisar o diferencial da utilização do Mankala Colhe Três Digital, se comparado tanto a um ambiente papel e lápis quanto à versão física do jogo, gostaríamos de apresentar os principais encaminhamentos futuros da pesquisa de Santos (2014):

- Tentar produzir o mapeamento completo das jogadas do Mankala Colhe Três com o auxílio de recursos tecnológicos mais sofisticados, de modo a subsidiar o professor/pesquisador na criação de atividades que simulem de situações reais de jogo e no aprimoramento de critérios de análise de estratégias dos jogadores;
- Possibilitar a construção de variações do jogo, como a modificação da quantidade inicial de sementes, do número de covas do tabuleiro ou até mesmo das regras do jogo, de modo a permitir uma ampliação de uso do jogo por meio da possibilidade de evolução das situações didáticas;
- Enfatizar o viés da Etnomatemática com o Mankala Colhe Três para explorar melhor o aspecto cultural do jogo, associando-o a sua provável origem africana;
- Pensar em diferentes formatos de dispositivo experimental, que favoreçam a passagem dos estudantes/sujeitos de pesquisa pelos tipos de situações didáticas (ação, formulação,

validação e institucionalização) por meio de modelos variados de interação entre os jogadores;

- Propor sequências metodológicas mais longas, favorecendo a obtenção de uma maior diversidade de informações e de situações de jogo em futuras pesquisas;
- Propor um dispositivo mais amplo em relação à quantidade de estudantes/sujeitos participantes da pesquisa, que poderia ser de uma turma inteira, favorecendo a presença de momentos de institucionalização, tipo de situação didática que não foi vivenciado no dispositivo experimental da pesquisa.

Tendo em vista que a presente pesquisa de doutorado considera as motivações advindas das indicações de encaminhamentos futuros de Santos (2014) para a concepção do jogo matemático digital por meio da Engenharia Didático-Informática, como foi detalhado no capítulo da Introdução deste trabalho, é possível perceber que os encaminhamentos levantados por Santos (2014) apresentam não apenas indicações de possíveis respostas às questões norteadoras da fase de Especificação, mas já revela, de certo modo, um esboço de importantes requisitos que o Mankala Colhe Três Digital deverá conter, visando a ampliação do potencial educativo do jogo.

Com relação ao público-alvo a quem se destina o Mankala Colhe três Digital, é importante destacar que sua versão física foi idealizada para ser trabalhada com estudantes dos Anos Iniciais do Ensino Fundamental, assim como os demais jogos produzidos no Projeto Rede. No entanto, como já citamos anteriormente, a pesquisa de Santos (2014) utilizou, como sujeitos de sua experimentação e análise, estudantes de 6º ano de Ensino Fundamental.

O autor enfatiza que essa escolha se deu por dois fatores que se complementam: sua formação profissional em Licenciatura em Matemática e a proximidade entre o que se espera de estudantes de 6º ano do Ensino Fundamental, em termos de mobilização de conhecimentos matemáticos, e as finalidades educacionais do jogo, levantadas pelos criadores no Projeto Rede (SANTOS, 2014).

Na nossa pesquisa, tentaremos não limitar o público-alvo a anos escolares específicos, pois nos estimulamos com as possibilidades que uma versão digital do jogo poderá trazer, se considerarmos as indicações de ampliação do seu potencial educativo realizadas por Santos (2014).

Sendo assim, partiremos da hipótese de que o Mankala Colhe Três Digital poderá servir de recurso didático tanto para o público em que já há resultados de pesquisas consolidados, envolvendo sua versão tradicional, quanto para uma ampliação desse público-alvo, resultante

das adaptações presentes em sua versão digital em termos estruturais do jogo e metodológicas de seu uso.

Por fim, quanto à composição da equipe de produção do jogo, concordamos com o que é trazido por Tiburcio (2020), sobre a necessidade de que esse seja um trabalho em uma perspectiva transdisciplinar. No entanto, compreendemos também que há um desafio nessa proposta, que envolve questões de diversas dimensões, como a limitação de recursos para que haja profissionais de diversas áreas disponíveis integrados ao projeto, ou as dificuldades de que os profissionais envolvidos na equipe possuam um perfil que favoreçam esse tipo de trabalho transdisciplinar.

Consideraremos, portanto, que o olhar transdisciplinar será dado não em termos de envolvermos uma equipe ampla no número de profissionais, mas uma equipe que possua competência para discutir temas de diversas áreas envolvidas no projeto com propriedade e de maneira integrativa.

No nosso trabalho, destacamos como membros da equipe:

- o pesquisador, que é professor de Matemática e desenvolve pesquisas nas áreas de Didática da Matemática e de Educação Tecnológica, tendo como foco o uso de jogos para o ensino e aprendizagem de matemática;
- o orientador de pesquisa, que possui ampla experiência nas áreas de Matemática e Informática, com ênfase em Geometria e Tecnologia Educativa (pesquisa e desenvolvimento) e ensino e divulgação da matemática e das ciências;
- o coorientador de pesquisa, que desenvolve pesquisas na área de integração de princípios da engenharia de software na concepção/desenvolvimento/integração/validação/implantação de sistemas de ensino e aprendizagem; Linguagens de modelização pedagógica; Cenarização pedagógica (concepção/edição/operacionalização/monitoramento/orquestração de cenários de aprendizagem); Pensamento computacional e aprendizagem colaborativa mediada por tecnologia; Transformação digital na educação com o uso das novas TICs em meios escolares e de formação (presencial ou à distância), de forma a promover o processo de ensino e aprendizagem de estudantes (jogos sérios, dispositivos móveis, ferramentas interativas, robótica educacional, Internet das coisas etc.);
- membros do grupo de pesquisa Atelier Digitas (CAC/UFPE/CNPq), por meio de contribuições pontuais.

### 3.2 ANÁLISES PRÉVIAS

A etapa de Análises Prévia da Engenharia Didático-Informática, conforme apresentado no Capítulo 2 do presente trabalho, é caracterizada por um levantamento analítico, de modo a compreender os encaminhamentos didáticos, epistemológicos, cognitivos e informáticos dos saberes delimitados na Especificação e que pretendem ser explorados por meio do software.

No nosso caso, consideraremos como base desse levantamento vários aspectos que envolvem o uso Mankala Colhe Três Digital como recurso a ser utilizado para o ensino e a aprendizagem da Matemática e seu potencial educativo como um todo, levando em conta estudos que podem contribuir para auxiliar no levantamento de requisitos epistemológicos, cognitivos, didáticos, tecnológicos e de outras possíveis dimensões do software.

Para a realização desse estudo analítico, escolhemos utilizar seções temáticas que trarão os encaminhamentos das dimensões integradas ao longo das discussões. Iniciaremos por meio de uma discussão geral acerca do uso de jogos matemáticos na Educação Matemática.

Abriremos espaço para estudarmos mais detalhadamente todos os aspectos em torno do Mankala Colhe Três, incluindo o que é trazido pelos criadores do jogo no Projeto Rede (ANDRADE *et al.*, 2013), um estudo epistemológico acerca das noções matemáticas suscetíveis de serem mobilizadas por meio do uso do jogo, além de uma discussão sobre os jogos denominados *mankala* e o uso desses tipos de jogos para o trabalho com a *Etnomatemática*.

Ainda se tratando das investigações a respeito do Mankala Colhe Três, serão trazidas algumas considerações importantes da pesquisa de Santos (2014) referentes a toda a análise realizada em seu trabalho sobre o potencial educativo do jogo em situações didáticas e as questões que ainda ficaram em aberto para futuras investigações envolvendo o jogo.

Também daremos espaço para uma apresentação dos principais elementos da Teoria das Situações Didáticas, tendo em vista que a teoria proposta por Brousseau (1997) fundamenta uma das faces de nossa pesquisa, na medida em que voltamos o olhar para a análise da concepção de uma versão digital do Mankala Colhe Três que considere os encaminhamentos trazidos por Santos (2014).

Traremos também uma breve discussão sobre a transposição didático-informática do Mankala Colhe Três para a criação de uma versão digital, a partir da utilização das indicações metodológicas da Engenharia Didático-Informática.

### 3.2.1 Os Jogos Matemáticos na Educação Matemática

Muitas pesquisas na área de Educação Matemática apontam para o quanto os jogos matemáticos podem acrescentar para as aulas de matemática. De fato, esta é uma importante tendência metodológica da Educação Matemática e muitos são os aspectos tratados para justificarem o uso desse tipo de recurso.

Um dos fatores mais difundidos acerca do uso de jogos no ensino e na aprendizagem de matemática é o aspecto lúdico que este tipo de recurso traz para o ambiente educativo. Falando um pouco mais acerca da importância da ludicidade, Grandó (2000) afirma que, na verdade, ela sempre fez parte da humanidade e se mostra necessária para as pessoas, independentemente de suas idades:

A necessidade do Homem em desenvolver as atividades lúdicas, ou seja, atividades cujo fim seja o prazer que a própria atividade pode oferecer, determina a criação de diferentes jogos e brincadeiras. Esta necessidade não é minimizada ou modificada em função da idade do indivíduo. Exercer as atividades lúdicas representa uma necessidade para as pessoas em qualquer momento de suas vidas (GRANDÓ, 2000, p. 1).

Ainda em relação à ludicidade, os Parâmetros Curriculares de Pernambuco argumentam sobre sua importância para o desenvolvimento do estudante, tendo em vista que, por se tratar de um aspecto que está enraizado na cultura do indivíduo, a ampliação da dimensão lúdica em sala de aula representaria mais uma maneira de exploração da realidade do estudante:

Em primeiro lugar, menciona-se a necessidade de ampliar a dimensão lúdica, importante para o desenvolvimento integral do estudante. Os jogos são, ao lado disso, um elemento que favorece a inserção do estudante em sua cultura, na medida em que a dimensão lúdica está enraizada nela. Os jogos seriam, assim, mais uma forma de exploração da realidade do estudante (PERNAMBUCO, 2012, p. 36).

Outros aspectos que são propiciados por meio dos jogos no ambiente educacional e que são fundamentais para o desenvolvimento matemático do estudante são a interação e a autonomia. Ao se deparar com um desafio do jogo, os alunos naturalmente interagem e assumem uma posição ativa para buscar a solução do problema apresentado, conforme é comentado nos Parâmetros Curriculares de Pernambuco:

No âmbito pedagógico, é fundamental o aspecto interativo propiciado pela experiência com jogos matemáticos. Os estudantes não ficam na posição de meros observadores, tomando conhecimentos de novos fatos, mas se transformam em elementos ativos, na tentativa de ganhar a partida ou na busca de um caminho para a

solução do problema posto a sua frente. Tal atitude é certamente muito positiva para a aprendizagem das ideias matemáticas subjacentes aos jogos (PERNAMBUCO, 2012, p. 37).

Complementando o que foi dito, os Parâmetros Curriculares Nacionais – 3º e 4º ciclos destacam o potencial da apresentação de problemas por meio da utilização de jogos em sala de aula, tendo em vista que essa maneira de apresentar o problema propicia diversas ações dos estudantes que, do ponto de vista de sua formação, se mostram muito interessantes:

Os jogos constituem uma forma interessante de propor problemas, pois permitem que estes sejam apresentados de modo atrativo e favorecem a criatividade na elaboração de estratégias de resolução e busca de soluções. Propiciam a simulação de situações problema que exigem soluções vivas e imediatas, o que estimula o planejamento das ações; possibilitam a construção de uma atitude positiva perante os erros, uma vez que as situações sucedem-se rapidamente e podem ser corrigidas de forma natural, no decorrer da ação, sem deixar marcas negativas (BRASIL, 1998, p. 46).

Os benefícios inerentes ao uso de jogos em sala de aula são também apresentados por Smole, Diniz e Milani (2007) que, ao enfatizarem a necessidade de que esse tipo de proposta metodológica seja sempre bem planejado e orientado, elencam várias habilidades matemáticas que podem ser desenvolvidas a partir dos jogos, que compõem o raciocínio lógico:

O trabalho com jogos nas aulas de matemática, quando bem planejado e orientado, auxilia o desenvolvimento de habilidades como observação, análise, levantamento de hipóteses, busca de suposições, reflexão, tomada de decisão, argumentação e organização, as quais estão estreitamente relacionadas ao assim chamado *raciocínio lógico* (SMOLE, DINIZ & MILANI, 2007, p. 9).

Muitas capacidades vistas como fundamentais para a compreensão da Matemática podem ser desenvolvidas por meio dos jogos. Esse aspecto é trazido pelos Parâmetros Curriculares de Pernambuco de modo a ressaltar o potencial metodológico do jogo nas aulas de Matemática, como é possível observar na citação que segue:

Convém lembrar que a observação precisa dos dados, a identificação das regras, a procura de uma estratégia, o emprego de analogias, a redução a casos mais simples, a variação das regras, entre outras possibilidades, são capacidades que podem ser desenvolvidas quando se trabalha com jogos na aula de Matemática (PERNAMBUCO, 2012, p. 37).

Na mesma linha, GITIRANA *et al.* (2013) elencam uma série de fatores que relacionam discussões sociais propiciadas pelos jogos às questões envolvendo o funcionamento do processo de aprendizagem matemática:

[...] o contato com os jogos auxilia a integração da criança com o grupo e a negociação de regras a serem seguidas. Tudo isto propicia uma boa discussão de questões da vida em sociedade. As regras de jogos e os algoritmos matemáticos muitas vezes têm o mesmo princípio: são regras a serem seguidas no desenvolvimento de uma atividade, no primeiro caso, jogar, no segundo realizar uma tarefa matemática. Em geral, chegam a identificar diferentes situações que devem ser analisadas e caminhos diferentes que podem ser tomados (GITIRANA *et al.*, 2013, p. 13).

É importante destacar que, quando se fala do potencial dos jogos matemáticos, há muitos aspectos que superam os limites dos conhecimentos matemáticos propriamente ditos ou adjacentes. A vivência desse tipo de experiência é de fundamental importância para o desenvolvimento do indivíduo, como pode ser visto na citação a seguir:

Além das ideias matemáticas, o desenvolvimento de atitudes importantes para a aprendizagem da matemática pode ser propiciado no desenvolvimento dos jogos. A vitória numa partida ou a descoberta da solução de um desafio são experiências relevantes para fortalecer a autoconfiança, indispensável ao processo de aprendizagem (GITIRANA *et al.*, 2013, p. 12).

Os autores complementam o argumento trazido na citação anterior enfatizando a necessidade de que sempre haja uma reflexão acerca do planejamento das atividades envolvendo o uso de jogos, principalmente em relação à adequação do jogo ao público a quem está se destinando, pois, um descompasso entre a faixa etária dos estudantes e o nível dos desafios propostos pelos jogos, por exemplo, pode causar um efeito contrário ao esperado, causando frustrações e queda de autoconfiança. Segue o trecho em que os autores reforçam a necessidade desse cuidado:

É bom notar, em contrapartida, que as derrotas repetidas e os insucessos frequentes diante dos desafios podem levar a frustrações e reforçar a ideia de incapacidade para compreender os fatos na área da Matemática. Por isso, é necessária uma reflexão constante sobre a adequação do jogo à faixa etária a que se destina bem como às características de cada turma (GITIRANA *et al.*, 2013, p. 12).

Os Parâmetros Curriculares de Pernambuco também endossam essa necessidade de que as atividades envolvendo o uso de jogos matemáticos sejam planejadas adequadamente, principalmente por existirem, por um lado, conceitos matemáticos envolvidos que precisam ser abordados corretamente, e, por outro lado, a necessidade de que o aspecto lúdico seja mantido. Na citação a seguir é apresentado o trecho do documento que trata desse cuidado no equilíbrio entre a dimensão lúdica e a manutenção do rigor conceitual do que se pretende abordar por meio do jogo:

A complexidade de alguns jogos, mesmo aqueles mais comuns, requer, de um lado, clareza sobre os vários conceitos matemáticos envolvidos e, de outro, um planejamento do momento e da maneira adequados para a sua utilização no processo de ensino e aprendizagem, para que seja garantida a riqueza conceitual, o prazer em participar da atividade e a conquista da autoconfiança (PERNAMBUCO, 2012, p. 38).

Finalizamos esta seção enfatizando que não é a presença do jogo matemático, por si só, na sala de aula que permitirá o alcance de todas as vantagens apresentadas ao longo desse texto. A utilização de um jogo matemático, assim como de qualquer material didático, exige um planejamento cuidadoso e uma orientação específica.

### 3.2.2 O Mankala Colhe Três

Como já foi mencionado anteriormente, o jogo Mankala Colhe Três foi criado entre os anos de 2010 e 2011 no âmbito do projeto “Formação docente: interdisciplinaridade e ação docente – Projeto Rede” (GITIRANA *et al.*, 2013). Esse projeto foi desenvolvido pela Universidade Federal de Pernambuco – UFPE em parceria com o Ministério da Educação – MEC, a Secretaria de Educação Básica, a União Nacional dos Dirigentes Municipais de Educação – UNDIME e as secretarias Estadual e Municipais de Educação de Pernambuco.

O Projeto Rede teve como objetivos:

- Promover a formação continuada de professores da educação infantil, do ensino fundamental e do ensino médio, incluindo a elaboração de material didático.
- Promover a formação de formadores para os programas e cursos de formação continuada.
- Conceber, elaborar e desenvolver material didático, novas metodologias e tecnologias de ensino para os processos ensino-aprendizagem (PROJETO REDE, 2011, p. 2).

Os objetivos apresentados acima foram levantados considerando distintas áreas de conhecimento que faziam parte do projeto (Alfabetização, História e Matemática). Para focar em cada área de conhecimento separadamente, foram criados subprojetos, contendo objetivos específicos do Projeto Rede como um todo. O *subprojeto 3*, intitulado “Jogos no ensino de matemática a partir de sucata” foi criado para buscar alcançar o objetivo específico “Promover a formação continuada de professores na área de matemática e elaborar material didático (jogos e materiais concretos)” (PROJETO REDE, 2011, p. 3).

O *subprojeto 3* teve alguns princípios norteadores, como o *reaproveitamento* (uso de materiais de sucata); o *jogo para todos*, de incentivo à confecção dos jogos pelos alunos, pelo professor, por familiares, pela comunidade etc., ampliando os espaços de uso do jogo; e a

*confeção do jogo como atividade de modelagem*, permitindo uma abordagem matemática também no processo de construção dos jogos (PROJETO REDE, 2011).

O Mankala Colhe Três é fruto do *subprojeto 3* do Projeto Rede, que, além do Mankala Colhe Três, permitiu a criação de outros sete jogos: Jogo da Velha com Figuras Geométricas; Jogo dos Polígonos; Jogo do Nim com Dados; Jogo dos Sinais; Bingo dos Números Racionais; Desafio das Operações; e Bingo das Grandezas e Medidas.

O Mankala Colhe Três é composto por um tabuleiro com cinco covas dispostas circularmente e 25 sementes ou objetos similares (pedras, pedaços de canudo, emborrachado etc.). Segue uma imagem com vários tabuleiros de Mankala Colhe Três produzidos no âmbito do Projeto Rede a partir de sucata e materiais de baixo custo:

Figura 5 – Imagem de tabuleiros de Mankala Colhe Três confeccionados com sucata e materiais de baixo custo



Fonte: adaptado de Andrade *et al.* (2011).

O formato e as regras do Mankala Colhe Três foram pensados de modo a permitir que estudantes dos anos iniciais do Ensino Fundamental trabalhassem múltiplos e divisores de um número. Desse modo, contando com 25 sementes e cinco covas, dispostas circularmente, e um número de participantes entre dois e quatro, o objetivo do jogo é colher o maior número de sementes, considerando as seguintes regras:

- O jogo inicia com cinco sementes por cova;
- Os próprios jogadores decidem quem será o primeiro a jogar e a sequência dos outros jogadores;

- Em sua vez, cada um dos jogadores escolhe uma das covas para retirar todas as suas sementes, dividir o conjunto de sementes em partes iguais e redistribuir as partes entre as covas consecutivas, seguindo o sentido horário. Vale destacar que é permitido passar todas as sementes para a cova seguinte, ao considerar que o número de partes é um, assim como há a possibilidade de que algumas covas recebam sementes mais de uma vez em uma mesma jogada, caso o número de partes seja maior que cinco;
- Sempre que restarem exatamente três sementes na última cova da redistribuição de uma jogada, o jogador deverá colhê-las;
- O jogo termina quando restarem apenas quatro sementes no tabuleiro ou quando ocorrerem cinco rodadas consecutivas sem colheita. Vence o jogo quem colher o maior número de sementes.

Como já abordamos na seção referente à Especificação, os criadores do jogo apresentaram uma série de finalidades educacionais suscetíveis de serem exploradas por meio do uso do Mankala Colhe Três, que, em termos da epistemologia do regime didático do saber (CHEVALLARD, 1991), podemos caracterizar como noções matemáticas, protomatemáticas ou paramatemáticas.

Ao considerar a divisibilidade entre números naturais como a noção matemática em foco nas situações didáticas envolvendo o Mankala Colhe Três, podemos situar brevemente tal noção no contexto da Teoria dos Números.

Segundo Domingues (1991),

Diz-se que um número natural  $a$  divide um número natural  $b$  se  $b = ac$  para algum  $c \in \mathbb{N}$ . Neste caso, diz-se também que  $a$  é divisor de  $b$  e que  $b$  é múltiplo de  $a$ . Ou ainda que  $b$  é divisível por  $a$ . Indicaremos por  $a \mid b$  o fato de  $a$  dividir  $b$ ; e se  $a$  não divide  $b$ , escrevemos  $a \nmid b$  (DOMINGUES, 1991, p. 31).

Ao comentar a definição de divisibilidade, Paiva *et al.* (2016) enfatizam que  $a$  e  $b$  podem ser quaisquer números, já que a única restrição é o fato de que, se  $a$  for nulo, então  $b$  também deve ser, o que recai no fato de  $c$  não ser único. Dessa forma, “uma das propriedades de divisibilidade abordada em Teoria dos Números é que  $n \mid n$ , para todo  $n$  inteiro, ou seja, para o caso de  $n = 0$ , tem-se que  $0 \mid 0$ , pois  $0 = 0 \cdot c$ , para todo  $c$  inteiro” (PAIVA *et al.*, 2016).

É importante destacar que, como pôde ser visto na citação anterior, a definição de divisibilidade também pode ser aplicada para os números inteiros. Na verdade, de maneira mais ampla, essa definição é válida para qualquer anel, que definiremos formalmente na citação que segue:

Sejam  $A$  um conjunto e  $(+)$  e  $(-)$  duas operações em  $A$ , chamadas de adição e multiplicação. A terna  $(A, +, \cdot)$  será chamada de *anel* se as operações gozarem das seguintes propriedades.

**A<sub>1</sub> (A adição é associativa)** *Quaisquer que sejam  $a, b, c \in A$ , tem-se que  $(a + b) + c = a + (b + c)$ .*

**A<sub>2</sub> (A adição é comutativa)** *Quaisquer que sejam  $a, b \in A$ , tem-se que  $a + b = b + a$ .*

**A<sub>3</sub> (Existe um elemento neutro para a adição)** *Existe  $\alpha \in A$  tal que  $\alpha + x = x$ , para todo  $x \in A$ .*

**A<sub>4</sub> (Todo elemento de A possui um simétrico)** *Para todo  $a \in A$ , existe  $a' \in A$  tal que  $a + a' = \alpha$ .*

**M<sub>1</sub> (A multiplicação é associativa)** *Quaisquer que sejam  $a, b, c \in A$ , tem-se que  $(a \cdot b) \cdot c = a \cdot (b \cdot c)$ .*

**M<sub>2</sub> (A multiplicação é comutativa)** *Quaisquer que sejam  $a, b \in A$ , tem-se que  $a \cdot b = b \cdot a$ .*

**M<sub>3</sub> (Existe um elemento neutro para a multiplicação)** *Existe  $e \in A$ , com  $e \neq 0$ , tal que  $x \cdot e = x$  para todo  $x \in A$ .*

**AM (A multiplicação é distributiva com relação à adição)** *Quaisquer que sejam  $a, b, c \in A$ , tem-se que  $a \cdot (b + c) = a \cdot b + a \cdot c$ . (HEFEZ, 1993, p. 23-24).*

Quanto às definições de números primos e compostos, que serão bastante abordadas ao longo da discussão sobre as finalidades educacionais do Mankala Colhe Três, temos que

Um número  $p \in \mathbb{N}$  se diz *primo* se (i)  $p \neq 0$  e  $p \neq 1$ ; (ii) Os únicos divisores de  $p$  são 1 e  $p$ . Um número  $a \in \mathbb{N}$ ,  $a \neq 0$  e  $a \neq 1$ , é chamado *composto* se  $a$  não é primo. Assim, um número composto sempre pode ser fatorado num produto  $a = bc$ , onde  $b \neq 1$  e  $c \neq 1$  (DOMINGUES, 1991, p. 52).

Nos parágrafos a seguir, apresentaremos cada uma das finalidades educacionais apresentadas pelos criadores do Mankala Colhe Três, acompanhada de uma breve justificativa.

A primeira delas é a de *desenvolver estratégias de quantificar mentalmente*. Apoiando-se em pesquisadores como Kamii (1997), Nunes (1997) e Panizza (2006), que discutem a evolução da habilidade da criança em quantificar no processo de aquisição do número, os criadores do jogo afirmam que o ato de contar objetos dispostos desordenadamente é uma habilidade mais complexa e que, ao jogar o Mankala Colhe Três, essa habilidade é demandada logo o início do jogo, quando é preciso saber quantas sementes há em cada cova.

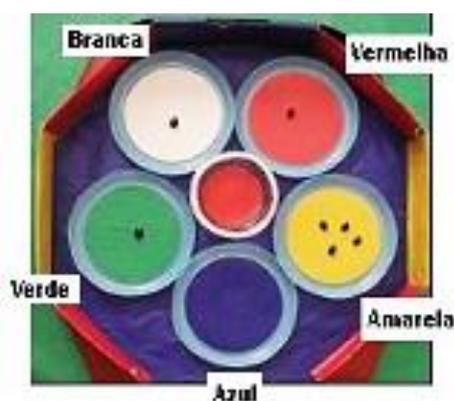
Segundo os autores do jogo, para realizar essa tarefa, os estudantes desenvolvem a capacidade de contar objetos dispostos desordenadamente de maneira mais rápida e visual, utilizando de esquemas de agrupamento, por exemplo (ANDRADE *et al.*, 2013).

Outra finalidade educacional do Mankala Colhe Três, apontada pelos criadores do jogo, é *resolver problemas com situações mistas: aditivas e multiplicativas*. Essa finalidade educacional apoia-se na Teoria dos Campos Conceituais, do teórico francês Gérard Vergnaud

(1991), que aponta a necessidade de lidar com diferentes situações que deem significado aos conceitos e procedimentos.

De fato, ao buscar desenvolver uma estratégia para realizar uma jogada no Mankala Colhe Três, o jogador é levado a articular situações com diferentes significados para a adição, multiplicação, subtração e divisão. Esse fato é ilustrado pelos criadores do jogo por meio do seguinte exemplo, em que o jogador, diante da configuração do tabuleiro apresentada, decide realizar sua jogada a partir da cova amarela:

Figura 6 – Configuração de uma jogada de Mankala Colhe Três



Fonte: Andrade *et al.* (2013, p. 30).

Segue o trecho em que os criadores do jogo analisam o exemplo acima para confirmar a articulação das situações com diferentes significados para as operações:

Um raciocínio vitorioso pode ser o seguinte: “*Faltam duas sementes para colher na cova verde. Se distribuir de dois em dois, eu consigo!*”. Com isso, o jogador desenvolve uma estratégia de jogada a partir da resolução da situação “quantas sementes faltam para eu completar as três?”, “em quantas partes devo distribuir para conseguir essa quantia na última cova?” e “quantas covas são da cova inicial até a pretendida?”. A primeira, sendo uma situação aditiva que resolvida por subtração; a segunda, uma situação multiplicativa de cota, e a terceira, outra situação aditiva. O sucesso da jogada vai depender da estratégia gerada a partir dessa correlação. (ANDRADE *et al.*, 2013, p. 30).

A finalidade educacional de *dividir por cálculo mental* é considerada devido ao fato de que, durante a realização das partidas de Mankala Colhe Três, a todo tempo o participante é conduzido a realizar divisões por cálculo mental. Com efeito, a cada jogada, o número de sementes da cova em que se escolhe jogar representa o dividendo, o número de partes que essa quantidade de sementes foi dividida, para posterior distribuição, representa o divisor e,

representando o quociente da divisão, o número de sementes que serão colocadas em cada cova (ANDRADE *et al.*, 2013).

Mais uma finalidade educacional do Mankala Colhe Três é a de *mapear as possibilidades*. O jogo é composto de muitas possibilidades de jogadas, que são regidas pelas escolhas do jogador.

Para que uma jogada seja realizada estrategicamente diante de um cenário em que existe a possibilidade de colher sementes, por exemplo, é necessário inicialmente saber de qual cova devem ser retiradas as sementes e em quantas partes essas sementes serão distribuídas, para que, a partir de então, se possa colher. No entanto, neste mesmo cenário, se houver mais de um tipo de jogada em que é possível colher sementes, uma jogada estratégica seria a de escolher a que não deixaria o adversário em uma situação de possível colheita.

É possível perceber que, mesmo quando se trata de apenas um cenário, são muitas as possibilidades de jogadas a cada rodada. Deste modo, o sucesso do jogador depende da eficiência do seu mapeamento das possibilidades.

O *reconhecimento de divisores de um determinado número* é mais uma finalidade educacional do Mankala Colhe Três. Esse reconhecimento é mobilizado implicitamente sempre que o jogador precisar decidir como irá dividir a quantidade de sementes removidas de uma cova para redistribuí-las.

Como exemplo da finalidade citada acima, se uma cova com 6 sementes for escolhida para a jogada, o jogador perceberá que há quatro tipos de jogadas possíveis: colocar todas as sementes na cova seguinte, que representaria o 1 como divisor do 6; distribuir dois grupos de três sementes, representando o 2 como divisor de 6; distribuir três grupos de duas sementes, representando o 3 como divisor do 6; e distribuir seis sementes uma a uma, representando o 6 como divisor do 6.

Outra finalidade, que também é mobilizada por meio da exploração das possibilidades de distribuição em partes iguais a partir de quantias existentes nas covas, é a de *decidir se um número é múltiplo de outro*. Isso se dá, por exemplo, quando um jogador repara que há uma cova com apenas uma semente e que pode ser uma candidata a ter suas sementes colhidas naquela jogada. Para isso, ele pode tentar analisar se, em algumas das outras covas, há uma quantidade múltipla de 2, para que, ao distribuir suas sementes de duas em duas, consiga ter a chance de colher sementes ao fim da redistribuição.

A última finalidade educacional apresentada pelos criadores do Mankala Colhe Três é a de *reconhecer números primos*. Esse reconhecimento tende a ser realizado rapidamente pelos jogadores, tendo em vista que sempre que uma cova com um número primo de sementes é

escolhida para ser jogada, o número de possibilidades de distribuição se restringe a dois: passar todas as sementes para a cova seguinte ou distribuir as sementes uma a uma, já que um número natural primo é aquele que possui apenas ele mesmo e o 1 como divisores.

É importante destacar que esse tipo de reconhecimento, assim como muitos outros citados nessa seção, é feito pelos estudantes a partir de certa experiência com o jogo, pois, inicialmente, não parece haver tanta distinção entre as escolhas das covas e das maneiras de distribuir as sementes, mas após a realização de várias partidas, as estratégias dos alunos tendem a ir aprimorando e, implicitamente, esses “teoremas” levantados vão sendo verificados.

A inspiração para a criação do Mankala Colhe Três veio do jogo *Mankala Ouri*, que, por sua vez, representa um dos diversos tipos de jogos denominados *mankala*. O termo *mankala*, também encontrado na literatura como *mancala*, é uma palavra de origem árabe que significa “mover”, “transferir”. É por isso que os jogos que trazem o nome *mankala* possuem como característica geral a distribuição de sementes, pedras ou objetos similares em buracos ou covas cavados no chão ou feitos em tabuleiros.

Esses jogos são mundialmente difundidos, existindo mais de 200 tipos diferentes de jogos com essas características espalhados pelo mundo (SANTOS, 2008a). É possível observar na imagem a seguir diferentes tipos de *mankalas* sendo jogados:

Figura 7 – Fotos de diferentes *mankalas* sendo jogados



Fonte: adaptado de Sosio (2022).

Não se sabe ao certo a origem exata dos jogos do tipo mankala. Estima-se que esses jogos possuem uma idade que passa dos 7000 anos e que tenham se originado em países do continente africano (SANTOS, 2008a; SANTOS, 2008b; SOMARIVA, 2011; CÂMARA, 2012). Segue uma citação que traz um pouco desse aspecto da origem dos mankalas:

Alguns consideram que os jogos da família mancala são os mais antigos do mundo, talvez na origem da própria civilização. Existem registros que indicam que a provável origem desses jogos tenha se dado no Egito e, a partir do Vale do Nilo, eles teriam se expandido progressivamente para o restante do continente africano e para o Oriente (SANTOS, 2008b, p. 14).

Os jogos mankala são repletos de aspectos culturais e filosóficos enraizados. Santos (2008a), por exemplo, ressalta que, ao simular situações de semeadura e de colheita, os jogadores são levados a colher sementes mesmo que em terras adversárias, revelando que, mesmo que joguem como oponentes, não é desejada a eliminação do adversário. Esse fato representa que a produção precisa ter como base a apropriação coletiva do solo e a transmissão desse solo às próximas gerações do mesmo modo que receberam.

Ao tratar das raízes culturais trazidas pelos jogos do tipo mankala, Santos (2008a) elenca uma série de aspectos que, a depender do povo, poderiam ser considerados. São eles:

- O jogo poderia ser associado ao movimento das estrelas ou a rituais sagrados;
- O uso do jogo poderia ter como objetivo a fartura na colheita, o que fazia com que só se jogasse durante o dia, pois, à noite, o jogo era deixado para que os deuses pudessem jogar para abençoar as plantações;
- O jogo poderia ser utilizado devido à morte de um integrante da comunidade. Neste caso, jogava-se durante o velório para que os maus espíritos se distraíssem enquanto o espírito do morto ia para um bom lugar;
- O mankala poderia ser jogado para decidir um novo líder, por meio de um campeonato em que todos os candidatos ao cargo de liderar a comunidade deveriam participar.

Muitos pesquisadores da área de Educação Matemática utilizam os jogos do tipo mankala para propor uma articulação com a *Etnomatemática*. Dentre as principais pesquisas que propõem essa articulação, podemos destacar Santos (2008a), Santos (2008b) e Câmara (2012). A Etnomatemática é um programa proposto pelo pesquisador em educação Matemática Ubiratan D'Ambrosio, que pode ser compreendido por meio da seguinte citação:

Indivíduos e povos têm, ao longo de suas existências e ao longo da história, criado e desenvolvido instrumentos de reflexão, de observação, instrumentos teóricos e, associados a esses, técnicas, habilidades (artes, técnicas, techné, ticas) para explicar, entender, conhecer, aprender, para saber e fazer como resposta a necessidades de sobrevivência e de transcendência (matema), em ambientes naturais, sociais e culturais (etno) os mais diversos. Daí chamarmos o exposto acima de Programa Etnomatemática. O nome sugere o corpus de conhecimento reconhecido academicamente como Matemática (D'AMBROSIO, 2005, p. 112).

Ao considerar a Etnomatemática como o conjunto de formas de matemática que são próprias de grupos culturais, ela possui duas dimensões que se articulam: introduzir a variável “cultura” no ensino e na aprendizagem de matemática; e questionar a matemática escolar.

As pesquisas que propõem a articulação entre os jogos do tipo mankala e a Etnomatemática trazem, de modo geral, a temática do resgate da cultura africana às escolas brasileiras (LOPES, 2008; SANTOS, 2008a; SANTOS, 2008b; SANTOS e CUNHA, 2010; SOMARIVA, 2011).

Partindo da grande influência que existe do povo africano para a cultura brasileira, os jogos do tipo mankala apresentam um papel educativo fundamental para a promoção da cultura africana. Esse debate foi colocado em evidência principalmente após a criação das leis 10.639/03 (BRASIL, 2003) e 11.645/08 (BRASIL, 2008), que propõem a inclusão no currículo oficial dos estabelecimentos de Ensino Fundamental e Médio das redes pública e privada a obrigatoriedade do estudo das histórias e das culturas africana, afro-brasileira e indígena.

### **3.2.3 A Teoria das Situações Didáticas e a análise do potencial educativo do Mankala Colhe Três**

Nesta seção, apresentaremos os principais resultados da pesquisa de Santos (2014), que buscou dar continuidade às investigações acerca do Mankala Colhe Três por meio de um trabalho de mestrado intitulado “Mankala Colhe Três: jogando e explorando conhecimentos matemáticos por meio de situações didáticas”.

Nossa intenção na realização dessa síntese da pesquisa de Santos (2014) é analisar e incorporar em nossa pesquisa todas as contribuições acerca do potencial educativo do jogo investigadas pelo autor.

Considerando que, ao longo do desenvolvimento do percurso metodológico de sua pesquisa, Santos (2014) investigou a fundo a modelagem das situações de utilização do jogo, de modo a favorecer a vivência de situações didáticas, traremos um panorama geral de sua pesquisa ao mesmo tempo em que apresentaremos cada um dos resultados obtidos pelo autor.

Em sua pesquisa, Santos (2014) utilizou elementos da Teoria das Situações Didáticas (BROUSSEAU, 2008), propostas pelo teórico francês Guy Brousseau, como fundamentação teórica. Tendo em vista que a Teoria das Situações Didáticas também será considerada na presente pesquisa, traremos, a seguir, uma subseção sobre a teoria, com enfoque nos principais elementos a serem abordados.

### 3.2.3.1 Elementos da Teoria das Situações Didáticas

Iniciaremos nossa explanação da Teoria das Situações Didáticas (TSD) por meio da apresentação dos conceitos de *situação* e de *meio*. Segundo Brousseau (1997), uma situação pode ser definida como o modelo de interação entre um sujeito e um meio “determinado”. O meio, por sua vez, considerado como subsistema autônomo antagônico ao sujeito, é preparado para desafiar o aluno a estimulá-lo a obter respostas a uma dada situação-problema.

Segue uma citação que traz um pouco mais detalhadamente os conceitos apresentados:

Consideremos um dispositivo criado por alguém que queira ensinar um conhecimento ou controlar sua aquisição. Esse dispositivo abrange um *meio* material – as peças de um jogo, um desafio, um problema, inclusive um exercício, fichas etc. – e as regras de interação com esse dispositivo, ou seja, o jogo propriamente dito. Contudo somente o funcionamento e o real desenvolvimento do dispositivo, as partidas de fato jogadas, a resolução do problema etc. podem produzir um efeito de ensino. Portanto, deve-se incluir o estudo da evolução da situação, visto pressupormos que a aprendizagem é alcançada pela adaptação do sujeito, que assimila o meio criado por essa situação, independentemente de qualquer intervenção do professor ao longo do processo (BROUSSEAU, 2008, p. 22).

Quando Brousseau (2008) ressalta que, para que seja produzido um efeito de ensino, é preciso que o sujeito se adapte, assimilando o *meio* criado para a situação de forma independente de qualquer intervenção do professor durante o processo, ele enfatiza que é preciso que existam momentos em que o aluno não perceba a intenção didática do professor, para que ele consiga aprender de fato. Esse aspecto é mais um elemento crucial na TSD, que é o conceito de *adidático*.

Para Brousseau (1997), o caráter *adidático* das situações se caracteriza por considerar o funcionamento normal dos conhecimentos, fora das condições didáticas, que seriam aquelas em que alguém decidiu pelo aluno que saber ele deveria aprender. Uma distinção importante deve ser feita entre os conceitos de *adidático* e “não didático”, já que em um meio *adidático* há a intenção, por parte do professor, de provocar a aprendizagem.

Mais um conceito fundamental da Teoria das Situações Didáticas é o de *Contrato Didático*, que pode ser compreendido como um sistema de responsabilidades recíprocas entre o professor e o aluno em uma situação didática. É importante destacar que o contrato didático se dá predominantemente de forma implícita. Segue uma citação em que o teórico aborda o conceito de contrato didático:

O contrato didático é a regra do jogo e a estratégia da situação didática. É o meio que o professor tem de a colocar em cena. Mas a evolução da situação modifica o contrato, que permite então a obtenção de situações novas. Da mesma maneira, o conhecimento é aquilo que se exprime através das regras da situação a-didática e através das estratégias. A evolução destas estratégias exige produções de conhecimentos que, por sua vez, permitem a concepção de novas situações a-didáticas (BROUSSEAU, 1996, p. 50).

Na citação acima, Brousseau (1996) comenta sobre a necessidade de ruptura do contrato didático nas situações didáticas, sendo, inclusive, o que leva o aluno a evolução dos conhecimentos. Na citação a seguir, Brousseau (1996) explica mais detalhadamente como ocorre essas rupturas de contrato e como elas são percebidas, tanto do ponto de vista do aluno quanto também pelo olhar do professor:

Em particular, as cláusulas de ruptura e o enquadramento do contrato não podem ser descritos antecipadamente. O conhecimento será precisamente aquilo que resolverá as crises resultantes dessas rupturas, que não podem ser pré-definidas. Contudo, no momento em que se dão, tudo se passa como se um contrato implícito ligasse o professor ao aluno: surpresa do aluno que não sabe resolver o problema e que insurge contra o facto de o professor não tê-lo sabido torna-lo capaz de o fazer, surpresa do professor que considerava razoavelmente suficientes as suas prestações... revolta, negociação, procura de um novo contrato, que depende do novo “estado”, dos saberes... adquiridos e visados.

O conceito teórico em didática não é, pois, o contrato (o contrato adequado, desadequado, verdadeiro ou falso), mas o *processo de busca de um contrato* hipotético (BROUSSEAU, 1996, p. 52-53).

A noção de contrato didático evidencia a existência de expectativas, geralmente não explícitas, dos alunos em relação ao professor e vice-versa, com respeito aos conhecimentos em jogo em uma situação didática. A ruptura desse contrato, que rege as relações entre professor e aluno e determina certa divisão de responsabilidades entre eles, acontece a partir do momento em que um dos parceiros age em desacordo com o que se espera dele. Essa dinâmica dos saberes veiculados na sala de aula faz com que a evolução dos conhecimentos modifique a divisão de responsabilidades e as expectativas recíprocas entre professor e alunos. O foco central não é avaliar se um contrato é “bom” ou “ruim”, mas evidenciar o processo de busca contínua de negociação de contratos hipotéticos condizentes com o estado de conhecimento.

Outro importante conceito da Teoria das Situações Didáticas é o de *devolução*. Para apresentar esse conceito, Brousseau parte de duas declarações:

- 1 O ensino tem por objetivo principal o funcionamento do conhecimento como produção livre do aluno em suas relações com um meio *adidático*.
- 2 O aluno adquire conhecimentos por meio de diversas formas de adaptação às restrições de seu entorno.

Partindo das declarações apresentadas acima, Brousseau (2008) conceitua o processo de devolução da seguinte maneira: “a *devolução* é o ato pelo qual o professor faz com que o aluno aceite a responsabilidade de uma situação de aprendizagem (*adidática*) ou de um problema e assumo ele mesmo as consequências dessa transferência” (BROUSSEAU, 2008, p. 91).

A partir da citação anterior, podemos compreender que, para que o processo de devolução seja efetivado de fato, é necessário que esse processo possibilite a presença do caráter adidático da situação de aprendizagem, permitindo, assim, que o aluno seja o responsável pela situação.

Ainda sobre o processo de devolução, segue mais uma citação do teórico, que complementa o que foi dito acima, enfatizando o importante papel do professor nesse processo:

Sabemos que o único meio de “fazer” matemática é procurar e resolver determinados problemas específicos e, a este propósito, colocar novas questões. O professor tem, pois, de efectuar, não a comunicação de um conhecimento, mas a devolução do problema adequado. Se esta devolução se opera, o aluno entra no jogo e, se ele acaba por ganhar, a aprendizagem teve lugar (BROUSSEAU, 1996, p. 51).

Na TSD, as interações, que são as relações entre um sujeito e o *meio*, são classificadas em, ao menos, três categorias: trocas de informação não codificadas ou sem linguagem, representadas pelas ações e as decisões que agem diretamente sobre o outro protagonista; trocas de informação codificadas numa linguagem, que seriam mensagens; e trocas de juízo ou opinião, caracterizadas por sentenças referentes a um conjunto de enunciados que exercem o papel de teoria.

O teórico enfatiza que, mesmo que haja uma classificação das interações, todas elas se imbricam: “Estas categorias são estritamente encaixadas umas nas outras, porque uma troca de juízo é uma troca de informações particulares, e esta é um tipo particular de ação e de decisão” (BROUSSEAU, 1996, p. 95).

Essas categorias de produções esperadas pelo aluno, por meio de sua relação com o meio, são base para a classificação dos tipos de situações didáticas propostas por Brousseau (1997): situação de *ação*; situação de *formulação*; e situação de *validação*.

Esses tipos de situações são regidos e revelados pelos próprios alunos por meio de suas *interações*. A seguir, serão apresentadas algumas particularidades que caracterizam os *esquemas* para cada um dos três tipos de situações didáticas mencionadas. Brousseau caracterizou ainda as situações de *institucionalização*, que serão discutidas mais adiante.

Em uma situação de *ação*, a interação do aluno com o *meio* se dá de maneira mais livre e as escolhas feitas inicialmente pelo aluno são quase que aleatórias. Esse caráter experimental permite que o aluno comece a perceber, por meio de suas escolhas na *interação*, certas regularidades nas retroações do *meio*, o que provoca ajustes nas escolhas subsequentes:

Se o meio reage com certa regularidade, o sujeito pode relacionar algumas informações às suas decisões (*feed-back*), antecipar suas respostas e considerá-las em suas futuras decisões. Os conhecimentos permitem produzir e mudar essas “antecipações”. A aprendizagem é o processo em que os conhecimentos são modificados (BROUSSEAU, 2008, p. 28).

Brousseau (2008) ressalta que os sujeitos revelam que estão em situações de *ação* a partir de táticas (ou procedimentos) ou declarações sobre que estão pensando. No entanto, o teórico enfatiza que as revelações do sujeito representam, na verdade, projeções do que, de fato, está sendo mobilizado por ele. Se tornando visível um modelo de ação fundamentalmente implícito (BROUSSEAU, 2008, p. 28).

Em uma situação de *formulação*, os alunos realizam interações razoavelmente mais elaboradas e possuem como característica principal a presença da linguagem, como pode ser observado no trecho a seguir:

A formulação de um conhecimento corresponderia a uma capacidade do sujeito de retomá-lo (reconhecê-lo, identificá-lo, decompô-lo e reconstruí-lo em um sistema linguístico). O *meio* que exigirá do sujeito o uso de uma formulação deve, então, envolver (efetivamente ou de maneira fictícia) um outro sujeito, a quem o primeiro deverá comunicar uma informação (BROUSSEAU, 2008, p. 29).

Na citação anterior, Brousseau (2008) mostra a necessidade, gerada pelo *meio*, do envolvimento de outro sujeito na situação de *formulação*, mesmo que esse envolvimento seja fictício, para que a comunicação – revelação da *formulação* – seja permitida.

Assim como uma situação de formulação traz uma evolução nas interações em relação às situações de ação, na situação de *validação* os alunos revelam características muito elaboradas em suas interações. Todas as informações trocadas pelos sujeitos são organizadas em forma de enunciados e teoremas. Os saberes exigidos pelo *meio* já se mostram consolidados

pelos sujeitos, que, por sua vez, já demonstram possuir domínio da situação, antecipando reações resultantes de suas escolhas. Em uma situação de *validação*, Brousseau traz que

O emissor já não é um informante, mas um proponente, e o receptor, um oponente. Pressupõe-se que possuam as mesmas informações necessárias para lidar com a questão. Colaboram na busca da verdade, ou seja, no esforço de vincular de forma segura um conhecimento a um campo de saberes já consolidados, mas entram em conflitos quando há dúvidas. Juntos encarregam-se das relações formuladas entre um meio e um conhecimento relativo a ele. Cada qual pode posicionar-se em relação a um enunciado e, havendo desacordo, pedir uma demonstração ou exigir que o outro aplique suas declarações na interação com o meio (BROUSSEAU, 2008, p.30).

Algo ressaltado pelo teórico e que ajuda a compor o que foi dito na citação anterior é o fato de que, em uma situação de validação, o aluno não mais se deixa convencer por intimidações ou autoridades, pois a validade é comunicada por meio de um sistema determinado, respeitado pelos alunos.

Inicialmente não caracterizada por Brousseau (1996), devido a ele acreditar que as situações de ação, formulação e validação já contemplassem todos os tipos de situações, a *institucionalização* recebeu o devido olhar por conta da necessidade que se percebia dos professores em se retomar um conteúdo já visto. Neste tipo de situação os alunos assumem o significado socialmente estabelecido de um saber que foi elaborado por eles mesmos, em situações de *ação, formulação e validação*.

No passado, acreditávamos que, ao considerarmos as situações de ação, formulação e validação, dispúnhamos já de todos os tipos possíveis de situação. Tínhamos situações de aprendizagem – no sentido dos psicólogos – e se poderia pensar que havíamos reduzido o ensino a sucessões de aprendizagem. [...] Demoramos a perceber que os professores realmente eram obrigados a “fazer alguma coisa”: tinham de dar conta da produção dos alunos, descrever os fatos observados e tudo que estivesse vinculado ao conhecimento em questão; conferir um *status* aos eventos da classe vistos como resultados dos alunos e do processo de ensino; determinar um objeto de ensino e identifica-lo; aproximar as produções dos conhecimentos das outras criações (culturais ou do programa) e indicar quais poderiam ser reutilizadas (BROUSSEAU, 2008, p. 31).

Essa necessidade de retomar o foco a determinados conhecimentos mobilizados nas situações de ação, formulação e validação é o que dará ao conhecimento visado o status de saber. A diferenciação entre conhecimento e saber, feita por Brousseau e Centeño (1991), será apresentada a seguir:

Os conhecimentos são meios transmissíveis (por imitação, iniciação, comunicação etc.), ainda que não necessariamente demonstráveis, de controlar uma situação e obter dela um resultado determinado, de acordo com uma expectativa e uma exigência social. O saber é o produto cultural de uma instituição que tem como objetivo

identificar, analisar e organizar os conhecimentos, a fim de facilitar a sua comunicação (BROUSSEAU & CENTEÑO, 1991, apud BROUSSEAU, 2008, p. 31-32).

Apesar de Brousseau (2008) afirmar que a ordem das situações de ação, formulação, validação e institucionalização representar um modo razoável para a construção de saberes pelos estudantes, ele esclarece que não existe uma lei geral que qualifique ou desqualifique esse processo em comparação com outros em que os saberes são primeiro reorganizados em discursos e só depois “aplicados” a situações pessoais e transformados em decisões, mesmo que um vá de encontro ao outro (BROUSSEAU, 2008, p. 33).

Brousseau (2008) utiliza o jogo “Quem diz 20?” para exemplificar o papel desempenhado pelas relações entre o funcionamento do conhecimento do aluno e as características das situações. Seguem as regras do jogo e a descrição geral da situação:

Entre dois jogadores, cada um deve chegar ao número 20 somando 1 ou 2 ao número dito pelo outro, alternadamente. O que começa diz 1 ou 2; o que continua soma 1 ou 2 a esse número. Por sua vez, o primeiro jogador acrescenta mais 1 ou 2, e assim sucessivamente. O que chegar primeiro ao número 20 ganha o jogo. [...] O professor explica a regra do jogo e começa uma partida na lousa, jogando com uma das crianças. A seguir, cede seu lugar a outro aluno (BROUSSEAU, 2008, p. 23).

Brousseau (2008) continua a proposta do jogo por meio de mais três fases. A primeira fase é caracterizada por jogos de “um contra um”, em que, após várias partidas jogadas entre si, as crianças começam a perceber que responder aleatoriamente não é a melhor estratégia e, a partir de então, começam a descobrir a vantagem de dizer 17.

Na segunda fase, a proposta da lição é que sejam realizados jogos de uma equipe contra outra, em que o professor escolhe um aluno de cada equipe para representar seu grupo em um jogo diante dos demais, que não podem intervir ao longo da partida, e, neste formato, as vitórias valem pontos para as equipes. Nesta etapa, as crianças começam a perceber a importância de discutir com os colegas de grupo a fim de definir estratégias.

Por fim, na terceira etapa, as equipes são conduzidas pelo professor a apresentar as estratégias descobertas para que tenham chegado à vitória no jogo, de modo a favorecer a demonstração da veracidade dos enunciados propostos ou criticá-los, podendo ser apresentadas provas de falsidade em algumas estratégias compartilhadas (BROUSSEAU, 2008).

É importante destacar que, assim como no jogo “Quem diz 20”, em que o objetivo da aula era “revisar a operação divisão, dando-lhe um significado diferente do aprendido em lições anteriores, de modo a favorecer, nas crianças, a descoberta e a demonstração de uma série de teoremas” (BROUSSEAU, 2008, p. 22), situações envolvendo o Mankala Colhe Três

favoreceram o levantamento de hipóteses e a descoberta de uma série de teoremas envolvendo propriedades dos números primos, dos números compostos, quantidade de divisores de um número etc. (SANTOS, 2014).

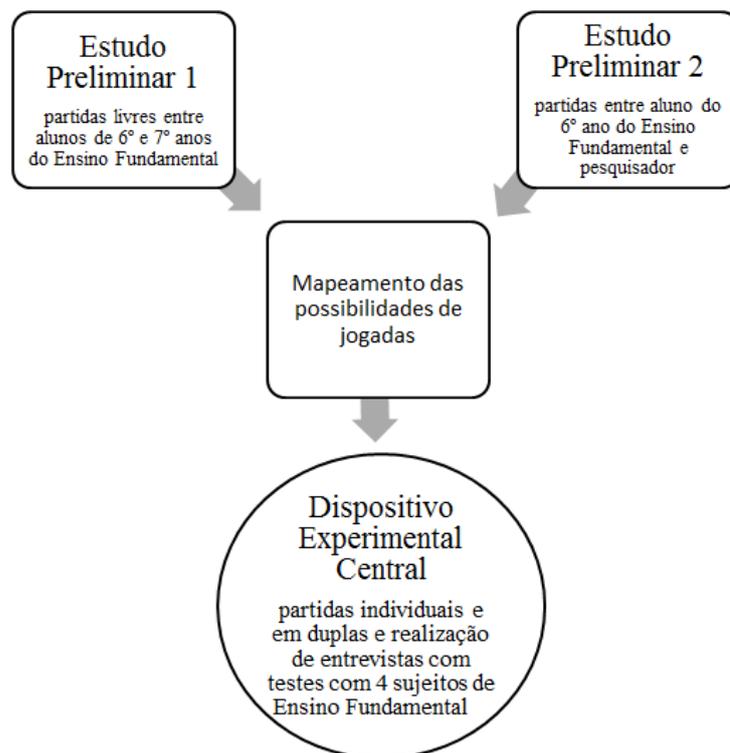
A Teoria das Situações Didáticas fundamentará todas as escolhas didáticas realizadas ao longo da concepção do Mankala Colhe Três Digital, considerando, ao seu olhar, o papel do professor, o papel dos alunos e a antecipação das situações de uso do jogo em sala de aula.

### 3.2.3.2 Análise do potencial educativo do Mankala Colhe Três

Com o intuito de investigar mais a fundo o potencial educativo do jogo Mankala Colhe Três, Santos (2014) desenvolveu uma pesquisa de mestrado cujo objetivo geral foi o de “investigar a contribuição do jogo Mankala Colhe Três para a aprendizagem de conhecimentos matemáticos por alunos de 6º ano do Ensino Fundamental” (SANTOS, 2014, p. 41).

Para buscar alcançar tal objetivo, o pesquisador realizou um percurso metodológico composto por dois Estudos Preliminares, uma etapa de mapeamento das possibilidades de jogadas e, por fim, o Dispositivo Experimental Central da pesquisa, como pode ser observado na ilustração abaixo:

Figura 8 – Esquema dos procedimentos metodológicos da pesquisa de Santos (2014)



Fonte: Santos (2014, p. 42).

No primeiro Estudo Preliminar, três estudantes de 6º e 7º anos do Ensino Fundamental realizaram duas partidas, que foram filmadas e transcritas. O jogo foi apresentado aos sujeitos por meio da descrição das regras do Mankala Colhe Três e de exemplos de possibilidades de jogadas. Ao longo da realização das partidas, o pesquisador reforçou as regras do jogo quando identificava alguma dificuldade de compreensão dos alunos ou quando eles o questionavam.

Para o segundo Estudo Preliminar, decidiu-se realizar partidas entre um estudante de 6º ano do Ensino Fundamental e o próprio pesquisador. A intenção dessa escolha foi de permitir que, durante as partidas, o pesquisador pudesse confrontar o aluno por meio de situações de jogo diversificadas. Esse momento também foi filmado e posteriormente, transcrito.

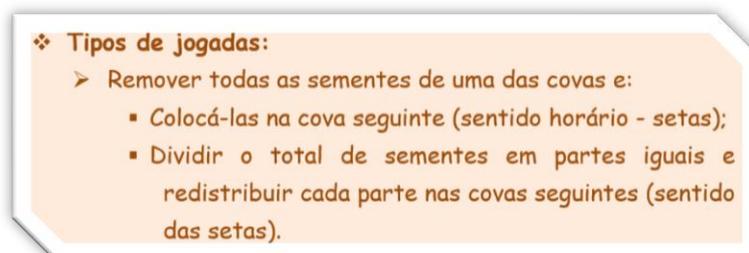
Os dois Estudos Preliminares realizaram um papel semelhante ao de pilotos na pesquisa, de modo que, a partir das análises desses dois momentos, o pesquisador se deparasse com um apanhado de informações importantes para a idealização do Dispositivo Experimental Central da pesquisa, cujo objetivo era permitir a coleta dos resultados da análise do potencial educativo do Mankala Colhe Três por meio da vivência de situações didáticas (SANTOS, 2014).

O principal resultado obtido do primeiro Estudo Preliminar foi o fato que:

os alunos encontravam certa resistência em escolher uma jogada que, associada às noções de divisores, mostrava que o número 1 pode ser considerado como divisor, o que é muito curioso, já que o número 1 é divisor de qualquer outro. Uma hipótese levantada para justificar este fato é a de que, em situações práticas, a ideia de divisor está muito ligada a um número que reparte outro em partes menores, o que não acontece no caso de o divisor ser o 1 (SANTOS, 2014, p. 47).

Uma alternativa escolhida por Santos (2014) para evitar esse tipo de problema no Dispositivo Experimental Central foi de esclarecer um pouco mais as possibilidades de jogadas, mas mantendo o jogo com as mesmas regras. Deste modo, propôs-se uma maneira de apresentar os tipos de jogadas que evitasse que o aluno pensasse que a jogada de remover todas as sementes de uma cova e as colocasse, todas, na cova seguinte não fazia parte do jogo:

Figura 9 – Tipos de jogadas no Mankala Colhe Três



Fonte: Santos (2014, p. 52).

Além desse resultado, no primeiro Estudo Preliminar também foi observado que os alunos permaneciam com algumas dúvidas sobre as regras do jogo durante a realização das partidas. Esse aspecto também foi destacado na análise do segundo Estudo Preliminar, o que fez com que Santos (2014) percebesse a necessidade de definir uma maneira precisa e padronizada de apresentar as instruções das regras do jogo no Dispositivo Experimental Central.

Segundo o autor, esse momento de instruções das regras do jogo se atrela ao processo de *devolução* da Teoria das Situações Didáticas, em que a responsabilidade da situação é aceita pelo sujeito aprendiz, de modo que este, a partir de então, assume as consequências dessa responsabilidade.

Brousseau (2008) enfatiza que, para que esse processo de devolução seja, de fato, alcançado pelos sujeitos, a vivência de situações *adidáticas* se mostra crucial. Sendo assim, Santos (2014) reforçou a necessidade de que, no Dispositivo Experimental Central, os alunos possuíssem todos os elementos possíveis para a prática e exploração do Mankala Colhe Três, sem a necessidade de intervenções do pesquisador.

Outro aspecto percebido por Santos (2014) na análise dos Estudos Preliminares foi que, em muitos momentos, o pesquisador tendia a se colocar durante as partidas para questionar os sujeitos sobre as estratégias que eles estavam utilizando. Percebeu-se, porém, que esse tipo de intervenção comprometia o caráter *adidático* das situações de jogo, sendo necessário, portanto, preservar esse momento em que os alunos exploram o jogo livremente.

Esse aspecto se atrela a mais uma questão que foi considerada e levada em conta na preparação do dispositivo experimental central:

para o pesquisador, a experimentação representa uma etapa de coleta de informações de uma pesquisa científica, para os sujeitos, esta fase representa um momento descontraído, de diversão e competição. Isso deve ser respeitado, pois se o pesquisador interfere na partida constantemente para que satisfaça suas necessidades científicas, corre o risco de comprometer a parte lúdica do jogo, influenciando, inclusive, na construção dos conhecimentos suscetíveis de serem explorados pelos alunos por meio do jogo (SANTOS, 2014, p. 52).

A etapa que precedeu a realização do Dispositivo Experimental Central, conforme ilustrado na Figura 8, caracterizada pelo mapeamento das possibilidades de jogadas do Mankala Colhe Três, foi pensada com o objetivo de favorecer a análise das jogadas utilizadas pelos sujeitos durante a realização do Dispositivo Experimental Central e para buscar novos elementos do jogo que ajudariam a elaborar a versão definitiva desse dispositivo (SANTOS, 2014).

É importante esclarecer que o que o autor chama de mapeamento do jogo consiste em elencar todas as configurações de jogo possíveis, considerando as regras originais do Mankala Colhe Três (SANTOS, 2014).

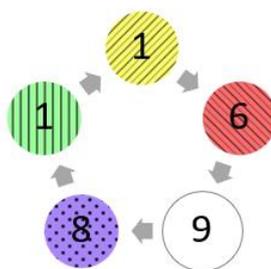
Para dar início ao mapeamento das possibilidades, Santos (2014) buscou uma estimativa da quantidade de possibilidades que poderia encontrar. Para isso, ele, inicialmente, partiu da questão: “De quantas maneiras podem ser distribuídas as sementes do *Mankala Colhe Três* nas cinco covas de seu tabuleiro?”, mesmo estando ciente de que a resposta dessa questão não forneceria, necessariamente, a quantidade exata de configurações possíveis de serem obtidas no jogo.

De fato, as regras de distribuição e colheita no Mankala Colhe Três poderiam não permitir que configurações obtidas pela simples organização das sementes no tabuleiro fossem possíveis em situações de jogo. De todo modo, Santos (2014) afirma que essa quantidade obtida forneceria um “teto” para as quantidades, de fato, possíveis no jogo.

Ao responder esse problema matemático, o autor encontrou o valor de 11238 maneiras de distribuir as sementes do Mankala Colhe Três nas cinco covas. Nesse cálculo, foram utilizadas Combinações Completas para todas as quantidades possíveis de sementes no tabuleiro do Mankala Colhe Três, que são: 25; 22; 19; 16; 13; 10; 7; e 4 (SANTOS, 2014).

A partir dessa estimativa, Santos (2014) iniciou o processo de mapeamento, que consistia em determinar todas as possibilidades de jogadas a partir de uma configuração do tabuleiro. Segue um exemplo que ilustra o método de mapeamento utilizado pelo pesquisador:

Figura 10 – Exemplo de configuração do tabuleiro do Mankala Colhe Três



Fonte: Santos (2014, p. 57).

A imagem acima representa uma simulação do tabuleiro do jogo, em que os números presentes em cada região circular representa a quantidade de sementes. Para o mapeamento das possibilidades, Santos (2014) representou essa configuração da seguinte maneira: “ $P_i = (1,1,6,9,8)$ ”, em que

“ $P_i$ ” representa a posição ou configuração do tabuleiro e o primeiro número da sequência representa a quantidade de sementes da cova verde, o segundo a quantidade de sementes da cova amarela e assim sucessivamente, até o último número, que representa a quantidade de sementes presentes na cova roxa (SANTOS, 2014, p. 57).

Santos (2014) enfatiza que o fato de o Mankala Colhe Três ser circular faz com que a expressão apresentada anteriormente represente, na verdade, uma classe de configurações do tabuleiro, que, no nosso exemplo seria: “ $(1,1,6,9,8) \equiv (1,6,9,8,1) \equiv (6,9,8,1,1) \equiv (9,8,1,1,6) \equiv (8,1,1,6,9)$ ”. Isso faz com que não seja necessário se ater as cores das covas para realizar o mapeamento do jogo, já que todas essas configurações representariam, para o jogador, situações semelhantes.

A seguir o autor apresenta todas as “ $J_{i,j}$ ” jogadas possíveis a partir da configuração “ $P_i$ ”:

$J_{i,1} = (0, 2, 6, 9, 8)^6 \leftarrow$  Remoção da semente da primeira cova para a cova seguinte.  
 $J_{i,2} = (1, 0, 7, 9, 8) \leftarrow$  Remoção da semente da segunda cova para a cova seguinte.  
 $J_{i,3} = (1, 1, 0, 15, 8) \leftarrow$  Remoção das sementes da terceira cova para a cova seguinte.  
 $J_{i,4} = (1, 1, 0, 12, 11) \leftarrow$  Remoção das sementes da terceira cova e redistribuição três a três.  
 $J_{i,5} = (3, 1, 0, 11, 10) \xrightarrow[\text{colheita}]{} (0, 1, 0, 11, 10) \leftarrow$  Remoção das sementes da terceira cova e redistribuição duas a duas (a implicação desta configuração representa a colheita de sementes, pois restaram exatamente três sementes na primeira cova, última da redistribuição).  
 $J_{i,6} = (2, 2, 1, 11, 9) \leftarrow$  Remoção das sementes da terceira cova e redistribuição uma a uma.  
 $J_{i,7} = (1, 1, 6, 0, 17) \leftarrow$  Remoção das sementes da quarta cova para a cova seguinte.  
 $J_{i,8} = (4, 4, 6, 0, 11) \leftarrow$  Remoção das sementes da quarta cova e redistribuição três a três.  
 $J_{i,9} = (3, 3, 8, 1, 10) \leftarrow$  Remoção das sementes da quarta cova e redistribuição uma a uma.  
 $J_{i,10} = (9, 1, 6, 9, 0) \leftarrow$  Remoção das sementes da quinta cova para a cova seguinte.  
 $J_{i,11} = (5, 5, 6, 9, 0) \leftarrow$  Remoção das sementes da quinta cova e redistribuição quatro a quatro.  
 $J_{i,12} = (3, 3, 8, 11, 0) \leftarrow$  Remoção das sementes da quinta cova e redistribuição duas a duas.  
 $J_{i,13} = (3, 3, 8, 10, 1) \leftarrow$  Remoção das sementes da quinta cova e redistribuição uma a uma.” (SANTOS, 2014, p. 57-58).

É importante destacar na citação anterior que a única possibilidade de colheita na configuração de tabuleiro exemplificada é representada pela jogada “ $J_{i,5}$ ”.

Santos (2014) iniciou o mapeamento das possibilidades, mas não conseguiu finalizá-lo, produzindo, portanto, apenas um mapeamento parcial das jogadas e configurações do tabuleiro do Mankala Colhe Três, possuindo 175 configurações diferentes, em que todas essas são

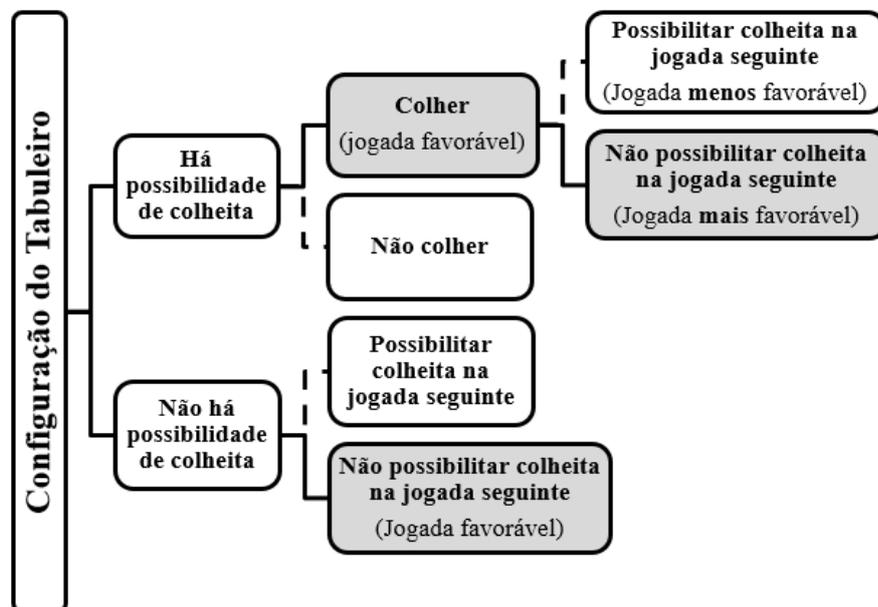
---

<sup>6</sup> O número marcado (tachado) representa a cova da qual foram removidas a(s) semente(s) para a realização da jogada. O número em negrito representa a última cova da redistribuição.

configurações do tabuleiro contendo 25 sementes. Esse mapeamento se mostrou muito útil para auxiliar na análise das jogadas feitas pelos sujeitos durante o Dispositivo Experimental Central além de trazer subsídios para a criação do teste presente na terceira etapa desse dispositivo, que conteve, em algumas atividades, situações reais de jogo para serem analisadas pelos sujeitos.

O mapeamento parcial do Mankala Colhe Três revelou a Santos (2014) que o jogo é muito aberto, tendo em vista que cada configuração do tabuleiro permite diversas possibilidades de jogadas. Deste modo, o pesquisador percebeu que as análises de jogadas a serem realizadas no experimento da pesquisa precisariam ser locais, no sentido de considerar o máximo de duas rodadas de antecipação de jogadas: a atual e, a partir desta, a do adversário. Segue um esquema proposto por Santos (2014) que traz uma maneira de hierarquizar as jogadas no Mankala Colhe Três:

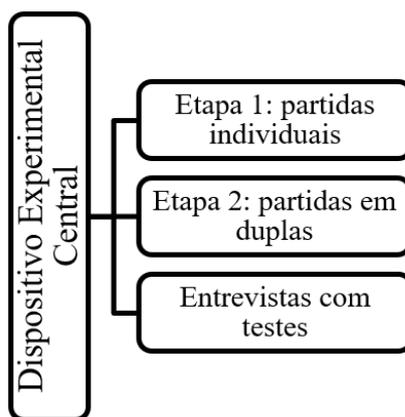
Figura 11 – Hierarquia de jogadas no Mankala Colhe Três



Fonte: Santos (2014, p. 60).

O Dispositivo Experimental Central da pesquisa de Santos (2014) foi realizado com quatro estudantes de 6º ano do Ensino Fundamental em uma escola privada da cidade do Recife-PE, sorteadas e nomeadas na pesquisa por: “A”, “B”, “C” e “D”. A estrutura do Dispositivo Experimental Central será apresentada a seguir:

Figura 12 – Etapas do Dispositivo Experimental Central da pesquisa de Santos (2014)



Fonte: Santos (2014, p. 44).

A intenção do autor em propor inicialmente partidas individuais, seguidas de partidas em duplas foi a de favorecer a vivência de situações didáticas (*ação, formulação e validação*), pois, em um primeiro momento, a ideia era que os sujeitos agissem no jogo livremente, de maneira mais exploratória. Já nas partidas em duplas, como há a necessidade de argumentação entre os membros de cada dupla, esperava-se que a decisão das escolhas a serem tomadas se apoiassem em estratégias debatidas (SANTOS, 2014).

Sobre a concepção das etapas do Dispositivo Experimental para permitirem a vivência de tipos distintos de situações didáticas, Santos afirma que:

Consideramos que na primeira etapa os sujeitos estão em processo de apropriação do funcionamento do jogo e, ao mesmo tempo, devem evoluir de jogadas arbitrárias para a elaboração de estratégias que aumentem as chances de ganhar o jogo (por meio da mobilização de conhecimentos matemáticos). Quando os alunos forem jogar em dupla, já devem ter um domínio razoável das regras do jogo e de possíveis estratégias de jogadas. Como os componentes da dupla devem se convencer mutuamente da jogada mais pertinente, haverá necessidade de construir uma argumentação. Por isso, pensamos que a explicitação das jogadas e a busca de convencimento do parceiro da dupla sobre a pertinência da estratégia proposta devem levar à vivência de situações de *formulação e validação* (SANTOS, 2014, p. 52-53).

Com o objetivo de permitir que o jogo fosse amplamente explorado pelos sujeitos participantes da pesquisa antes de propor as partidas em duplas, Santos (2014) enfatiza que as etapas 1 e 2 aconteceram em dias distintos e que, no intervalo entre as duas etapas, foram deixados na escola onde os sujeitos estudavam tabuleiros do Mankala Colhe Três, fichas de regras e imagens com distintos tabuleiros do jogo construídos com material de sucata e baixo custo.

A etapa 3 se caracterizou pela realização de entrevistas baseadas na realização de testes com situações envolvendo o Mankala Colhe Três. O objetivo dos testes foi o de complementação da coleta de informações a respeito da exploração e do desenvolvimento de determinados conhecimentos matemáticos por meio de experiências com o jogo (SANTOS, 2014).

Os testes continham quatro questões em que foram exploradas diferentes situações em torno do Mankala Colhe Três:

1. *Proposta de realização de jogadas favoráveis*, que consistia em fornecer uma configuração do tabuleiro aos estudantes e perguntar qual seria a melhor jogada a se realizar a partir da configuração;
2. *Argumentação a respeito de escolhas de jogadas feitas por sua dupla*, em que o pesquisador representava um colega de jogo do sujeito, realizava jogadas e questionava o que ele achava de cada opção de jogada realizada;
3. *Mapeamento das possibilidades de jogadas a partir de uma determinada configuração do tabuleiro*, que avaliava o poder de mapeamento de jogadas dos sujeitos, ao fornecer uma configuração específica do tabuleiro para que ele anotasse todas as possibilidades de jogadas existentes;
4. *Argumentação a respeito dos conhecimentos matemáticos que acreditam que foram explorados durante as situações de jogo*, em que era perguntado aos estudantes quais conhecimentos matemáticos eles acharam que utilizaram enquanto jogavam o Mankala Colhe Três.

Segundo o autor, todos os testes foram filmados e realizados individualmente. Além disso, todas as questões foram as mesmas para os estudantes e todas as configurações do tabuleiro presentes nas atividades dos testes foram estrategicamente escolhidas, por meio de uma análise *a priori* considerando cada escolha que os alunos poderiam tomar. As configurações de tabuleiro utilizadas nos testes foram selecionadas do mapeamento parcial das jogadas e configurações (SANTOS, 2014).

Os resultados da pesquisa de Santos (2014) foram analisados e distribuídos em cinco blocos de análise:

1. A dimensão lúdica e o desenvolvimento de atitudes do Mankala Colhe Três;
2. Conhecimentos matemáticos mobilizados pelos alunos de 6º do Ensino Fundamental por meio de situações envolvendo o Mankala Colhe Três;
3. O processo de devolução e o caráter adidático nas situações envolvendo o Mankala Colhe Três;

4. As interações e os tipos de situações didáticas envolvendo o Mankala Colhe Três;
5. A evolução das estratégias dos alunos de 6º ano do Ensino Fundamental nas partidas de Mankala Colhe Três.

As análises de cada um dos blocos foram embasadas e justificadas por meio das transcrições das partidas dos Estudos Preliminares e do Dispositivo Experimental Central e pelos resultados dos testes da etapa de Entrevistas. Em relação à dimensão lúdica do jogo e do desenvolvimento de atitudes, Santos (2014) percebeu que

as experiências com o *Mankala Colhe Três* propiciaram, de fato, momentos de ludicidade e permitiram as interações dos sujeitos, que em alguns momentos eram feitas entre oponentes e em outras entre parceiros de jogo, favorecendo, neste último, o trabalho colaborativo. Além disso, as situações de jogo permitiram a mobilização de importantes atitudes de convivência, como o fato de adversários de jogo se entrosarem ao ponto de realizarem análises de jogadas juntos (SANTOS, 2014, p. 187-188).

Em relação à identificação dos conhecimentos matemáticos mobilizados por alunos do 6º ano do Ensino Fundamental em situações envolvendo o jogo Mankala Colhe Três, Santos (2014) buscou identificar os conhecimentos que fossem compatíveis com o ano escolar dos sujeitos. Sendo assim,

foram percebidos que os sujeitos mobilizavam e reconheciam o trabalho com situações mistas (aditivas e multiplicativas) durante as partidas do jogo. Foi percebido também que, a todo instante, os sujeitos (alguns de modo mais evidente que outros) eram postos a mapear as possibilidades de jogadas a partir das configurações do tabuleiro. Esse mapeamento, inclusive, tendia a ficar cada vez mais elaborado ao passar do tempo de jogo. Foi percebido também que, durante as suas jogadas, os sujeitos reconheciam múltiplos e divisores de um número e também a característica de divisibilidade dos números primos (SANTOS, 2014, p. 188).

Em relação à análise do processo de devolução e ao caráter adidático das situações com o *Mankala Colhe Três*, Santos (2014) foi capaz de perceber que

este processo foi bem feito a ponto de fortalecer o caráter adidático das situações com o jogo. Como planejado, a maneira como o Dispositivo Experimental Central da pesquisa se deu permitiu a passagem dos alunos por situações de ação, formulação e validação, embora o aspecto da validação ficasse ainda relativamente implícito. Os conhecimentos matemáticos em foco estavam fortemente ligados à capacidade de os sujeitos realizarem jogadas favoráveis (SANTOS, 2014, p. 189).

Sobre o último bloco de análise, referente à evolução das estratégias de jogadas dos sujeitos participantes da pesquisa, Santos (2014) realizou uma análise do percurso individual das alunas ao longo de todas as etapas do Dispositivo Experimental Central. Foi percebido que o poder de mapeamento das possibilidades conduzia a realização de jogadas melhores e, deste

modo, estava diretamente atrelado à evolução das estratégias. Como resultado geral desta análise, Santos (2014) traz que

De uma maneira geral, as alunas, nomeadas por “A”, “B”, “C” e “D”, apresentaram uma evolução significativa de suas estratégias durante este dispositivo, mesmo que o tempo destinado à exploração e apropriação do jogo tenha sido razoavelmente limitado. Além disso, foi claramente percebido que a evolução das estratégias das alunas se deu de maneiras diferentes, em relação à velocidade de evolução e ao nível de estratégia alcançado. Essa análise mostrou, por exemplo, que a aluna “B” apresentou uma evolução muito rápida de suas estratégias e que também conseguiu realizar jogadas muito elaboradas durante as partidas. Já em outro extremo, percebeu-se que a aluna “D”, apesar de em alguns momentos apresentar jogadas bem elaboradas, deixava claro que não mostrava se preocupar em realizar um mapeamento satisfatório das configurações do tabuleiro nas suas jogadas, fazendo com que suas escolhas, mesmo após muita exploração do jogo, se mostrassem quase que arbitrárias (SANTOS, 2014, p. 189).

Apesar de os resultados obtidos por Santos (2014) em sua investigação tenham sido suficientes para o alcance de seus objetivos de pesquisa, em suas considerações finais, o autor elenca alguns encaminhamentos para futuras pesquisas envolvendo o Mankala Colhe Três.

Os encaminhamentos futuros de Santos (2014) foram apresentados tanto na Introdução deste trabalho, de forma resumida, quanto na fase de Especificação do Mankala Colhe Três Digital. Vale lembrar que os encaminhamentos futuros da pesquisa de Santos (2014) nos conduziram a escolher o Mankala Colhe Três como recurso para aplicação da Engenharia Didático-Informática em sua versão formalizada por Tiburcio (2020), buscando atender encaminhamentos da pesquisa de Tiburcio, no que se refere a colocar a EDI em prática e buscar desenvolver softwares educativos de tipologias diferentes de ambientes de simulação/micromundos.

### **3.2.4 A Transposição Didático-Informática do Mankala Colhe Três**

Essa seção trará algumas considerações sobre caracterizações de softwares educativos, a integração de softwares no sistema didático e, ao retomar a discussão do processo de transposição didático-informática, iniciada no capítulo desta Tese relativo à EDI, serão apresentadas algumas considerações sobre o caso do Mankala Colhe Três Digital.

Gostaríamos inicialmente de caracterizar o tipo de software que pretendemos abordar com a proposta de uma versão digital do Mankala Colhe Três. Segundo Santos (2009), há uma primeira distinção a ser realizada, entre *software educativo* e *software educativo propriamente dito*. O autor diferencia aqueles softwares que foram programados com fins educativos daqueles

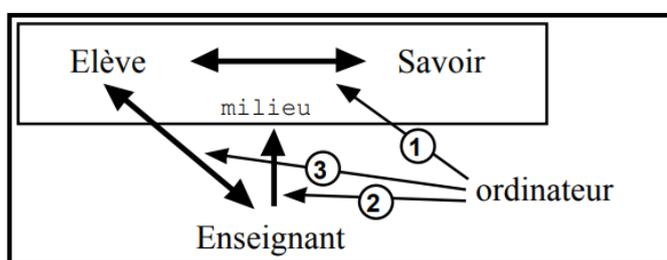
que, mesmo que seja usado com esses fins, não foram programados para isso. Segue um trecho em que o pesquisador traz essa consideração de forma detalhada:

O software educativo é, primeiramente, um espaço para proporcionar a construção de conhecimentos. Nesse sentido, qualquer software pode ser considerado educativo, como um software aplicativo (um tratamento de textos ou uma planilha de cálculos), um software lúdico (um jogo, um simulador) ou um software de autoria (uma meta-linguagem de programação). Até um software básico (como o Windows ou o DOS ou, ainda, o Linux) pode ser utilizado com finalidades educativas. Entretanto, o software educativo propriamente dito é aquele desenvolvido com a finalidades educativas explícitas demandando, para subsidiar sua produção, procedimentos específicos, relacionados a um conhecimento aprofundado dos processos cognitivos humanos, seja ele de natureza lúdica (um jogo educativo) ou de conteúdo escolar (um software para o ensino de Química, por exemplo), seja ele estático (em cd-rom) ou distribuído (para a Internet) (SANTOS, 2009, p. 21).

Em relação a esta distinção, apontada por Santos (2009), podemos inicialmente considerar que nossa proposta de software, que é o Mankala Colhe Três Digital, representaria um software educativo propriamente dito, já que ele está sendo concebido com fins educativos. No entanto, nossa abordagem vai mais além, quando buscamos integrar o computador no Sistema Didático, como investigado por Bellemain (1992), na concepção do software de Geometria Dinâmica Cabri-géomètre.

O pesquisador apresenta um modelo de integração do computador no Sistema Didático, como pode ser visto a seguir:

Figura 13 – Integração do Computador no Sistema Didático



Fonte: Bellemain (1992, p. 75).

Na ilustração acima, o número “1” se refere ao computador como elemento do meio, o número “2” ao computador na participação do meio e o número “3” ao computador participando da avaliação das aprendizagens ou da institucionalização dos conhecimentos (BELLEMAIN, 1992).

A proposta apresentada por Bellemain (1992) endossa a promoção da atividade do aluno na interação com um meio preparado para favorecer as aprendizagens, como é proposto por Brousseau (1997).

Ramos (2014), partindo na mesma linha trazida por Bellemain (1992), traz a discussão sobre as características de softwares educativos e apresenta uma classificação de tipos de softwares que podem ser considerados como aqueles que intervêm como elemento do meio. São eles: micromundos; simulação; ambientes de realidade virtual; e ambientes de programação.

Ao apresentar essa classificação, Ramos (2014) volta ao olhar para o jogo digital que ela pretende conceber, que é o Bingo dos Racionais, já discutido amplamente neste trabalho. A autora enfatiza que, pelo fato de o Bingo dos Racionais ser um jogo, ele pode fornecer um viés de condicionar o meio. Para isso, a autora traz uma abordagem sobre a noção de jogo no ensino e na aprendizagem de Matemática.

Ramos (2014) se apropria dessa noção trazida por Guy Brousseau, e retomada por Bellemain (1992), que trata a noção de jogo como

[...] uma forma de organizar o meio didático pela elaboração de situações a-didáticas "ad hoc" que favorecem a construção pelos jogadores (alunos) dos conhecimentos que justificam as estratégias vencedoras, é importante destacar o caráter contextualizado desses conhecimentos. A descontextualização, e a recontextualização em novas situações, dos conhecimentos que emergem de atividades em um jogo devem ser consideradas (RAMOS, 2014, p. 42).

Na citação acima Ramos (2014) apresenta a necessidade de que os conhecimentos construídos pelos sujeitos em situações de jogo precisam ser descontextualizados (das situações de jogo) para que possam ser recontextualizados em outras situações. Esse aspecto se relaciona com o tipo de situação didática *institucionalização*, que já teve a oportunidade de ser discutida no presente texto. De fato, como Brousseau (2008) argumenta,

Tal como os *teoremas-em-ato* desapareceriam rapidamente diante da ausência de uma formulação e uma comprovação, os conhecimentos particulares, e até mesmo os públicos, continuariam contextualizados e tenderiam a desaparecer na maré das lembranças cotidianas, caso não fossem recolocadas em um repertório especial, cuja importância e uso não foram confirmados pela cultura e pela sociedade (BROUSSEAU, 2008, p. 32).

Ao propor a concepção de uma versão digital do Mankala Colhe Três, por meio da Engenharia Didático-Informática, precisamos estar cientes de que o software que pretendemos produzir não se limitará a uma simples transposição do jogo físico do Mankala Colhe Três, pois,

neste caso, estaríamos perdendo a oportunidade de inserir no jogo todas as contribuições específicas do contexto computacional. Da mesma forma, não faria sentido propor a criação de um jogo digital que não considerasse todo aporte relacionado aos estudos sobre ensino e aprendizagem.

Nessa perspectiva, precisaremos realizar uma análise detalhada dos aspectos do que chamaremos de Transposição Didático-Informática, também chamada por “transposição informática” ou “transposição didática informática” em alguns textos científicos. Justificamos nossa escolha de nomenclatura para evitar a interpretação de que a Transposição didático-informática representaria um tipo de transposição didática, ao invés de tratá-la como um novo conceito, que traz o olhar para transformações no saber ocasionados pela presença das particularidades do ente computacional.

Para que possamos compreender um pouco melhor a transposição didático-informática, é importante fazer a distinção entre *universo interno*, *interface* e *universo externo*. O *universo interno* compõe a parte da programação de um computador, onde são definidas as operações a serem realizadas pela máquina; a *interface* é a parte responsável pela comunicação entre o computador e o usuário; o *universo externo* é o meio em que o usuário se situa, podendo ter acesso a outras formas, como pesquisar o conteúdo apresentado (BALACHEFF, 1994).

Segundo Balacheff (1994), a transposição informática é identificada como o processo de transformação que ocorre entre o *universo interno* e o *universo externo*. Almouloud (2007) ressalta que a *interface* é tida como o meio da visualização e retroações que comportamentos reveladores das propriedades do *universo interno* sejam vistos.

Em relação ao jogo que pretendemos conceber, sabemos que particularidades referentes a cada um dos meios precisam ser consideradas para que possamos idealizar um recurso que seja capaz de aprimorar o potencial educativo que a versão física do Mankala Colhe Três já continha.

Como pôde ser observado na seção que trata das origens do jogo Mankala Colhe Três e das pesquisas que analisaram o potencial educativo desse jogo, muitos são os aspectos educativos propiciados por situações envolvendo esse recurso em sua versão física, que envolve distintas naturezas (didáticas, pedagógicas, sociais, culturais etc.). No entanto, ao considerar uma versão digital do jogo, muitos aspectos precisam ser repensados, de modo a garantir o equilíbrio entre as dimensões lúdica e educativa, por exemplo, entre outros aspectos.

Tomemos como exemplo a principal ação realizada durante as partidas de Mankala Colhe Três: a manipulação de sementes. Ao propor uma versão digital do jogo, é sabido de imediato que essa ação não será mais realizada, ao menos não da mesma forma. E sabe-se

também da importância da manipulação de materiais concretos nos processos de aquisição e mobilização de noções matemáticas. Sendo assim, é preciso que seja analisada cada modificação que o jogo poderá sofrer e quais rebatimentos isso pode trazer para o potencial lúdico e educativo do jogo.

Podemos trazer mais um exemplo: sabemos que algumas capacidades mobilizadas pelo jogo se relacionam com o reconhecimento de múltiplos e divisores de um número. Esse reconhecimento é necessário não apenas para realizar boas estratégias de jogo, mas também para realizar as jogadas. Sendo assim, a tarefa de realizar jogadas corretamente representa algo importante para o desenvolvimento das noções pretendidas. Pensando no cenário de um jogo digital, é preciso analisar sobre como a máquina irá sugerir as jogadas: se serão disponibilizadas as opções de distribuição, se o software aceitará a inserção de jogadas equivocadas etc.

Além dessas questões apresentadas acima, que trazem aspectos da relação do *universo externo* com a *interface*, muitas outras questões podem vir a aparecer quando se analisa as restrições do *universo interno*. Essas questões que se remetem ao *universo interno* serão discutidas ao longo do processo de concepção do Mankala Colhe Três Digital.

### 3.3 LEVANTAMENTO DE REQUISITOS

O Levantamento de Requisitos da Engenharia Didático-Informática compreende, segundo Tiburcio (2020) a análise, a especificação e a validação de requisitos.

Como citamos anteriormente, Tiburcio orienta que a obtenção dos requisitos seja realizada por meio de quatro etapas, a serem detalhadas nos parágrafos seguintes.

Os questionamentos que norteiam a primeira etapa são: “*Como o ensino e a aprendizagem podem ser favorecidos? Como a compreensão dos saberes é auxiliada com o uso do software? Quais recursos e situações o software propõe para ajudar o usuário a compreender os conhecimentos?*” (TIBURCIO, 2020, p. 172-173).

Acreditamos que o ensino e a aprendizagem das noções suscetíveis de serem mobilizadas por meio do jogo podem ser favorecidos, primeiramente, pelos aspectos referentes ao potencial educativo do Mankala Colhe Três em sua versão tradicional (física), em que já possuímos um leque de resultados validados, tanto pelos criadores do jogo (ANDRADE *et al.*, 2013) quanto por Santos (2014) em sua pesquisa sobre o potencial educativo do jogo em situações didáticas.

Além disso, também acreditamos que o desenvolvimento de uma versão digital do jogo que considere os encaminhamentos para pesquisas futuras de Santos (2014) permitirá uma

ampliação do potencial educativo do jogo, tendo em vista que sua natureza computacional trará uma diversidade de possibilidades de situações de uso do jogo, reconstruído por um viés informático.

Os questionamentos que orientam a segunda etapa da obtenção de requisitos se relacionam à uma análise externa. São eles: “*quais funcionalidades existem em produtos da área? Quais são os possíveis diferenciais do software que se pretende desenvolver? O que o software trará de novo referente ao que já existe?*”.

Sobre essa análise externa, é importante enfatizar que, no caso de nossa pesquisa, o jogo Mankala Colhe Três tradicional representa a base do software educativo a ser concebido. Sendo assim, o comparativo que faremos é entre o jogo físico e a nossa proposta de jogo digital.

Considerando o exposto acima, acreditamos que os diferenciais da versão digital foram inicialmente elencados por Santos (2014) nos encaminhamentos futuros de seu trabalho, já que acreditamos que cada uma das sugestões levantadas pelo autor pode ser atendida por meio da concepção de uma versão digital do Mankala Colhe Três.

A terceira etapa da obtenção de requisitos é caracterizada pela elaboração do *documento de requisitos*, que, segundo Tiburcio (2020), deve conter as informações coletadas durante as etapas anteriores em uma linguagem clara e objetiva. O autor orienta a construção de um quadro que elenque os requisitos considerando as dimensões epistemológica, cognitiva, didática, informática e, caso necessário, de outras naturezas.

Ao buscar sistematizar os requisitos do Mankala Colhe Três Digital no documento de requisitos conforme descrito por Tiburcio (2020), nos deparamos com algumas questões que nos geraram reflexões acerca dessa etapa proposta pelo modelo de processo da Engenharia Didático-Informática.

O primeiro questionamento que nos surgiu foi relacionado à necessidade de uma distinção entre *requisitos educativos de natureza informática* e *requisitos computacionais* para a produção de um software educativo.

Ao inserir elementos da dimensão informática no documento de requisitos proposto por Tiburcio (2020), levantados ao longo do estudo teórico das Análises Prévia, tratamos de aspectos educativos a serem possibilitados ou potencializados por meio da informática. Portanto, na nossa percepção, ainda se trata de requisitos educativos, propostos a partir da análise da dimensão informática.

Acreditamos, no entanto, que há outra categoria de requisitos computacionais que precisam ser explicitados em um documento formal de requisitos para a produção do software

educativo, relacionados ao desempenho, usabilidade, confiabilidade, segurança, disponibilidade, manutenibilidade, tecnologias envolvidas na elaboração do software etc.

Considerando o exposto, elaboramos dois documentos de requisitos distintos para a concepção do Mankala Colhe Três Digital: o *documento de requisitos educativos*, inspirado na proposta de Tiburcio (2020), sistematizado no Quadro 3, e o *Modelo de Casos de Uso*, que representa o documento que caracterizará a realização da etapa de Concepção e Análise *a priori* do Mankala Colhe Três Digital, contendo detalhadamente os requisitos funcionais e não funcionais do sistema.

Outra questão que nos fez refletir foi o modo de sistematizar os requisitos educativos do Mankala Colhe Três Digital. Como estamos propondo a concepção de uma versão digital de um jogo matemático já existente, os requisitos do software precisam levar em consideração tanto as contribuições educativas advindas da versão tradicional do jogo quanto as novas contribuições originárias da presença da dimensão informática. Sendo assim, nossos requisitos educativos foram organizados considerando:

- A manutenção das finalidades educacionais do Mankala Colhe Três em sua versão tradicional (física), estipuladas pelos desenvolvedores do jogo (ANDRADE *et al.*, 2013);
- A manutenção do potencial educativo do Mankala Colhe Três, analisado por Santos (2014);
- Consideração dos encaminhamentos futuros levantados por Santos (2014);
- Novas possibilidades de uso do jogo considerando a presença da dimensão informática.

Ao analisar as contribuições trazidas pelos estudos das Análises Prévias, decidimos categorizar os nossos requisitos cognitivos, epistemológicos, didáticos e informáticos em dois blocos, que chamamos de *cognitivo-epistemológico* e *didático-informático* respectivamente, além de incluirmos uma dimensão denominada *sociocultural*, como pode ser visto a seguir:

Quadro 4 – Documento de requisitos educativos do Mankala Colhe Três Digital

Bloco cognitivo-epistemológico	Bloco didático-informático	Dimensão sociocultural
<ul style="list-style-type: none"> <li>• Permitir a quantificação mental;</li> <li>• Permitir a resolução de problemas com situações mistas;</li> <li>• Permitir a divisão por cálculo mental;</li> <li>• Permitir o mapeamento de possibilidades;</li> <li>• Permitir o reconhecimento de divisores de um número;</li> <li>• Permitir a identificação de múltiplos de um número;</li> <li>• Permitir o reconhecimento de números primos e compostos;</li> <li>• Permitir a manipulação direta das sementes do jogo.</li> </ul>	<ul style="list-style-type: none"> <li>• Permitir o uso do jogo por diferentes modelos de interação;</li> <li>• Permitir o equilíbrio entre ludicidade e intencionalidade didática;</li> <li>• Permitir o mapeamento completo das jogadas;</li> <li>• Possibilitar variações da configuração do jogo (número de covas, sementes, novas regras etc.);</li> <li>• Permitir o uso do jogo por um número grande de estudantes simultaneamente;</li> <li>• Permitir a realização de partidas de maneira mais ágil, de modo a favorecer a criação de sequências de ensino mais longas.</li> </ul>	<ul style="list-style-type: none"> <li>• Enfatizar o viés da Etnomatemática para explorar melhor o aspecto cultural do jogo, associando-o a sua provável origem africana.</li> </ul>

Fonte: elaborada pelo autor.

Decidimos categorizar os requisitos educativos dessa forma pelo fato de percebermos uma aproximação entre os requisitos oriundos das dimensões cognitiva e epistemológica, assim como os das dimensões didática e informática.

O Quadro 4 nos fornece um guia inicial do que esperamos que o Mankala Colhe Três Digital atenda em termos de requisitos educativos. Como afirmado anteriormente, decidimos detalhar os requisitos do software considerando tanto os requisitos educativos quanto os requisitos computacionais por meio da produção do Modelo de Casos de Uso do Mankala Colhe Três Digital. Esse documento, que representa um dos artefatos oriundos do processo RUP, baseará a etapa de Concepção e Análise *a priori*, como poderá ser mais bem compreendido na seção seguinte.

A quarta e última etapa para a obtenção dos requisitos, proposta por Tiburcio (2020), é apresentada pelo autor como segue:

Neste momento, a equipe irá verificar os requisitos quanto à pertinência, consistência e integralidade. Neste processo podem ser descobertos erros quanto aos requisitos levantados. Assim, os requisitos podem ser modificados, a fim de corrigir os problemas encontrados (TIBURCIO, 2020, p. 173).

Compreendemos que esse tipo de verificação deverá ser realizado ao longo do processo de produção do software, tendo em vista que, como destacado por Tiburcio, “a Engenharia Didático-Informática orienta que o gerenciamento dos requisitos deve ser realizado durante todo o ciclo de vida do software” (TIBURCIO, 2020, p. 172).

## 4 REALIZAÇÃO DO CICLO HIPOTÉTICO-EXPERIMENTAL NA CONCEPÇÃO DO MANKALA COLHE TRÊS DIGITAL

Neste capítulo, apresentaremos nossas considerações sobre a realização das etapas relacionadas ao ciclo Hipotético-experimental da EDI para a concepção do Mankala Colhe Três Digital.

A primeira etapa a ser apresentada é a de Concepção e Análise *a priori*. Como detalharemos a seguir, a realização dessa etapa é representada pela produção do Modelo de Casos de Uso do Mankala Colhe Três Digital. Esse documento guiará toda a discussão da primeira seção do presente capítulo.

Após nossas considerações acerca da Concepção e Análise *a priori* será dado início à abordagem sobre a etapa de Prototipação e, seguida, entraremos na discussão referente à etapa de Desenvolvimento, que, como será detalhado mais adiante, nos levou a realizar uma adaptação na condução das próximas etapas de nossa pesquisa.

### 4.1 CONCEPÇÃO E ANÁLISE A *PRIORI*

A etapa de Concepção e Análise *a priori* da Engenharia Didático-Informática é caracterizada pelo desenvolvimento das situações de utilização do software educativo, considerando os referenciais teóricos e metodológicos elencados como auxiliares para ensino e aprendizagem. É nessa fase que são idealizadas as interações com os usuários (TIBURCIO, 2020).

Para a realização dessa etapa, decidimos construir o Modelo de Casos de Uso do Mankala Colhe Três Digital, que representa um tipo de documento de requisitos centrado no detalhamento dos requisitos funcionais (Casos de Uso) e não funcionais do sistema. Esse documento foi escolhido para ser utilizado em nossa pesquisa justamente por possuir grande afinidade com as características da Concepção e Análise *a priori*, além de nos oferecer uma complementação ao documento de requisitos educativos, por meio dos requisitos não funcionais.

A utilização do Modelo de Casos de Uso, para entender e modelar o problema relacionado ao software a ser desenvolvido, representa uma técnica recomendada pelo Processo Unificado da *Rational* (RUP<sup>7</sup>).

---

<sup>7</sup> Sigla relativa à versão original em inglês *Rational Unified Process*.

Para compreendermos melhor esse documento, apresentaremos algumas definições da Linguagem de Modelagem Unificada (UML<sup>8</sup>) (BOOCH *et al.*, 2006). Iniciemos pelo *caso de uso*, que, segundo os criadores dessa linguagem, “é uma descrição de um conjunto de sequências de ações, inclusive variantes, que um sistema executa para produzir um resultado de valor observável por um ator” (BOOCH *et al.*, 2006, p. 230). Veremos mais adiante que o caso de uso, na modelagem de um sistema, é representado visualmente por uma elipse.

É importante destacar a distinção a ser feita entre a visão interna e a visão externa de um sistema na realização de uma modelagem, pois, como enfatizam Booch *et al.*, “um caso de uso descreve *o que* um sistema (ou um subsistema, classe ou interface) faz, mas ele não especifica *como* isso é feito” (BOOCH *et al.*, 2006, p. 232). Para que o comportamento de um caso de uso seja especificado, podemos descrever o *fluxo de eventos* desse caso de uso de forma clara, de modo a facilitar a compreensão de todos os interessados no projeto do software.

Sobre o fluxo de eventos, Kruchten (2003) afirma que

A parte mais importante do caso de uso no fluxo de requisitos é seu fluxo de eventos. O fluxo de eventos descreve a sucessão de ações entre o ator e o sistema. É escrito em linguagem natural, numa prosa simples, consistente, com um uso preciso de condições que utilizam um glossário comum do domínio do problema (KRUCHTEN, 2003, p. 84).

O fluxo de eventos de um caso de uso deverá incluir detalhadamente o momento de início e fim do caso de uso, além de informações sobre as interações com os atores, a transferência de objetos e detalhamentos sobre a presença de fluxos principais e alternativos.

Outra definição importante acerca da UML é a de *ator*. Segundo Booch *et al.* (2006),

um ator representa um conjunto coerente de papéis que os usuários de casos de uso desempenham quando interagem com esses casos de uso. Tipicamente, um ator representa um papel que um ser humano, um dispositivo de hardware ou até outro sistema desempenha com o sistema (BOOCH *et al.*, 2006, p. 231).

A representação visual de um ator é um “boneco palito”, como poderá ser observado mais adiante. Partindo das primeiras definições da Linguagem de Modelagem Unificada, podemos apresentar seguinte definição de Modelo de Casos de Uso:

O modelo de caso de uso, então, consiste no conjunto de todos os casos de uso para o sistema ou uma porção do sistema, junto com o conjunto de todos os atores que interagem com estes casos de uso, descrevendo assim a funcionalidade completa do sistema. Fornece um modelo das funções planejadas do sistema e seu ambiente e pode

---

<sup>8</sup> Sigla relativa à versão original em inglês *Unified Modeling Language*.

servir como um contrato entre o cliente e os desenvolvedores (KRUCHTEN, 2003, p. 85).

O Modelo de Casos de Uso do Mankala Colhe Três Digital foi produzido e seu conteúdo foi inserido no decorrer desta seção do trabalho. Esse tipo de documento de requisitos apresenta, geralmente, os seguintes elementos acerca do sistema a ser produzido:

- a. Histórico de alterações;
- b. Introdução e descrição geral do projeto;
- c. Diagrama de casos de uso;
- d. Requisitos funcionais do sistema (casos de uso);
- e. Requisitos não funcionais do sistema.

O *histórico de alterações* do Modelo de Casos de Uso converge com as orientações trazidas por Tiburcio (2020) sobre a necessidade de que não tratemos o documento de requisitos do software como um documento rígido e imutável. De fato, nosso Modelo de Casos de Uso sofreu ao menos três modificações consideráveis ao longo do processo de concepção do jogo, resultantes de ajustes acerca do que se espera do jogo, além de adaptações em termos da estrutura textual do documento, como pode ser visto no quadro a seguir:

Quadro 5 – Histórico de alterações do Modelo de Casos de Uso do Mankala Colhe Três Digital

<b>Data</b>	<b>Versão</b>	<b>Descrição</b>	<b>Autor(es)</b>
05/jul/2022	1.0	Construção do Modelo de Casos de Uso	Tarcisio Rocha
19/fev/2023	1.1	Aprimoramento do Modelo de Casos de Uso	Tarcisio Rocha
28/jul/2023	1.2	Aprimoramento do Modelo de Casos de Uso	Tarcisio Rocha

Fonte: elaborado pelo autor

Na *introdução e descrição geral do projeto*, trazemos como base os aspectos apontados na etapa anterior da presente pesquisa, referente aos requisitos educativos do software, que consideram tanto as contribuições trazidas pelas pesquisas que envolveram o Mankala Colhe Três em sua versão tradicional quanto as novas contribuições que a versão digital do jogo poderá trazer, após a inserção da dimensão informática nos estudos realizados.

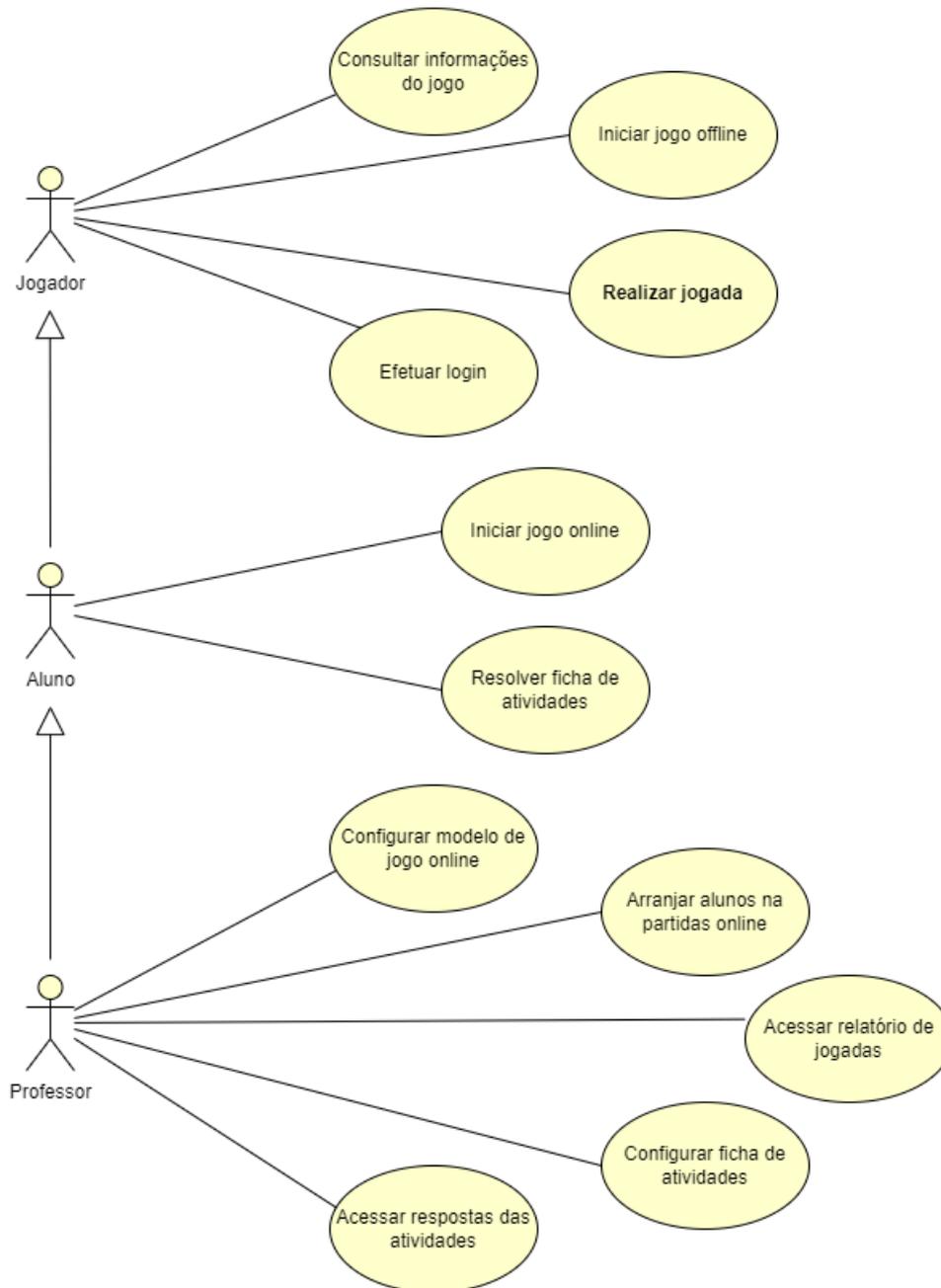
Um importante elemento que compõe o Modelo de Casos de Uso é o *Diagrama de Casos de Uso*. Esse tipo de diagrama representa um dos 14 tipos de diagramas de modelagem da UML (BOOCH *et al.*, 2006).

Um diagrama de caso de uso é um diagrama que mostra um conjunto de casos de uso, atores e seus relacionamentos. Esse tipo de diagrama contém: assunto; casos de uso; atores; e relacionamentos de dependência, generalização e associação (BOOCH *et al.*, 2006). Além disso, esse tipo de diagrama se mostra essencial para o gerenciamento dos requisitos funcionais de um sistema, pois, como enfatizam os criadores dessa linguagem de modelagem,

Esse diagrama é valioso, por oferecer um ponto de partida comum para usuários finais, especialistas de domínio e desenvolvedores para a visualização, especificação, construção e documentação de suas decisões sobre os requisitos funcionais desse sistema (BOOCH *et al.*, 2006, p. 247).

Apresentamos, a seguir, o Diagrama de Casos de Uso do Mankala Colhe Três, presente no nosso Modelo de Casos de Uso, contendo os requisitos funcionais do sistema:

Figura 14 – Diagrama de Casos de Uso do Mankala Colhe Três Digital



Fonte: elaborado pelo autor.

O diagrama apresentado acima contém os casos de uso do sistema, que são requisitos funcionais, representados por elipses, além dos atores, representados por bonecos palito. As linhas e setas representam relacionamentos, a serem detalhados a seguir.

O Diagrama de Casos de Uso possui alguns tipos de relacionamentos da UML, no entanto, focaremos nossa descrição nos relacionamentos presentes no Diagrama de Casos de Uso do Mankala Colhe Três.

Os relacionamentos entre um ator e um caso de uso são sempre de *associação*, representado por uma linha contínua. A associação entre um ator e um caso de uso define uma funcionalidade do sistema do ponto de vista do usuário (BOOCH *et al.*, 2006).

Relacionamentos entre atores são sempre de *generalização*. Esse tipo de relacionamento, representado por uma seta contínua de ponta triangular, considera que um ator herda os casos de uso que estão relacionados ao outro ator (BOOCH *et al.*, 2006).

No nosso diagrama da Figura 14 podemos perceber, por exemplo, que o *jogador* se relaciona com quatro funcionalidades do sistema. O *aluno*, por sua vez, por ser herdeiro do ator jogador, pode realizar todos os quatro casos de uso que um jogador realiza, mas possui mais outros dois casos de uso específicos, que são *realizar jogo online* e *resolver ficha de atividades*. Por fim, percebemos que o *professor* pode realizar, além das funcionalidades disponíveis a um aluno, que inclui as funcionalidades do *jogador*, outras cinco funcionalidades do sistema, representadas pelas elipses relacionadas diretamente ao *professor*.

Ao observar o seu Diagrama de Casos de Uso, é possível perceber que o Mankala Colhe Três Digital poderá ser utilizado tanto para a realização de partidas *offline*, sem que haja necessariamente uma intenção didática nesse uso, focando, deste modo, no potencial lúdico do jogo em sua versão tradicional, como também por usuários cadastrados e logados no sistema, que utilizarão as funcionalidades pensadas para permitir uma ampliação do potencial educativo do jogo, se comparado a sua versão física.

Apresentaremos, a seguir, os requisitos funcionais do Mankala Colhe Três Digital e, posteriormente, os requisitos não funcionais do sistema.

#### **4.1.1. Requisitos funcionais do Mankala Colhe Três Digital**

Após a apresentação do Diagrama de Casos de Uso, o Modelo de Casos de Uso traz o olhar aos requisitos funcionais do sistema, representados por casos de uso. Como veremos a seguir, cada caso de uso apresentado pode conter as seguintes informações: descrição; atores envolvidos; entradas e precondições; saídas e pós-condições; fluxo de eventos principal; e fluxo de eventos secundário. Além disso, ao longo dessa apresentação das funcionalidades do sistema, são trazidas observações adicionais que ajudam a justificar as escolhas realizadas. Na subseção a seguir, apresentaremos detalhadamente cada um dos casos de uso pensados para o sistema.

#### 4.1.1.1 Consultar informações do jogo

A funcionalidade “Consultar informações do jogo” foi criada com o intuito de permitir que os jogadores, independente se estão “logados” no sistema ou não, possam acessar tanto as regras do jogo, de forma textual, quanto informações do Mankala Colhe Três e sua origem, para que o aspecto cultural do jogo possa ser enfatizado. Segue o quadro do caso de uso:

Quadro 6 – Caso de uso consultar informações do jogo

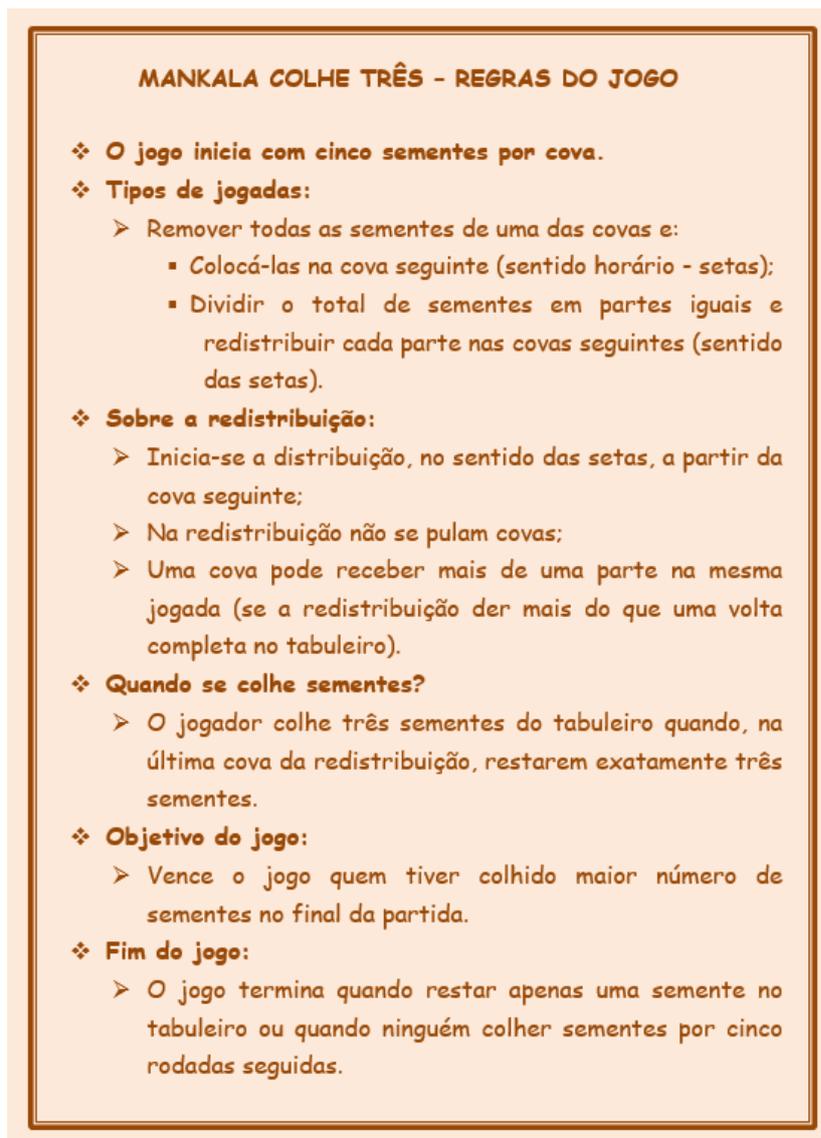
<b><i>Consultar informações do jogo</i></b>	
<b><i>Descrição</i></b>	Função que permite ao usuário ter acesso a informações relacionadas aos aspectos culturais dos jogos do tipo mankala e das regras do jogo, por meio de textos, imagens, vídeos e links de acesso a páginas da Web.
<b><i>Ator</i></b>	Jogador.

Fonte: elaborado pelo autor.

Todas as informações a serem disponibilizadas ao jogador sobre o aspecto histórico e cultural dos jogos da família mankala possuem o papel fundamental de exploração da origem desses jogos, relacionando à sua origem africana, de modo a estabelecer um resgate cultural rico e fundamental para o ambiente escolar e fora dele. A partir dessa funcionalidade, o professor terá a oportunidade de tratar de temáticas muito importantes, fundamentando-se, inclusive, no que é discutido nos trabalhos que relacionam o uso dos jogos mankala e a Etnomatemática, conforme discutido na seção 3.2.2 deste trabalho.

Em relação às regras do jogo disponibilizadas neste caso de uso, destacamos que elas são baseadas nas sugeridas e utilizadas por Santos (2014), que considera o jogo em sua versão de criação no Projeto Rede (ANDRADE *et al.*, 2013), mas com pequenos ajustes na forma de instrução. No nosso caso, fizemos mais um pequeno ajuste em uma das regras de finalização de partida: ao invés de o jogo finalizar quando restarem quatro sementes no tabuleiro, substituímos por apenas uma semente, tendo em vista que a regra anterior poderia causar frustrações nos jogadores que, diante de uma situação possível de colheita em um tabuleiro com quatro sementes, precisavam encerrar a partida. Segue a ficha de regras:

Figura 15 – Ficha de regras do Mankala Colhe Três



Fonte: adaptada de Santos (2024, p. 45).

A ficha de regras apresentada na Figura 15, baseada nos estudos de Santos (2014), veio da busca por contribuir com o processo de *devolução* das situações envolvendo o jogo, aspecto de grande importância da Teoria das Situações Didáticas para o favorecimento da vivência de situações *adidáticas* pelos sujeitos. No nosso caso, a disponibilização da ficha de regras como uma funcionalidade do Mankala Colhe Três Digital além de servir como elemento central do processo de devolução das situações, também possui o papel de auxiliar jogadores na utilização do jogo fora do ambiente escolar e de servir de base instrucional do jogo para possíveis adaptações posteriores das regras pelo professor, como detalharemos em casos de uso mais adiante.

#### 4.1.1.2 Iniciar jogo *offline*

O caso de uso “Iniciar jogo *offline*” representa uma funcionalidade do jogador que possibilita a realização de partidas do jogo sem que haja, necessariamente, intenção didática em seu uso. As partidas *offline* foram idealizadas para duas modalidades diferentes, permitindo a realização de partidas entre dois jogadores no mesmo computador, que se alternam nas jogadas, ou jogador contra o computador. Neste último tipo, pensou-se na presença de três níveis de dificuldade de jogo, que dependem do nível de jogada que o computador realizará durante as partidas, em termos de análise de jogadas favoráveis. Segue o quadro que apresenta as informações do caso de uso:

Quadro 7 – Caso de uso iniciar jogo online

<b><i>Iniciar jogo offline</i></b>	
<b><i>Descrição</i></b>	Função em que o usuário Jogador dá início a realização de uma partida <i>offline</i> , tanto em partidas do tipo “jogador 1 contra jogador 2” quanto em partidas “jogador contra computador”.
<b><i>Ator</i></b>	Jogador.
<b><i>Fluxo de eventos principal</i></b>	<ol style="list-style-type: none"> <li>1. O usuário seleciona, dentre as opções do menu principal do sistema, o início de jogo <i>offline</i>;</li> <li>2. O sistema exibirá uma mensagem demandando do usuário a seleção da modalidade do jogo <i>offline</i>, dentre as opções “jogador 1 contra jogador 2” e “jogador contra computador”; <ul style="list-style-type: none"> <li>• No caso de o jogador selecionar a opção de jogo <i>offline</i> “jogador contra computador”, o sistema exibirá uma mensagem demandando do usuário a seleção do nível de dificuldade da partida, dentre as opções “fácil”, “médio” e “difícil”;</li> </ul> </li> <li>3. Após a seleção da modalidade do jogo <i>offline</i>, o sistema demandará do usuário a inserção do(s) nome(s) dos jogadores, de modo a identificar os adversários de partida de maneira personalizada;</li> <li>4. O fluxo desse caso de uso se encerra por meio da exibição do tabuleiro do Mankala Colhe Três Digital em sua versão tradicional (25 sementes, distribuídas igualmente em 5 covas dispostas</li> </ol>

---

circularmente) preparado para a realização da primeira jogada. Em relação a quem iniciará as partidas, quando estas forem do tipo “Jogador contra Computador” será feita pelo usuário Jogador; já nas partidas do tipo “Jogador 1 contra Jogador 2”, esse início poderá ser feito por qualquer um dos jogadores, validada pelo que a realizar primeiro.

---

Fonte: elaborado pelo autor.

É importante destacar três observações relacionadas ao último passo do fluxo principal de eventos:

- A escolha de permitir que qualquer um dos dois jogadores possa iniciar a partida baseia-se nas regras da versão clássica do jogo, que propõe que a escolha de quem inicia as partidas seja feita pelos próprios jogadores;
- A escolha de manter fixa a configuração do jogo na versão *offline* se deve ao fato de que nossa ideia de propor outras opções de configuração em uma versão digital do Mankala Colhe Três tem o objetivo de criar variações do jogo com finalidades didáticas, o que se distancia da proposta de jogo *offline*;
- A presente pesquisa não possui o foco de realizar um estudo minucioso sobre as implicações das variações dos valores das variáveis (número de covas, de sementes por cova, regras de finalização de jogo), mas produzir um software educativo que permita essa variação, favorecendo esse tipo de análise posteriormente, em pesquisas futuras.

Na seção seguinte, relativa ao caso de uso *realizar jogada*, trataremos, como observações adicionais, mais detalhes de como idealizamos a hierarquização das jogadas pelo computador nas partidas *offline* da modalidade “jogador contra computador”, de modo a caracterizar as jogadas dos níveis “fácil”, “médio” e “difícil”.

#### 4.1.1.3 Realizar jogada

Este caso de uso representa uma das principais funcionalidades do Mankala Colhe Três Digital e se destina ao ator jogador, que inclui, por herança, o aluno e o professor. Apresentaremos, a seguir, o quadro do caso de uso *realizar jogada*, que, além de sua descrição, também contém elementos de entrada e precondições, de saída e pós-condições, além do fluxo de eventos principal e alternativos. Após a apresentação do quadro com os elementos do caso

de uso, traremos algumas observações adicionais referentes a como estamos idealizando pelo sistema: a colheita de sementes; a finalização de partidas; e a realização de jogadas pelo computador nas partidas *offline* do tipo “jogador contra computador”.

Quadro 8 – Caso de uso realizar jogada

<b><i>Realizar jogada</i></b>	
<b><i>Descrição</i></b>	Neste caso de uso, o ator deverá realizar uma das possíveis jogadas a partir de uma cova pré-selecionada.
<b><i>Ator</i></b>	Jogador.
<b><i>Entradas ou pré-condições</i></b>	O jogador precisa ter realizado o caso de uso <i>iniciar jogo offline</i> ou o caso de uso <i>iniciar jogo online</i> .
<b><i>Saídas ou pós-condições</i></b>	<p>Após a confirmação e registro do tipo de jogada realizada, que representa a última etapa do fluxo de eventos principal deste Caso de Uso, o sistema poderá fornecer um dos quatro tipos de saídas:</p> <ul style="list-style-type: none"> <li>• O sistema “passa a vez” para o adversário, por meio da permissão de seleção de uma cova pelo outro jogador;</li> <li>• O sistema realiza uma colheita de sementes e, em seguida, “passa a vez” para o adversário, por meio da permissão de seleção de uma cova pelo outro jogador;</li> <li>• O sistema realiza uma colheita de sementes e, em seguida, finaliza a partida, pelo critério de número total de sementes no tabuleiro;</li> <li>• O sistema finaliza a partida pelo critério de número de rodadas sem que haja colheita de sementes.</li> </ul>
<b><i>Fluxo de eventos principal</i></b>	<ol style="list-style-type: none"> <li>1. <b><i>Selecionar cova:</i></b> o usuário Jogador, que inclui como herança os atores Aluno e Professor, deverá selecionar uma das covas do tabuleiro que contenha ao menos uma semente, por meio de um clique.</li> <li>2. <b><i>Selecionar uma quantidade de sementes da cova:</i></b> <ol style="list-style-type: none"> <li>2.1. <b><i>Manipulação de sementes:</i></b> após a seleção da cova pelo Jogador, o sistema exibirá uma ampliação da cova selecionada, em uma região da interface do usuário, com suas sementes manipuláveis, de modo a permitir uma livre exploração das sementes dessa cova</li> </ol> </li> </ol>

---

antes da escolha da jogada a ser realizada. O sistema deverá permitir que o Jogador mova as sementes na região da cova ampliada, especificada na interface do usuário, ao clicar e arrastar cada uma das sementes da cova selecionada;

2.2. *Determinação do tipo de jogada:* após a manipulação de sementes, o usuário deverá selecionar uma quantidade de sementes dessa cova que seja divisor do número total de sementes da cova, para representar quantas sementes comporão cada grupo a ser distribuído nas covas seguintes, segundo o sentido horário do tabuleiro, caracterizando, desse modo, a realização do tipo de jogada. Ou seja, o número “ $q$ ” de sementes selecionado para a realização da jogada precisa ser divisor do número total “ $p$ ” de sementes da cova.

- i. Essa seleção será iniciada por meio de um clique em um botão na interface do usuário denominado “selecionar semente(s)”.
- ii. Posteriormente, o usuário poderá selecionar as  $q$  sementes tanto por meio de cliques sob elas quanto por meio da seleção simultânea de várias sementes, ao clicar e arrastar o ponteiro do mouse sob uma região da cova que cubra várias sementes, tornando mais prática a seleção em casos de covas com muitas sementes.

2.3. *Confirmação e registro do tipo de jogada realizada:* após a seleção de um número  $q$  de sementes da cova, o Jogador deverá confirmar a sua escolha do tipo de jogada, por meio do clique no botão “realizar jogada”, disponível na interface do usuário. A partir de então, o sistema exibirá a realização da jogada, por meio da retirada de todas as sementes da cova selecionada e a redistribuição das  $p$  sementes,  $q$  a  $q$ , nas covas seguintes (sentido horário do tabuleiro) e fará o registro da jogada, por meio de representações do tabuleiro do jogo conforme Santos (2014) propôs na realização do mapeamento parcial do Mankala Colhe Três, explicitados na seção 3.2.3.2. deste trabalho.

---

---

*Fluxo de eventos secundário (alternativos ou exceções)*

- **Erro de escolha de cova:** Caso o Jogador selecione uma cova sem sementes para a realização de sua jogada, o sistema exibirá uma mensagem de erro de seleção de cova e permitirá ao usuário uma nova escolha. Caso esse procedimento se repita por três vezes em uma mesma partida, o jogo é finalizado por “erros frequentes” e o sistema exhibe as regras do Jogo;
- **Erro de seleção de quantidade de sementes:** Caso o Jogador selecione uma quantidade de sementes diferente das possíveis para a realização da jogada (inclusive zero, para o caso de não ter selecionado semente alguma) e clique no botão “realizar jogada”, o sistema exibirá uma mensagem de erro de tipo de jogada e, posteriormente, o Jogador voltará a ter a vez para a escolha de uma nova cova. Caso esse procedimento se repita por três vezes em uma mesma partida, o jogo é finalizado por “erros frequentes” e o sistema exhibe as regras do Jogo.

---

Fonte: elaborado pelo autor.

Gostaríamos de fazer um destaque ao subtópico 2.1 do Fluxo de eventos principal: a *Manipulação de sementes*. Como discutimos na seção deste trabalho referente à transposição didático-informática do Mankala Colhe Três, precisamos nos ater às modificações que a criação de uma versão digital do jogo pode trazer e como essas modificações podem ter impactos positivos ou negativos ao aspecto educativo do jogo. De fato, ao pensarmos na realização de uma jogada no Mankala Colhe Três Digital, nos deparamos com possíveis perdas educativas que a ausência de manipulação direta com as sementes do jogo poderia causar, principalmente ao considerar os jogadores mais novos, e, por isso, decidimos criar essa possibilidade de manipulação direta e livre das sementes durante o planejamento da jogada pelo usuário.

Como pode ser visto nas possibilidades de saídas ou pós-condições do presente caso de uso, o sistema precisa lidar com a colheita de sementes e com a finalização de partida pelos critérios de número de sementes restantes no tabuleiro e de número de rodadas sem que haja colheita de sementes. Em termos de observações adicionais, ilustraremos como temos idealizado estas ações:

- **Colheita de sementes:** conforme já apresentado na ficha de regras do Mankala Colhe Três, presente no caso de uso *consultar informações do jogo*, a colheita de sementes é realizada se, após a realização de uma jogada, restarem exatamente 3 sementes na última

cova de redistribuição. Neste caso, o sistema deverá retirar as sementes desta cova e deverá adicionar 3 sementes à pontuação do jogador que realizou a jogada;

- **Finalização de partida pelo número total de sementes no tabuleiro:** Se após uma colheita de sementes restarem apenas “ $x$ ” sementes no tabuleiro como um todo, o sistema deverá: (i) encerrar a partida; (ii) exibir o número de sementes colhidas por ambos os adversários de jogo; e (iii) exibir mensagem informando qual o vencedor do jogo, ou se houve um empate.
  - Para o modelo de jogo *offline*, que mantém as regras do Mankala Colhe Três em sua versão clássica (física) adaptada,  $x = 1$ ;
  - No modelo de jogo *online*, o valor de  $x$  é definido no momento da configuração do modelo de jogo *online* pelo Professor, como será detalhado em outro Caso de Uso deste documento;
- **Finalização de partida pelo número de rodadas sem colheita de sementes:** Se, no decorrer da partida, não houver colheita de sementes por “ $y$ ” rodadas consecutivas, o sistema deverá: (i) encerrar a partida; (ii) exibir o número de sementes colhidas por ambos os adversários de jogo; e (iii) exibir mensagem informando qual o vencedor do jogo, ou se houve um empate.
  - Para o modelo de jogo *offline*, que mantém as regras do Mankala Colhe Três em sua versão clássica (física),  $y = 5$ ;
  - No modelo de jogo *online*, o valor de  $y$  é definido no momento da configuração do modelo de jogo *online* pelo Professor, como será detalhado em outro Caso de Uso deste documento.

Por fim, conforme elucidado no caso de uso *iniciar jogo offline*, nas partidas do tipo “jogador contra computador” algumas jogadas serão realizadas pelo sistema. Apesar de essa ação do sistema não fazer parte necessariamente do presente Modelo de Casos de Uso, destinada aos atores Jogador, Aluno e Professor, trataremos a seguir o que temos pensado para esse funcionamento do sistema, que realizará jogadas de diferentes maneiras, a depender do nível de dificuldade de jogo determinado pelo jogador *offline* (fácil, médio ou difícil).

Assim como nas jogadas realizadas pelo ator jogador, considerando  $p \in N$  o número total de sementes da cova selecionada para a realização da jogada e  $q \in N$  o número de sementes de cada parte a ser redistribuída no tabuleiro, as seguintes condições são necessárias na realização das jogadas pelo computador:

- $1 \leq q \leq p$ ;

- $\frac{p}{q} \in \mathbb{N}$  ( $q$  é divisor de  $p$ ).

Para que as jogadas realizadas pelo computador nas partidas do tipo “jogador contra computador” sejam classificadas em níveis de dificuldade, o sistema deverá possuir um mapeamento das possibilidades de jogadas, mesmo que de forma parcial, considerando duas rodadas do jogo e considerar os seguintes procedimentos:

- No nível *fácil*, o sistema selecionará aleatoriamente uma cova que contenha, ao menos, uma semente e realizará aleatoriamente uma das jogadas possíveis a partir dessa cova;
- No nível *médio*, o sistema selecionará aleatoriamente (se houver mais de uma possibilidade) uma cova que permita a realização de uma jogada que resulte em uma *colheita de sementes* e efetuará a colheita. Caso não haja a possibilidade de colheita de sementes na presente posição do jogo, o sistema deverá selecionar uma das covas aleatoriamente e realizar uma jogada aleatoriamente a partir dessa cova;
- No nível *difícil*, o sistema selecionará aleatoriamente (se houver mais de uma possibilidade) uma cova que permita a realização da melhor *jogada favorável*, e a realizará. Consideramos a classificação de jogadas favoráveis pela hierarquização realizada por Santos (2014) e explanada na seção 3.2.3.2 do presente trabalho, que traz o olhar para a análise da possibilidade de colheita de sementes na presente jogada e da posição de jogo deixada para o adversário na jogada seguinte.

#### 4.1.1.4 Efetuar *login*

Este caso de uso representa uma funcionalidade que, ao seu realizada, possibilita aos usuários cadastrados no sistema terem acesso a funcionalidades específicas de seu perfil (professor ou aluno). Segue o quadro do caso de uso *efetuar login*:

Quadro 9 – Caso de uso efetuar *login*

<b><i>Efetuar login</i></b>	
<b><i>Descrição</i></b>	Função em que o usuário cadastrado no sistema realiza o <i>login</i> para acessar funcionalidades específicas de seu perfil (Professor ou Aluno).
<b><i>Ator</i></b>	Jogador.
<b><i>Entradas ou condições</i></b>	O usuário deve estar cadastrado no sistema.

---

**Saídas ou pós-condições** Se o caso de uso for realizado com sucesso o usuário terá acesso a algumas funcionalidades adicionais do sistema de acordo com o seu perfil (Aluno ou Professor).

---

**Fluxo de eventos principal**

1. Deve aparecer um formulário solicitando o login e a senha do usuário;
2. O usuário informa os dados solicitados e solicita a efetivação do *login*;
3. O *login* do usuário é validado verificando se ele é Aluno ou Professor;
4. Em seguida, o sistema efetiva o *login* do usuário, exibe uma mensagem de sucesso e vai oferecer ao ator as funcionalidades adicionais correspondentes ao seu perfil.

---

**Fluxo de eventos secundário (alternativos ou exceções)**

- **Usuário não Cadastrado:** se o usuário ainda não for cadastrado, aparecerá uma opção para que ele possa se cadastrar no sistema, executando, assim, o caso de uso;
- **Senha Inválida:** caso a senha do usuário não seja válida, ou seja, não confira com o *login* informado, o sistema deverá apresentar uma mensagem solicitando que seja digitada a senha correta;
- **Login Inexistente:** caso o usuário digite um *login* que não exista, o sistema deve exibir uma mensagem solicitando que ele digite o login correto;
- **Esquecer senha e/ou login:** caso o ator esqueça a sua senha e/ou o seu *login*, o sistema deverá fornecer uma opção para que ele possa receber um e-mail com seu login e senha corretos.

---

Fonte: elaborado pelo autor.

#### 4.1.1.5 Iniciar jogo *online*

O caso de uso *iniciar jogo online* tem como ator o *aluno*, que, além das funcionalidades de um *jogador*, representa um perfil de usuário cadastrado e logado no sistema. Como será detalhado no quadro abaixo, para que o aluno inicie um jogo *online*, é necessário que o professor tenha preparado para ele esse tipo de jogo, por meio da configuração do modelo de jogo *online* e do arranjo dos alunos nas partidas *online*. Segue o quadro do caso de uso *iniciar jogo online*:

Quadro 10 – Caso de uso iniciar jogo *online*

<b><i>Iniciar jogo online</i></b>	
<b><i>Descrição</i></b>	Função em que o aluno seleciona a opção de iniciar jogo <i>online</i> no menu principal do sistema. Essa função será disponibilizada pelo sistema após o professor configurar o modelo de jogo <i>online</i> e arranjar os alunos nas partidas <i>online</i> .
<b><i>Ator</i></b>	Aluno.
<b><i>Entradas ou precondições</i></b>	Os usuários aluno e professor precisam estar logados no sistema e o professor deverá ter realizado os casos de uso <i>configurar modelo de jogo online</i> e <i>arranjar alunos nas partidas online</i> .
<b><i>Fluxo de eventos principal</i></b>	<ol style="list-style-type: none"> <li>1. Clicar no botão do menu inicial “iniciar jogo <i>online</i>”;</li> <li>2. <b><i>Informações sobre as regras de finalização de jogo configuradas pelo professor e sobre o arranjo dos alunos também realizado pelo professor:</i></b> verificar as mensagens informativas sobre as regras de finalização de jogo e se o jogo configurado pelo professor indicará que o aluno deverá permanecer na sua máquina para jogar individualmente contra outro aluno ou se ele precisará se juntar a outro colega em uma mesma máquina, para a realização de partidas do tipo “dupla contra dupla”. Essas indicações serão apresentadas pelo sistema por meio de mensagens na tela do aluno;</li> <li>3. O fluxo desse caso de uso se encerra por meio da exibição do tabuleiro do Mankala Colhe Três Digital em sua versão configurada pelo professor, preparado para a realização da primeira jogada, que poderá ser feita por qualquer um dos dois adversários. O sistema reconhecerá a primeira jogada daquele(s) que a fizer primeiro.</li> </ol>

Fonte: elaborado pelo autor.

Destacamos que a escolha de permitir que qualquer um dos dois jogadores possa iniciar a partida é baseada nas regras da versão clássica do jogo, que propõe que a escolha de quem inicia as partidas sejam feitas pelos próprios jogadores.

#### 4.1.1.6 Resolver ficha de atividades

O caso de uso *resolver ficha de atividades* representa mais uma funcionalidade destinada ao ator *aluno*. Este caso de uso ficará disponível ao aluno a partir do momento em que o professor configurar a ficha de atividades. Segue, o quadro abaixo, os detalhes do presente caso de uso:

Quadro 11 – Caso de uso resolver ficha de atividades

<b><i>Resolver ficha de atividades</i></b>	
<b><i>Descrição</i></b>	Função em que o aluno resolve a ficha de atividades configurada e disponibilizada pelo professor.
<b><i>Ator</i></b>	Aluno.
<b><i>Entradas ou precondições</i></b>	O usuário deverá ter efetuado o <i>login</i> como aluno e o professor precisará ter configurado e disponibilizado a ficha de atividades no sistema, por meio da realização do caso de uso <i>configurar ficha de atividades</i> .
<b><i>Saídas ou pós-condições</i></b>	Se o caso de uso for realizado, o sistema deverá registrar as respostas das atividades, para que o professor possa consultá-las posteriormente.
<b><i>Fluxo de eventos principal</i></b>	<ol style="list-style-type: none"> <li>1. Clicar no botão “Resolver ficha de atividades”, disponível no menu principal do sistema, que resultará na exibição da ficha de atividades configurada pelo professor, contendo espaços de inserção de texto nas respostas de cada atividade;</li> <li>2. Cada atividade poderá ser respondida pela inserção de informações digitadas pelo jogador. Nas atividades 2 e 4, as respostas deverão ser inseridas por meio de textos enquanto nas atividades 1 e 3, os alunos deverão digitar os números que representam as quantidades de sementes nas covas.</li> </ol>

Fonte: elaborado pelo autor.

Um exemplo de ficha de atividades configurada pelo professor, que ilustra bem a maneira como o presente caso de uso está sendo idealizado, está presente mais adiante neste modelo de casos de uso, mais especificamente na seção referente ao caso de uso *configurar ficha de atividades*.

#### 4.1.1.7 Configurar modelo de jogo *online*

Este caso de uso possui como ator o *professor*. Veremos, no quadro a seguir, que a configuração do modelo de jogo *online* permitirá ao professor realizar variações nos elementos do jogo (número de covas do tabuleiro, número inicial de sementes por cova e regras de finalização de partida).

Quadro 12 – Caso de uso configurar modelo de jogo *online*

<b><i>Configurar modelo de jogo online</i></b>	
<b><i>Descrição</i></b>	Função em que o professor configura o tipo de jogo a ser realizado pelos alunos, por meio do controle das seguintes variáveis: (a) número de covas do tabuleiro; (b) número de sementes por cova no início do jogo; regras de finalização das partidas, que por sua vez, contém como novas variáveis (c) o número de rodadas sem colheita de sementes e (d) o número de sementes restantes no tabuleiro.
<b><i>Ator</i></b>	Professor.
<b><i>Entradas ou precondições</i></b>	O professor precisa estar logado no sistema.
<b><i>Saídas ou pós-condições</i></b>	Como saída da configuração do modelo de jogo, o sistema fornecerá aos alunos a explicitação das regras de finalização do jogo, por meio de uma mensagem na tela do aluno, e o tabuleiro do jogo, com o número de covas e o número inicial de sementes por cova, conforme configurado pelo professor.
<b><i>Fluxo de eventos principal</i></b>	<ol style="list-style-type: none"> <li>1. Ao realizar o <i>login</i> do sistema, o professor terá, dentre as opções disponíveis no menu principal, a função "Configurar modelo de jogo <i>online</i>". Inicialmente, o professor deverá clicar nessa opção;</li> <li>2. O sistema demandará ao professor escolher e inserir no sistema: o número "c" de covas do tabuleiro, que pode variar entre 4, 5 ou 6; o número "s" de sementes por cova no início da partida, que pode variar entre 4, 5 ou 6; e a definição do término das partidas, que, por sua vez, possui como variáveis a quantidade "x" de sementes restantes no</li> </ol>

---

tabuleiro, a ser detalhada mais adiante, e a quantidade “ $y$ ” de rodadas consecutivas sem que haja colheita de sementes, que pode variar de 3 a 10 rodadas.

- a. Em relação aos possíveis valores de “ $x$ ”, consideraremos as duas possibilidades seguintes:
  - i. o jogo é finalizado quando o número de sementes do tabuleiro impossibilitar a colheita de sementes:
    - $x$  pode variar entre 0, 1 ou 2, a depender da quantidade de covas e de sementes por cova configuradas.
  - ii. o jogo é finalizado quando o tabuleiro contiver o menor número de sementes para a possibilidade de colheita, ou seja, 3 a mais do que no caso anterior:
    - $x$  pode variar entre 3, 4 ou 5, a depender da quantidade de covas e de sementes por cova configuradas.

---

Fonte: elaborado pelo autor.

Neste momento, os valores das variáveis foram escolhidos intuitivamente, ao considerar a manutenção do aspecto lúdico do jogo e uma limitação na quantidade de sementes, para favorecer a presença de posições de jogo que permitam colheitas de sementes. Destacamos que nosso objetivo com esse caso de uso é de permitir que o professor possua alternativas de modificações do jogo e, com isso, potencializar o jogo didaticamente.

#### 4.1.1.8 Arranjar alunos nas partidas *online*

Este caso de uso representa mais um tipo de controle que o professor realiza sob as partidas *online* dos alunos. O arranjo dos alunos se refere aos modelos de interação entre os sujeitos nas partidas do jogo.

Como veremos no quadro a seguir, o professor conduzirá os alunos a realizarem partidas *online* tanto individualmente, em que jogam “um aluno contra um aluno”, quanto em duplas ou casos adaptáveis, para permitir a interação de jogadores em uma mesma equipe. Essa interação possibilita o uso de linguagem no compartilhamento de estratégias de jogo e, deste modo,

favorece a vivência de situações de *formulação* e *validação* de conhecimentos matemáticos suscetíveis de serem explorados por meio do jogo, sob o olhar da Teoria das Situações Didáticas, de Guy Brousseau.

Quadro 13 – Caso de uso arranjando alunos nas partidas online

<b><i>Arranjar alunos nas partidas online</i></b>	
<b><i>Descrição</i></b>	Função em que o professor seleciona a maneira como os alunos irão se distribuir para a realização das partidas com computadores distintos: "1 aluno contra 1 aluno"; "2 alunos contra 2 alunos"; e casos adaptáveis.
<b><i>Ator</i></b>	Professor.
<b><i>Entradas ou precondições</i></b>	Os usuários aluno e professor precisam estar logados no sistema e o professor deverá ter realizado o Caso de Uso "Configurar modelo de jogo online".
<b><i>Saídas ou pós-condições</i></b>	Se o caso de uso for realizado com sucesso, o sistema fornecerá aos alunos uma mensagem de indicação sobre a permanência em sua máquina, para jogar individualmente contra outro aluno, ou se ele precisará se juntar a outro(s) colega(s), ambos em uma mesma máquina, para a realização de partidas do tipo “dupla contra dupla” ou “equipe contra equipe”.
<b><i>Fluxo de eventos principal</i></b>	<ol style="list-style-type: none"> <li>1. O professor terá acesso a lista de alunos logados e deverá demandar ao sistema o sorteio dos alunos em:           <ol style="list-style-type: none"> <li>a. grupos de 2 adversários para as partidas de um aluno contra outro, em computadores distintos.               <ol style="list-style-type: none"> <li>i. Caso o número de alunos seja ímpar, o sistema formará uma dupla de alunos, para jogarem juntos contra um dos alunos.</li> </ol> </li> <li>b. grupos de 4 alunos, organizados por pares, de modo a definir os adversários em partidas em duplas (2 alunos contra 2 alunos, em dois computadores distintos, com cada dupla dividindo o mesmo computador).               <ol style="list-style-type: none"> <li>i. Caso o número de alunos não seja múltiplo de 4, o sistema formará um ou mais trios de alunos, de modo a sempre</li> </ol> </li> </ol> </li> </ol>

---

propiciar a presença de interações entre companheiros de jogo.

2. Após o sorteio realizado pelo sistema, o professor terá a opção de editar manualmente os integrantes de cada uma das partidas.
- 

Fonte: elaborado pelo autor.

Esse Caso de Uso representa o atendimento de um dos encaminhamentos para futuras pesquisas envolvendo o Mankala Colhe Três apresentadas por Santos (2014), voltado à possibilidade de o professor explorar distintos modelos de interação entre os sujeitos de maneira prática e eficiente. Esse aspecto potencializa o papel didático do jogo no sentido que permite um controle melhor pelo professor em termos de análise da evolução das situações pelos alunos, através do favorecimento da criação do meio para a vivência de situações de *ação, formulação, validação* e, posteriormente, para um ambiente propício à *institucionalização* das situações.

#### 4.1.1.9 Acessar relatório de jogadas

O caso de uso *acessar relatório de jogadas* se destina ao professor e tem o objetivo de permitir que todas as jogadas realizadas pelos alunos sejam registradas pelo sistema. Esse registro fornece ao professor informações importantes para acompanhamento e controle da produção dos alunos para posteriores análises. Veremos, no quadro a seguir, os detalhes dessa funcionalidade do sistema:

Quadro 14 – Caso de uso realizar relatório de jogadas

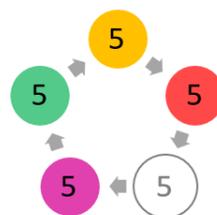
<b><i>Acessar relatório de jogadas</i></b>	
<b><i>Descrição</i></b>	Função em que o professor acessa os relatórios das jogadas dos alunos em partidas <i>online</i> .
<b><i>Ator</i></b>	Professor.
<b><i>Entradas ou pré-condições</i></b>	O professor deverá ter feito o <i>login</i> no sistema.
<b><i>Saídas ou pós-condições</i></b>	O sistema deverá fornecer, para visualização e <i>download</i> , o relatório de jogadas dos alunos “por partidas” e “por alunos”.

---

***Fluxo de eventos principal***

1. Ao realizar o *login* do sistema, o professor terá, dentre as opções disponíveis no menu principal, a função "Acessar relatório de jogadas". Inicialmente, o professor deverá clicar nessa opção;
2. O sistema demandará ao professor escolher se deseja acessar o relatório de uma partida realizada pelos estudantes ou se deseja acessar o relatório de um aluno;
3. Em qualquer um dos casos, o sistema fornecerá a sequência ordenada das jogadas, representadas por meio de representações do tabuleiro, como a que segue:

Figura 16 – Representação de posição de jogo do Mankala Colhe Três Digital no relatório de jogadas



Fonte: elaborada pelo autor

em que cada círculo representaria uma cova do tabuleiro do jogo e cada número representa a quantidade de sementes presentes nas covas.

Fonte: elaborado pelo autor.

Santos (2014) afirma que, ao desenvolver sua pesquisa, precisou realizar a transcrição de todas as partidas realizadas pelos sujeitos participantes do seu experimento para ser capaz de analisar a mobilização dos conhecimentos matemáticos mobilizados por eles e a evolução das estratégias dos alunos. Considerando que o pesquisador realizou seu experimento com apenas quatro sujeitos e em apenas dois momentos práticos de realização de partidas de jogo, percebemos que esse tipo de registro de jogo se torna inviável, se não considerarmos as possibilidades advindas da informática. Sendo assim, podemos afirmar que a funcionalidade de registro automático das jogadas representa um importante avanço em termos do potencial didático do Mankala Colhe Três Digital.

#### 4.1.1.10 Configurar ficha de atividades

Conforme comentamos na seção referente ao caso de uso *resolver ficha de atividades*, para que o aluno possa resolver uma ficha de atividades no próprio sistema do jogo, o professor deverá configurar antecipadamente essa ficha. Essa configuração consiste em o professor selecionar valores para representar as quantidades de sementes em covas do tabuleiro, representando posições possíveis de jogo.

Apresentamos abaixo o quadro do caso de uso *configurar ficha de atividades* e, logo após, traremos um exemplo de ficha de atividades possível de ser configurada por um professor, utilizada por Santos (2014) de forma física (impressa em papel) e que guiou a idealização de nossa ficha digital configurável.

Quadro 15 – Configurar ficha de atividades

<b><i>Configurar ficha de atividades</i></b>	
<b><i>Descrição</i></b>	<p>Função em que o professor realiza a edição da ficha de atividades a ser disponibilizada aos alunos. A ficha de atividades se baseia em quatro categorias de atividades, relacionadas a situações de jogo no Mankala Colhe Três Digital, como discriminado a seguir:</p> <ol style="list-style-type: none"> <li>1. Realização de jogadas favoráveis a partir de posições específicas do jogo;</li> <li>2. Argumentação a respeito de escolhas de jogadas feitas por um hipotético parceiro de jogo a partir de posições específicas do jogo;</li> <li>3. Mapeamento das possibilidades de jogadas a partir de uma posição específica do jogo;</li> <li>4. Argumentação a respeito de noções da Matemática que acreditam que foram mobilizadas durante as situações de jogo.</li> </ol>
<b><i>Ator</i></b>	Professor.
<b><i>Entradas ou pré-condições</i></b>	O professor deverá ter realizado o <i>login</i> no sistema.
<b><i>Saídas ou pós-condições</i></b>	Após a realização do Caso de Uso, o sistema deverá disponibilizar, aos alunos, a consulta e a resolução das atividades.

---

***Fluxo de eventos principal***

1. Ao realizar o *login* do sistema, o professor terá, dentre as opções disponíveis no menu principal, a função "Configurar ficha de atividades". Inicialmente, o professor deverá clicar nessa opção;
2. O sistema demandará ao professor a configuração do modelo de jogo (número de covas, sementes e regras de finalização de jogo) e a escolha das posições de jogo que estarão disponíveis em cada uma das atividades, de modo a permitir a visualização e a resolução pelos alunos.
  - a. Para auxiliar essa escolha, o sistema fornecerá alguns exemplos de posições de jogo classificadas pela quantidade de opções de colheita de sementes na presente jogada e da quantidade de possibilidades de colheitas na jogada seguinte, segundo as regras e configuração atual do tabuleiro, de modo a permitir o trabalho com a realização de jogadas favoráveis.

---

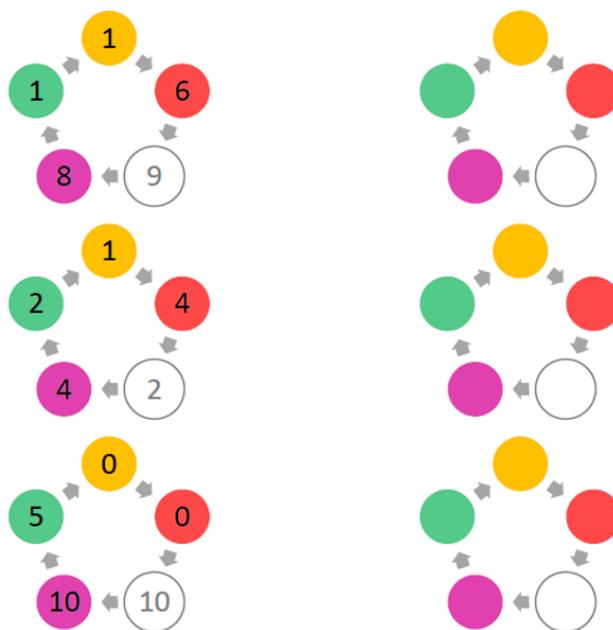
Fonte: elaborado pelo autor.

Apresentaremos, a seguir, como exemplo de ficha de atividades configurada pelo professor, a ficha física utilizada por Santos (2014) em sua pesquisa, atividade por atividade. A configuração da ficha de atividades consiste em o professor selecionar as posições de jogo que serão apresentadas nas atividades da ficha, ou seja, o professor deverá determinar o número de sementes por cada cova nas representações do tabuleiro do jogo. É importante destacar que o sistema também permitirá a configuração do modelo de jogo (número de covas, de sementes por cova e regras de finalização de jogo) para a configuração das atividades.

Segue um recorte da atividade 1 da ficha de atividades utilizada por Santos (2014):

Figura 17 – Exemplo da atividade 1 da ficha de atividades

1. Partindo de cada uma das seguintes configurações do tabuleiro do *Mankala Colhe Três*, realize a jogada que você acha mais favorável.



Fonte: Santos (2014, p. 196).

A configuração dessa primeira atividade se dá pela escolha das posições de jogo nas representações do tabuleiro da esquerda. No exemplo acima, podemos perceber que a primeira posição de jogo permite apenas uma maneira de colheita e sementes, que é a partir da remoção das 6 sementes da cova vermelha e sua redistribuição de duas em duas, resultando em 3 sementes na última cova da redistribuição (verde).

Por meio de uma análise similar a realizada na primeira posição de jogo, percebemos que a segunda posição permite quatro opções de colheita: (1) a partir da branca, redistribuindo de uma em uma; (2) a partir da roxa, redistribuindo de uma em uma; (3) a partir da roxa, redistribuindo de duas em duas; (4) a partir da verde, redistribuindo de duas em duas. Do mesmo modo, podemos concluir que a terceira posição não permite colheita de sementes.

Santos (2014) escolheu as posições de jogo da Figura 16 com o intuito de perceber o poder de mapeamento de possibilidades de jogadas dos sujeitos participantes de sua pesquisa, que se atrelava ao nível de evolução de estratégia de cada um deles. É importante lembrar que, para Santos (2014), *jogadas favoráveis* são determinadas a partir da análise da possibilidade de colheita na presente posição de jogo e pela consideração das possibilidades de jogada pelo

adversário na jogada seguinte. Sendo assim, na atividade apresentada, podemos destacar, por exemplo, que, como a segunda configuração permite várias possibilidades de colheita, os alunos com maior poder de mapeamento das possibilidades de jogo tenderiam a escolher uma jogada que colhesse sementes e que, na jogada seguinte, não permitisse colheita pelo adversário.

Ao considerar a configuração da ficha de atividades pelo professor, essa primeira atividade permitirá uma análise mais precisa e diversificada das estratégias dos alunos.

A seguir, apresentaremos o exemplo da segunda atividade da ficha de atividades a ser configurável pelo professor:

Figura 18 – Exemplo da atividade 2 da ficha de atividades

2. Imaginemos que estamos jogando em duplas e que sou seu companheiro de equipe. Partindo da configuração da esquerda, realizei um tipo de jogada, resultando no tabuleiro à direita. O que você me diria? Concorde ou discorde? Por quê?

a. Removi as sementes da cova branca e as redistribuí de uma em uma.



b. Removi as sementes da cova roxa e as redistribuí de duas em duas.



c. Removi as sementes da cova amarela e as redistribuí de uma em uma.



Fonte: Santos (2014, p. 196-197).

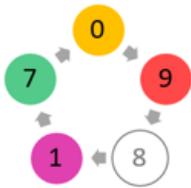
Nessa segunda atividade, há uma simulação de situações interativas entre os alunos para a tomada de decisão nas jogadas, sempre relacionada ao mapeamento de possibilidades de jogadas favoráveis. Esse tipo de atividade exige a realização de argumentações mais explícitas

sobre estratégias de jogo e, deste modo, fornece ao professor informações importantes acerca da mobilização e consolidação dos conhecimentos matemáticos envolvidos.

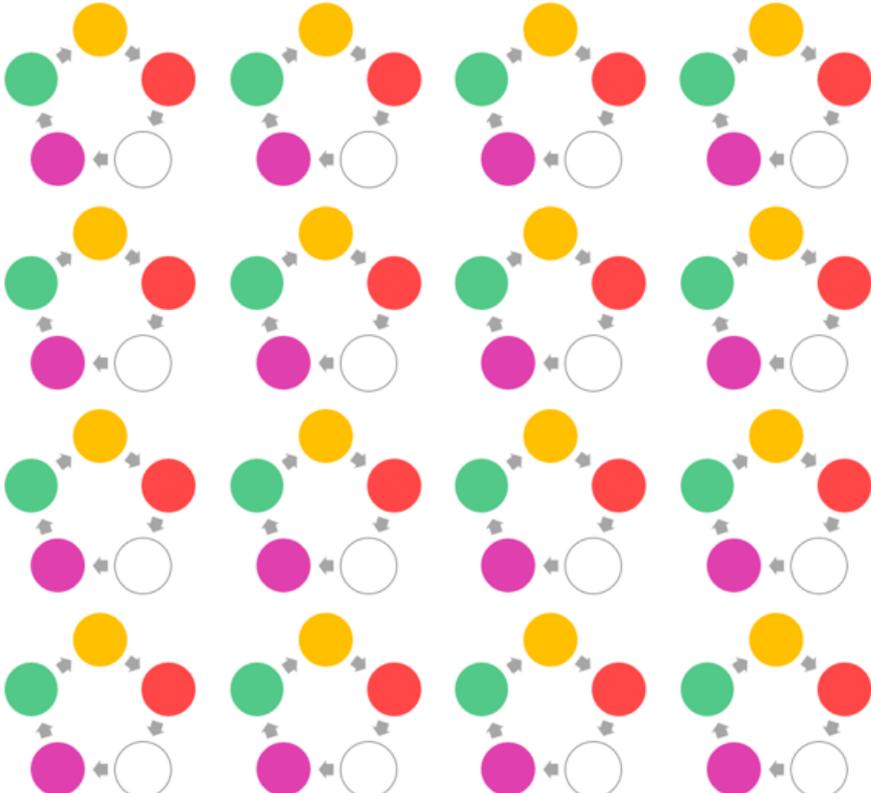
A terceira atividade da ficha de atividades foca mais diretamente no nível de mapeamento de possibilidades de jogadas a partir de uma posição de jogo, como ilustrado na figura a seguir:

Figura 19 – Exemplo da atividade 3 da ficha de atividades

3. Partindo da seguinte configuração do tabuleiro, apresente todas as possibilidades de jogadas que podem ser realizadas (anexo).



**Anexo do teste:**



Fonte: Adaptado de Santos (2014, p. 197-198).

Nas três atividades apresentadas pelos exemplos acima, é importante destacar que, assim como informado no último item do quadro referente ao caso de uso em foco, a escolha das posições de jogo para inserção na ficha na sua versão digital será auxiliada por meio de sugestões fornecidas pelo sistema, categorizados pela quantidade de maneiras de colheitas na presente jogada e da quantidade de possibilidades de colheita na jogada seguinte.

Para finalizar a apresentação do exemplo de ficha de atividades, apresentaremos a atividade 4, que poderá ser apresentada conforme proposição de Santos (2014) ou, caso o professor ache pertinente, a pergunta poderá ser adaptada ao seu critério. Segue o recorte da atividade 4 proposta por Santos (2014):

Figura 20 – Exemplo da atividade 4 da ficha de atividades

4. Que conhecimentos matemáticos você acha que utilizou enquanto jogava o *Mankala Colhe Três*? Em que momentos?

Fonte: Santos (2014, p. 197).

Esse é um tipo de atividade que auxilia o professor na sistematização da aprendizagem dos alunos, de modo a trazer um status de saber ao que foi explorado e mobilizado pelos alunos nas situações envolvendo a realização de partidas no Mankala Colhe Três Digital.

#### 4.1.1.11 Acessar respostas das atividades

O caso de uso *acessar respostas das atividades* é mais uma funcionalidade específica do *professor* que, conforme pode ser observado no quadro a seguir, terá o acesso às resoluções das atividades dos alunos direto do sistema.

Quadro 16 – Acessar respostas das atividades

<b><i>Acessar respostas das atividades</i></b>	
<b><i>Descrição</i></b>	Função em que o professor acessa as respostas das fichas de atividades dos alunos.
<b><i>Ator</i></b>	Professor.

---

<b><i>Entradas ou condições</i></b>	O professor deve ter feito login no sistema.
<b><i>Saídas ou pós-condições</i></b>	O sistema deverá disponibilizar a visualização e o <i>download</i> das respostas dos alunos, categorizados por aluno e por atividade.
<b><i>Fluxo de eventos principal</i></b>	<ol style="list-style-type: none"> <li>1. Ao realizar o <i>login</i> do sistema, o professor terá, dentre as opções disponíveis no menu principal, a função "Acessar respostas das atividades". Inicialmente, o professor deverá clicar nessa opção;</li> <li>2. O sistema permitirá ao professor o acesso às respostas das atividades dos alunos, categorizados por atividade e por aluno, para visualização e <i>download</i>.</li> </ol>

---

Fonte: elaborado pelo autor.

Assim como acontece no caso de uso *acessar relatório de jogadas*, o acesso automático às respostas dos alunos representa uma funcionalidade muito importante da versão digital do Mankala Colhe Três, se comparado à versão física do jogo, principalmente na viabilização de um acompanhamento mais rápido e preciso da produção dos alunos.

#### **4.1.2. Requisitos não funcionais do Mankala Colhe Três Digital**

Nesta seção apresentaremos os requisitos não funcionais do Mankala Colhe Três Digital. Sobre esse tipo de requisito, Brito (2010) afirma que

Os requisitos não-funcionais são os atributos de qualidade que um sistema deve ter ou determinam quão bem ele deve realizar uma determinada função. Os atributos de qualidade tornam um sistema atrativo, seguro, usável ou até mesmo rápido em seu uso. Requisitos não-funcionais possuem uma grande importância na fase inicial de levantamento bem como no desenvolvimento de software. Eles possuem um papel relevante para que um sistema atenda às necessidades do cliente, pois definem restrições ou atributos de qualidade desejáveis em um software (BRITO, 2010, p. 49).

Os requisitos não funcionais do Mankala Colhe Três Digital estão categorizados em três seções: Necessidade de outros sistemas (Banco de Dados e Sistema Operacional); Requisitos de Processo (padrões, implementação); e Requisitos de Produto (usabilidade, confiabilidade, desempenho e segurança). É importante destacar que, como os requisitos não funcionais também estão fortemente atrelados à etapa de Desenvolvimento do software na EDI, modificações em tais requisitos são bastantes suscetíveis em etapas posteriores do projeto.

#### 4.2.1.1 Necessidade de outros sistemas

Segue o quadro dos requisitos não funcionais relacionados à necessidade de outros sistemas:

Quadro 17 – Requisito não funcional: necessidade de outros sistemas

<i>Necessidade de outros sistemas</i>	
<b>Descrição</b>	<p>Estes requisitos indicam que o sistema necessita de outros softwares para poder executar as suas tarefas. Os sistemas necessários são:</p> <p><b>4.1.1 Banco de Dados (BD):</b> Será necessário um Banco de Dados (BD) para armazenar:</p> <ol style="list-style-type: none"> <li>i. as informações do jogo (regras, história dos jogos do tipo mankala etc.);</li> <li>ii. as informações sobre os usuários (nome, <i>login</i>, senha e hierarquia de acesso ao sistema);</li> <li>iii. uma base de posições de jogo, que serão utilizadas tanto para a realização das partidas <i>offline</i> quanto como referência para a configuração das fichas de atividades;</li> <li>iv. as respostas das atividades dos alunos, para posterior consulta pelo Professor;</li> <li>v. as jogadas realizadas pelos alunos.</li> </ol> <p>O BD tem que oferecer flexibilidade, portabilidade e os melhores recursos, mas como a aplicação é simples pode-se usar um BD relacional.</p> <p><b>4.1.2 Sistema Operacional (SO):</b> O sistema precisará de um suporte de um SO. Como um outro requisito é portabilidade, pode-se usar as marcas mais conhecidas no mercado Windows, Linux, Android, iOS ou MacOS.</p>

---

Fonte: elaborado pelo autor.

#### 4.2.1.2 Requisitos de processo

No quadro abaixo, estão agrupados e detalhados os requisitos não funcionais relativos ao processo de desenvolvimento do Mankala Colhe Três Digital, que incluem os padrões a serem utilizados e os requisitos relacionados à implementação do sistema:

Quadro 18 – Requisito não funcional: requisitos de processo

<b><i>Requisitos de processo</i></b>	
<b><i>Descrição</i></b>	Os requisitos de processo estão relacionados ao processo de desenvolvimento do sistema.
<b><i>Padrões</i></b>	Utilização do processo RUP como o processo para o desenvolvimento do sistema e os padrões de concepção Entidade Controle Fronteira - ECF, para a identificação e alinhamento de cada elemento participante do comportamento do sistema.
<b><i>Implementação</i></b>	Esta seção descreve os requisitos não funcionais associados à implementação do aplicativo. <ol style="list-style-type: none"> <li>i. O sistema será desenvolvido como uma aplicação web;</li> <li>ii. O sistema será desenvolvido em HTML5 e Javascript para a camada de visão e controle, e a camada de modelo será desenvolvida em PHP;</li> <li>iii. Será utilizado o Zend Framework para dar suporte ao desenvolvimento no padrão MVC;</li> <li>iv. O banco de dados que suportará o ambiente será o PostgreSQL.</li> <li>v. O sistema deverá ser implantado em um servidor Web, rodando o Apache 1.8 ou versão superior;</li> <li>vi. As estações clientes deverão executar o sistema através dos browsers Microsoft Internet Explorer, Mozilla Firefox ou Google Chrome.</li> </ol>

Fonte: elaborado pelo autor.

#### 4.2.1.3 Requisitos de produto

Os requisitos não funcionais de produto do sistema reúnem os seguintes aspectos: usabilidade, confiabilidade, desempenho e segurança. Segue o quadro dos *requisitos de produto*:

Quadro 19 – Requisito não funcional: requisitos de produto

<b><i>Requisitos de produto</i></b>	
<b><i>Descrição</i></b>	Os requisitos de produto estão relacionados às características desejadas que o sistema deve ter.
<b><i>Usabilidade</i></b>	<p>Esta seção descreve os requisitos não funcionais associados à facilidade de uso da interface com o usuário, material de treinamento e documentação do aplicativo.</p> <ol style="list-style-type: none"> <li>i. As mensagens de erro do sistema deverão ser precisas e construtivas, fazendo com que o usuário identifique sua origem e como proceder após sua ocorrência;</li> <li>ii. A interface do sistema deverá ser agradável e objetiva, ou seja, suas funcionalidades e informações deverão ser bem intuitivas;</li> <li>iii. Os usuários Professores, após curto tempo de treinamento, devem ser capazes de operar o sistema, diminuindo a necessidade de consultas ao sistema de suporte, para a execução de suas tarefas.</li> </ol>
<b><i>Confiabilidade</i></b>	<p>Esta seção descreve os requisitos não funcionais associados à frequência, severidade de falhas do sistema e habilidade de recuperação delas, bem como à corretude do aplicativo.</p> <ol style="list-style-type: none"> <li>i. O sistema deve estar sempre disponível;</li> <li>ii. Os dados não podem ser corrompidos;</li> <li>iii. Os dados serão mantidos e gerenciados por um Sistema de Gerenciamento de Banco de Dados (SGBD);</li> <li>iv. A base de dados deve estar sempre íntegra;</li> <li>v. Os dados que o usuário insere no sistema deverão ser validados a fim de evitar que dados errôneos sejam armazenados prejudicando a corretude e consistência da base de dados;</li> <li>vi. O histórico do uso do sistema deverá ser salvo no banco de dados.</li> </ol>
<b><i>Desempenho</i></b>	Esta seção descreve os requisitos não funcionais associados à eficiência, uso de recursos e tempo de resposta do aplicativo.

- 
- i. O sistema deve ter um tempo de resposta de, no máximo 5 segundos;
  - ii. O sistema deverá suportar no mínimo, 100 acessos simultâneos.
- 

### Segurança

Esta seção descreve os requisitos não funcionais associados à integridade, privacidade e autenticidade dos dados do aplicativo.

- i. O usuário será autenticado no aplicativo através de senha;
  - ii. O aplicativo deve garantir que os dados estão protegidos de acessos não autorizados.
- 

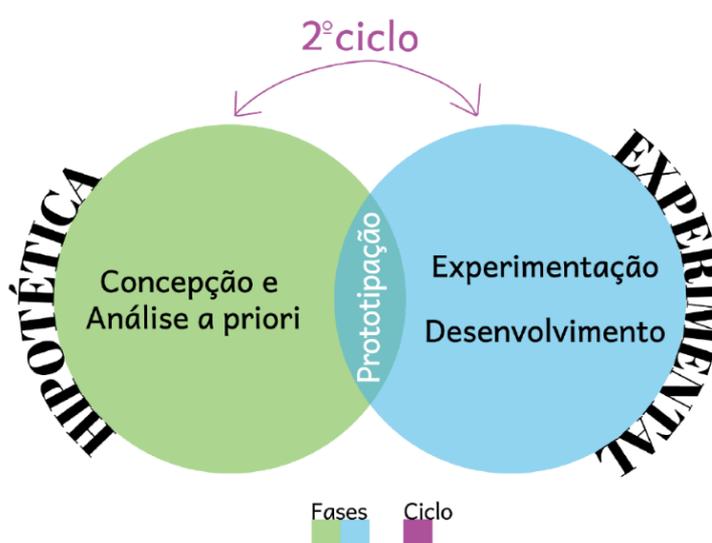
Fonte: elaborado pelo autor.

Apresentaremos, a seguir, as discussões acerca da etapa e Prototipação do Mankala Colhe Três Digital.

## 4.2 PROTOTIPAÇÃO

A etapa de prototipação da Engenharia Didático-Informática representa umas das etapas de transição entre fases, compondo o ciclo *Hipótese-Experimental* da EDI. Segue uma ilustração que fornece informações acerca desse ciclo:

Figura 21 – Representação do segundo ciclo do modelo de processo da EDI



Fonte: Tiburcio (2020, p. 174).

Discorreremos sobre características que a etapa de prototipação possui oriundas tanto da fase Hipotética, caracterizada pela realização da etapa de Concepção e Análise *a priori*, quanto da fase Experimental, composta pelo Desenvolvimento e pela Experimentação.

Podemos iniciar essa discussão a partir das relações existentes entre as etapas de Concepção e Análise *a priori* e a etapa de Prototipação. Segue uma citação de Tiburcio que nos ajuda a compreender as articulações entre etapas:

Na *concepção e análise a priori*, são desenvolvidas as situações de utilização do software, os referenciais teóricos e metodológicos, elencados pela equipe, que, em hipótese, auxiliam no ensino e na aprendizagem dos elementos a serem trabalhados, são considerados e as interações com os usuários, a serem desenvolvidas, devem estar fundamentadas nesse levantamento. Essas situações devem ser criadas ao mesmo tempo em que o protótipo do software é idealizado, visto que a prototipação serve para que os profissionais de design e arquitetura de software compreendam os requisitos e funcionalidades e consigam executá-las com o que se dispõe de hardware e software (TIBURCIO, 2020, p. 174).

Em relação ao andamento da nossa pesquisa, no que se refere à realização das etapas do modelo de processo da EDI para a concepção do Mankala Colhe Três Digital, podemos perceber que a produção do Modelo de Casos de Uso nos fornece elementos bastante detalhados do software que idealizamos e que servirão de referência para o desenvolvimento do protótipo do jogo, principalmente nos fluxos de cada caso de uso.

Ainda sobre as características “hipotéticas” da etapa de Prototipação, Tiburcio apresenta uma indicação para a sistematização das idealizações do software, como pode ser visto na citação a seguir:

Indicamos que a prototipação seja feita “em telas” – os líderes da equipe, de posse dos requisitos, podem utilizar editores de textos ou imagens para simular como será o ambiente a ser desenvolvido, com as funcionalidades, botões, menus etc. Assim exibindo as telas iniciais do produto e como algumas de suas funções a serem executadas (TIBURCIO, 2020, p. 175).

Quando Tiburcio apresenta a expressão “prototipação feita em telas”, na nossa compreensão, significa representar em telas as idealizações do software, advindas da etapa de Concepção e Análise *a priori*, tendo em vista que consideraremos como *protótipo* uma versão inicial do sistema, mas que seja minimamente implementada e passível de utilizações, mesmo que em cenários de testes, pela equipe de desenvolvimento (professores e/ou alunos).

A partir do exposto, podemos iniciar a discussão sobre as relações existentes entre as etapas de Prototipação e a etapa de Desenvolvimento, que compõem a fase Experimental da EDI, por meio da seguinte afirmação de Tiburcio: “com as ideias do protótipo fundamentadas,

é dado início ao design e arquitetura do software, o *desenvolvimento*” (TIBURCIO, 2020, p. 175).

É possível perceber que, enquanto a etapa de Concepção e Análise *a priori* fornece uma idealização de uma primeira versão software a ser desenvolvido (protótipo), a etapa de desenvolvimento é responsável por construir computacionalmente esse protótipo, por meio do design e da arquitetura do software.

Sendo assim, consideraremos a realização da Prototipação na nossa pesquisa pelas duas perspectivas acima citadas: a da Concepção e Análise *a priori*, por meio do detalhamento das funcionalidades do sistema, que, no nosso caso, está presente no Modelo de Casos de Uso do jogo; e a do Desenvolvimento, que representa a construção de uma versão inicial do sistema, a ser discutida na próxima seção deste trabalho.

### 4.3 DESENVOLVIMENTO

Como afirmamos na seção anterior, as etapas de Prototipação e Desenvolvimento da EDI se relacionam fortemente, tendo em vista que o Desenvolvimento trata justamente da elaboração efetiva do protótipo, que representa uma versão inicial do software a ser produzido.

Os resultados esperados para a etapa de Desenvolvimento, segundo Tiburcio (2020), são os componentes internos (especificações de sistemas operacionais, linguagem de programação, definição de hardware etc.) e os componentes externos (interfaces, layout, objetos, relações entre objetos, funcionalidades etc.) do software.

Além dessa articulação, o Desenvolvimento também se relaciona com outra etapa do modelo de processo da EDI, que é a etapa de Experimentação, como podemos verificar por meio da citação a seguir:

Consideramos o ciclo hipotético-experimental visto que o protótipo do software está na interseção entre o que se considera como hipótese e o que será desenvolvido, uma vez que há a expectativa de que o produto atenda aos objetivos levantados pela equipe. Dessa forma, as situações que foram idealizadas são implementadas no software (ou não) ao passo que a experimentação acontece (TIBURCIO, 2020, p. 175).

Segundo o autor, a etapa de Desenvolvimento ocorre de forma simultânea a Experimentação. Ainda sobre essa característica da fase experimental da EDI, Tibúrcio complementa afirmando que

Os testes iniciais, o feedback da utilização, as percepções dos membros da equipe, as compreensões e incompreensões das funcionalidades, tudo isso acontece de forma simultânea à elaboração do produto. O software deve ser criado considerando a validação e invalidação das hipóteses e requisitos, ambos testados em caráter experimental (TIBURCIO, 2020, p. 175).

É preciso refletir sobre o significado de “experimentação” no modelo de processo da Engenharia Didático-Informática, pois, como vimos nas citações anteriores, uma possível compreensão para essa expressão seria o conjunto de testes do software pela equipe de desenvolvimento durante o processo de construção desse produto.

Há outra compreensão para a Experimentação, também apresentada no modelo de processo da EDI, que é fortemente atrelada ao significado dessa expressão na Engenharia Didática (ARTIGUE, 1996), que é o de colocar o software em situações reais de utilização com potenciais usuários do produto, que vai além dos testes pela equipe de desenvolvimento ou de utilizações em ambientes simulados. Segue uma citação de Tiburcio enfocando esse significado atribuído à Experimentação, na abordagem referente ao início do ciclo *experimental-operacional*:

Este ciclo contempla a *experimentação* que é clássica, no sentido de colocar o software em desenvolvimento em situações reais de utilização, não considerando apenas os testes realizados pela equipe ou o ambiente de utilização simulado, mas com potenciais usuários do produto. Coloca-se em funcionamento o que foi construído, caso haja exigências de alterações, modificações e diversas outras implementações, devem ser realizadas ao passo que se experimenta (TIBURCIO, 2020, p. 177).

Nesse segundo sentido, a Experimentação representa, no percurso metodológico sobre as situações de uso do recurso, a etapa que une as hipóteses e idealizações oriundas da Análise *a priori* com os resultados obtidos na Análise *a posteriori* dessa experimentação.

Ao analisar as utilizações do termo “experimentação” na EDI, acreditamos que é preciso tomar certo cuidado com o que consideramos ser uma ambiguidade, pois, na nossa compreensão, considerar a realização de modificações no software ao passo que se experimenta pelos reais usuários em situação de análise do recurso, no segundo sentido apresentado para a experimentação, de certo modo, compromete o que se espera de resultados da Análise *a posteriori* e, conseqüentemente, fragiliza a robustez presente no processo de Validação, fundamentalmente interno na Engenharia Didática e na EDI.

Com o objetivo de sistematizar o que trouxemos nos parágrafos anteriores, reforçamos que compreendemos a necessidade de que haja modificações no software durante o uso de suas versões iniciais, caracterizando a “experimentação” como uma espécie de testes do software e,

por outro lado e não menos importante, consideramos esse tipo de experimentação como distinto do que considera o uso do software como recurso educativo, já relativamente “validado”, a ser utilizado em situações educativas reais, cujo foco é a busca pela validação das situações de ensino e aprendizagem que se utilizaram do software educativo como um elemento do *meio*, preparado pelo professor. Nesse segundo sentido, a validação, a ser realizada internamente, por meio do confronto entre a Análise *a priori* e a Análise *a posteriori*, traz o olhar às situações envolvendo o recurso, e não ao recurso propriamente dito.

Complementando o que foi dito no parágrafo acima, poderíamos ilustrar essa distinção entre os significados de experimentação se imaginássemos a seguinte situação: ao “experimentar” um software educativo em produção por meio da EDI em um ambiente real de ensino, nos deparássemos com algum insucesso em termos de objetivos de aprendizagem esperados pelos alunos, seria preciso saber distinguir se o que ocasionou essa avaliação negativa se referiu a necessidade de ajustes no software – experimentação como testes do recurso – ou no modo como o educador planejou seu uso – experimentação no sentido mais próximo ao da Engenharia Didática. Por isso, pensamos que é importante compreender e distinguir esses dois olhares da experimentação na EDI.

No Capítulo 6 da presente pesquisa traremos análises do uso da EDI e propostas de reconsiderações de suas etapas, similarmente ao exposto no parágrafo anterior, em que apresentaremos um modelo de processo de software da EDI destinado a jogos matemáticos de simulação para situações didáticas, como já comentado na introdução deste trabalho.

Retomando a discussão a respeito da etapa de Desenvolvimento, vimos que ela é caracterizada pelo design e a arquitetura do software e que produz, como resultados, os componentes internos e externos do sistema. Partindo disso, gostaríamos de iniciar uma problemática, já abordada na seção referente à Especificação do software, que gira em torno da composição da equipe de desenvolvimento e sua perspectiva transdisciplinar, essencial na produção de softwares educativos pela EDI.

Considerando que as atividades de arquitetura e de design de software pertencem ao domínio da Engenharia de Software, acreditamos que é fundamental que a equipe de desenvolvedores do software educativo, proposta pela EDI, contenha um profissional dessa área ou que, como sugerimos na seção sobre a Especificação, haja profissionais na equipe que possua competência para discutir temas de diversas áreas envolvidas no projeto com propriedade e de maneira integrativa, inclusive acerca da Engenharia de Software.

Um primeiro desafio que gostaríamos de destacar para a composição dessa equipe de desenvolvimento na perspectiva transdisciplinar é que nem sempre é possível conseguir um

profissional da Engenharia de Softwares disponível nos ambientes em que temos situado as pesquisas envolvendo a Engenharia Didático-Informática, além de que, nessa perspectiva, esse profissional ainda precisaria possuir um perfil que permitisse o diálogo direto e integrado às temáticas envolvendo o ensino e a aprendizagem.

Podemos situar os desenvolvimentos de softwares educativos que se utilizaram dos princípios da Engenharia Didático-Informática, ou que conduziram à formalização atual desses princípios metodológicos, sempre em ambientes de pesquisas científicas inseridas em instituições públicas de ensino superior e por meio de pesquisadores da área de Educação Matemática.

A dificuldade de contar com a disponibilidade de participação e permanência de profissionais de diversas áreas de forma integrada à equipe de desenvolvimento proposta pela EDI é uma realidade, assim como acreditamos ser bastante difícil dispormos de profissionais que possuam o perfil profissional que permita esse diálogo necessário entre o que preconiza a Educação Matemática e a Engenharia de Software.

Diante desse cenário, acreditamos que a difusão e utilização ampla da Engenharia Didático-Informática podem estar sendo bastante dificultadas por carência de recursos humanos, financeiros e materiais dos educadores matemáticos que pretendem produzir softwares educativos de qualidade, principalmente no que se refere às questões de composição da equipe transdisciplinar para a etapa de Desenvolvimento.

Nessa perspectiva, refletimos sobre a possibilidade de propor uma contribuição metodológica à etapa de Desenvolvimento da EDI, no caso particular dos jogos matemáticos de simulação para situações didáticas, tendo em vista que nossa pesquisa possui como objetivo a proposição de um modelo de processo de software a ser construído a partir das análises de uso da EDI em sua versão atual na concepção do Mankala Colhe Três.

Nossa intenção com essa reflexão é buscar trazer mais detalhes acerca da fase de Desenvolvimento da EDI para os educadores matemáticos interessados em produzir jogos matemáticos de simulação para situações didáticas. Essa contribuição metodológica para a fase de Desenvolvimento foi idealizada com o objetivo de auxiliar o diálogo entre educadores matemáticos e desenvolvedores de software sem que haja prejuízos consideráveis em termos da qualidade do software educativo a ser produzido.

Essa nova problemática da nossa pesquisa nos remeteu a um novo objetivo de pesquisa: *construir uma metodologia para a etapa de Desenvolvimento da EDI que forneça um percurso para transformação dos requisitos em código a ser implementado no desenvolvimento de jogos matemáticos de simulação para situações didáticas.* É importante destacar que esse novo

objetivo faz parte do que preconiza um dos nossos Objetivos Específicos da pesquisa, que é o de “apresentar contribuições metodológicas para a Engenharia Didático-Informática”, explicitado no Capítulo 1 deste trabalho.

A criação desse novo objetivo de pesquisa está diretamente articulada à motivação advinda da produção do Modelo de Casos de Uso do Mankala Colhe Três, pelas contribuições do RUP e da UML. De fato, ao estudarmos um pouco mais a fundo esse modelo de processo e essa linguagem de modelagem, percebemos que eles possuem um grande potencial para a realização dos primeiros passos do desenvolvimento de um software para um público que ainda não possui, necessariamente, um domínio razoável de computação, principalmente ao considerar que a modelagem UML dirige o olhar do desenvolvimento a partir do uso de diagramas visuais para representar as estruturas e os comportamentos do sistema.

Com o objetivo de ampliar a viabilidade de uso da Engenharia Didático-Informática por educadores matemáticos que buscam produzir jogos matemáticos de simulação para situações didáticas, construiremos uma metodologia, baseada nas orientações do RUP e na modelagem UML, que auxilie a transformação dos requisitos advindos dos estudos realizados nas etapas iniciais da EDI em artefatos oriundos da UML que se aproximem ao máximo do código final a ser implementado.

Os processos de criação e uso parcial dessa metodologia serão apresentados no capítulo seguinte deste trabalho.

## 5 DOS REQUISITOS À IMPLEMENTAÇÃO: CONSTRUÇÃO DE UMA METODOLOGIA DE AUXÍLIO À ETAPA DE DESENVOLVIMENTO DE JOGOS MATEMÁTICOS DE SIMULAÇÃO PARA SITUAÇÕES DIDÁTICAS

Como justificado anteriormente, a análise acerca da realização da fase de Desenvolvimento da EDI nos levou a elaboração de um novo objetivo para a pesquisa: *construir uma metodologia para a etapa de Desenvolvimento da EDI que forneça um percurso para transformação dos requisitos em código a ser implementado no desenvolvimento de jogos matemáticos de simulação para situações didáticas.*

Durante o estudo e a utilização do RUP e da UML para a elaboração do Modelo de Casos de Uso do Mankala Colhe Três Digital, percebemos que eles poderiam trazer contribuições para a pesquisa que superariam a sistematização dos requisitos do jogo e da etapa de Concepção e Análises Prévias. De fato, esse documento pode se mostrar muito útil na modelagem de um sistema, servindo de artefato de entrada para o início da discussão acerca da arquitetura do software, como detalharemos ao longo deste capítulo.

Nossa discussão acerca da metodologia que iremos propor neste capítulo será apresentada em dois blocos, descritos a seguir:

- I. Aportes da Linguagem de Modelagem Unificada (UML) e do Processo Unificado da *Rational* (RUP) na Engenharia Didático-Informática;
- II. Metodologia para a etapa de Desenvolvimento da EDI na produção de jogos matemáticos de simulação para situações didáticas.

O primeiro bloco trará uma fundamentação teórica sobre a UML e o RUP, acompanhada de uma articulação com a nossa proposta de metodologia. No segundo bloco, apresentaremos a construção de nossa metodologia, apoiada nos princípios da engenharia de software discutidos, e finalizaremos por meio de uma exemplificação de uso da metodologia desenvolvida na modelagem parcial do Mankala Colhe Três Digital.

### 5.1 APORTES DA LINGUAGEM DE MODELAGEM UNIFICADA (UML) E DO PROCESSO UNIFICADO DA *RATIONAL* (RUP) NA ENGENHARIA DIDÁTICO-INFORMÁTICA

Nesse primeiro bloco, apresentaremos ao leitor alguns elementos de base da UML e do RUP que serão considerados nas discussões deste capítulo. Em seguida, analisaremos os

possíveis aportes oriundos dessas duas abordagens da Engenharia de Software para a etapa de Desenvolvimento do modelo de processo da Engenharia Didático-Informática.

### 5.1.1 Considerações gerais sobre a UML

Iniciaremos nossa explanação sobre a Linguagem de Modelagem Unificada – UML partindo de sua definição, apresentada pelos seus desenvolvedores Grady, Booch, James Rumbaugh e Ivar Jacobson:

A UML (Unified Modeling Language) é uma linguagem-padrão para a elaboração da estrutura de projetos de software. Ela poderá ser empregada para a visualização, a especificação, a construção e a documentação de artefatos que façam uso de sistemas complexos de software (BOOCH *et al.*, 2006, p. 13).

Como percebemos na citação acima, o uso da UML é bastante vasto e é ideal para projetos de softwares complexos. Além dessa variedade de empregos da UML, os autores endossam que o guia do usuário, produzido por eles, se interessa a qualquer pessoa que seja envolvida na produção, implantação e na manutenção de software, sendo direcionado principalmente para os membros de equipes de desenvolvimento responsáveis pela criação de modelos UML, mesmo que também seja adequado a quem lê esses modelos, trabalhando em conjunto para compreender, construir, testar e gerar um sistema complexo de software (BOOCH *et al.*, 2006).

A UML fornece uma maneira de visualizar a arquitetura de um sistema por meio de diagramas, elementos, componentes individuais do sistema, como eles podem interagir com outros componentes do software e como o sistema funcionará. Segue uma fala dos criadores dessa linguagem sobre um diagrama UML:

Um diagrama é a apresentação gráfica de um conjunto de elementos, geralmente representadas como gráficos de vértices (itens) e arcos (relacionamentos). São desenhados para permitir a visualização de um sistema sob diferentes perspectivas; nesse sentido, um diagrama constitui uma projeção de um determinado sistema. Em todos os sistemas, com exceção dos mais triviais, um diagrama representa uma visão parcial dos elementos que compõem o sistema (BOOCH *et al.*, 2006, p. 26).

Partindo da citação anterior, é importante destacar que um diagrama em UML é uma representação gráfica parcial que modela apenas um aspecto do sistema, ao longo de seu ciclo de vida. Sendo assim, a UML apresenta vários diagramas distintos para representar aspectos do sistema sob diversas perspectivas.

Os diagramas da UML podem ser classificados em duas categorias: *Diagramas estruturais* e *Diagramas comportamentais*. Os diagramas estruturais, destinados à visualização das partes estáticas de um sistema, são: *diagrama de classes*; *diagrama de componentes*; *diagrama de estrutura composta*; *diagrama de objetos*; *diagrama de implantação*; *diagrama de artefatos*. Já os diagramas comportamentais, que permitem a visualização das partes dinâmicas de um sistema, são: *diagrama de caso de uso*; *diagrama de sequência*; *diagrama de comunicação*; *diagrama de gráfico de estados*; *diagrama de atividades*.

Dentre os diagramas apresentados, veremos mais adiante que nos utilizaremos de quatro deles, que comporão a metodologia de desenvolvimento baseada na modelagem de jogos matemáticos de simulação para situações didáticas. Apresentaremos, a seguir, as definições de cada um dos diagramas a serem utilizados na presente pesquisa.

Iniciaremos apresentando o que caracteriza o diagrama de classe, como pode ser visto a seguir:

Um *diagrama de classe* exibe um conjunto de classes, interfaces e colaborações bem como seus relacionamentos. Esses diagramas são encontrados com maior frequência em sistemas de modelagem orientados a objeto e abrangem uma visão estática da estrutura do sistema. (BOOCH *et al.*, 2006, p. 26).

Veremos que o diagrama de classes do sistema representará o produto esperado para aproximar o que o sistema possui de requisitos ao código a ser implementado. Desse modo, veremos mais adiante que a construção do diagrama de classes do sistema será um dos grandes objetivos do processo de modelagem dos casos de uso.

Um importante diagrama que utilizaremos na nossa pesquisa é o diagrama de sequências. Assim como o diagrama de comunicação, ele pode ser caracterizado como um tipo de diagrama de *interação*, por exibir uma interação e consistir em um conjunto de objetos e papéis, inclusive mensagens trocadas entre eles. A característica particular do diagrama de sequências é que sua ênfase está na ordenação temporal das mensagens (BOOCH *et al.*, 2006).

Por fim, gostaríamos de detalhar o que caracteriza o diagrama de atividade. Segue o que abordam Booch *et al.*:

Um *diagrama de atividade* exibe a estrutura de um processo ou outra computação, como o fluxo de controle e os dados de cada etapa de uma computação. Abrange a visão dinâmica do sistema e é importante principalmente para a modelagem da função de um sistema e dá ênfase ao fluxo de controle entre objetos (BOOCH *et al.*, 2006, p. 28).

Esses diagramas são utilizados para fazer a modelagem de aspectos dinâmicos do sistema, por meio de atividades sequenciais em um processo computacional. Outro diagrama, já enfatizado na pesquisa, é o Diagrama de Casos de Uso. Sobre esse diagrama, pode ser afirmado que

Um *diagrama de caso de uso* exibe um conjunto de casos de uso e atores (um tipo especial de classe) e seus relacionamentos. Diagramas de casos de uso abrangem a visão estática de casos de uso do sistema. Esses diagramas são importantes principalmente para a organização e a modelagem de comportamentos do sistema (BOOCH *et al.*, 2006, p. 27).

Sobre a representação gráfica do caso de uso, é importante lembrar que, durante a realização da Concepção e Análise *a priori* do Mankala Colhe Três Digital, apresentamos o Diagrama de Casos de Uso do Mankala Colhe Três Digital, que mostra representações gráficas do caso de uso, assim como do *ator* e de alguns tipos de *relacionamentos*.

A UML possui quatro tipos de relacionamentos: *dependência*; *associação*; *generalização*; e *realização*. Descrevemos brevemente a utilização de relacionamentos de associação e generalização na produção do Diagrama de Casos de Uso do Mankala Colhe Três. Apresentaremos, a seguir, a definição do relacionamento e dependência, já que ele também será utilizado na presente pesquisa:

Uma *dependência* é um relacionamento semântico entre dois itens, nos quais a alteração de um (o item independente) pode afetar a semântica do outro (o item dependente). Graficamente, uma dependência é representada por linhas tracejadas, possivelmente com setas e ocasionalmente incluindo um rótulo (BOOCH *et al.*, p. 25).

Segue uma representação gráfica do relacionamento *dependência*:

Figura 22 – Representação gráfica do relacionamento de dependência



Fonte: elaborada pelo autor.

Um aspecto que acreditamos ser importante comentar é que a UML, por se tratar de uma linguagem, ela contribui no desenvolvimento de um software, mas depende de um modelo de processo que a utilize, como podemos perceber na citação a seguir:

A UML é apenas uma linguagem e, portanto, é somente uma parte de um método para desenvolvimento de software. A UML é independente do processo, apesar de ser perfeitamente utilizada em processo orientado a casos de usos, centrado na arquitetura, iterativo e incremental (BOOCH *et al.*, 2006, p. 13).

Veremos, na próxima seção deste capítulo, que o modelo de processo unificado RUP possui as características de ser orientado a casos de uso, centrado na arquitetura, iterativo e incremental, características trazidas no final da citação acima.

### 5.1.2 Considerações gerais sobre o RUP

Como citamos na seção anterior, a UML representa uma linguagem unificada para expressar os vários modelos, mas não traz informações sobre como desenvolver o software. Sendo assim, para que seja obtido o maior proveito da UML, é preciso considerar um processo que contenham as seguintes características: orientado a casos de uso; centrado na arquitetura; iterativo e incremental (BOOCH *et al.*, 2006).

O RUP é o processo unificado desenvolvido pelos mesmos criadores da UML na empresa *Rational* e agora é propriedade da empresa IBM. Caracterizado por uma estrutura de processo para desenvolvimento de sistemas de computador orientados a objetos que pode ser adaptada e estendida para levar em conta as necessidades da organização de quem a utiliza, adotando e incorporando muitas das melhores práticas de desenvolvimento de engenharia de software moderno (DOGBOEVI, 2020).

O termo *processo* representa, em geral, um conjunto de passos parcialmente ordenados com a intenção de atingir uma meta. No *Rational Unified Process* (RUP), essa meta é a de produzir softwares da mais alta qualidade que atenda às necessidades do usuário final de acordo com planejamentos e orçamentos previsíveis (KRUCHTEN, 2003).

Ao reconhecer que os modelos de processo convencionais apresentam uma visão única do processo do software, o RUP é descrito em três perspectivas distintas:

- Uma perspectiva dinâmica, que mostra as fases do modelo ao longo do tempo.
- Uma perspectiva estática, que mostra as atividades realizadas no processo.
- Uma perspectiva prática, que sugerem boas práticas a serem usadas durante o processo (SOMMERVILLE, 2011, p. 34)

O RUP é dividido em quatro fases: *iniciação*; *elaboração*; *construção*; e *transição*. Na iniciação, são estabelecidos a visão, o escopo e o plano inicial para o projeto. É nessa fase em que devem ser identificadas todas as entidades externas (pessoas e sistemas) que vão interagir

com o sistema e definir as interações (SOMMERVILLE, 2011). Uma pergunta norteadora para essa fase seria *o que se pretende produzir?*

A fase de elaboração é caracterizada pelo design, implementação e testes da arquitetura do sistema, para completar o plano do projeto. A questão norteadora para essa fase pode ser *como pretendo produzir?*

A fase de *construção* cria a primeira versão do sistema operacional e a *transição* fornece o sistema a seus usuários finais. Segundo Booch *et al.* (2006), as fases de iniciação e elaboração abrangem as atividades criativas e de engenharia do ciclo de vida do desenvolvimento, enquanto a construção e transição constituem sua produção.

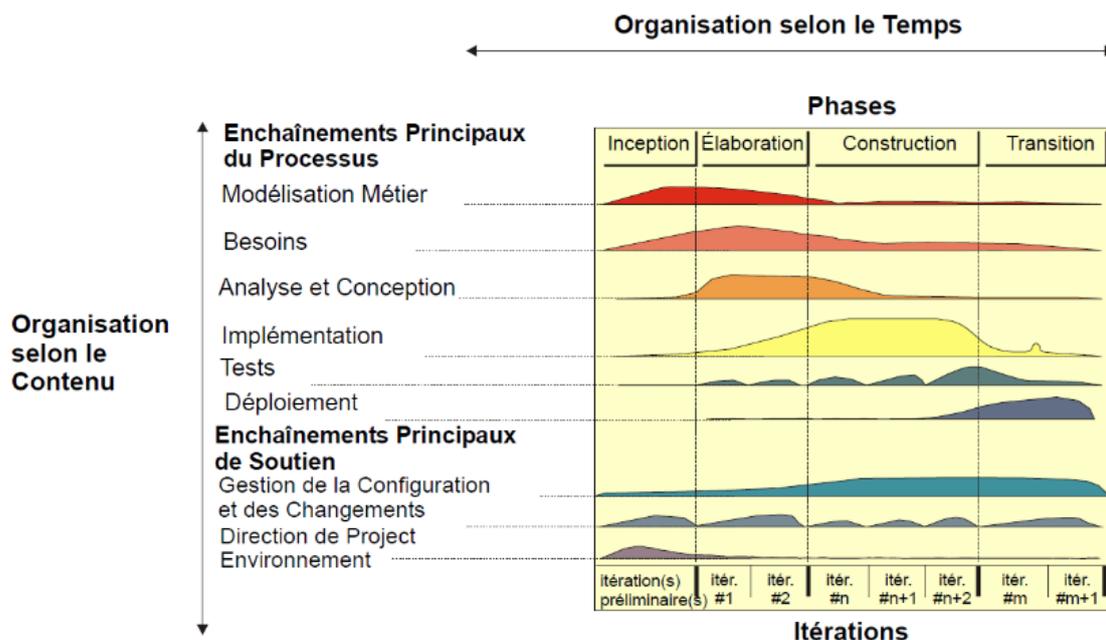
O RUP considera que existem iterações dentro de cada uma das fases. Sobre o significado de iteração, Booch *et al.* (2006) afirmam que

Uma iteração representa um ciclo completo de desenvolvimento, desde a captação de requisitos na análise até a implementação e a realização de testes, resultando na versão de um projeto executável. A versão não precisa incluir um conjunto completo de características da versão comercial. Sua finalidade fornece uma base sólida para avaliar e testar, assim como uma linha de base uniforme para o próximo ciclo de desenvolvimento (BOOCH *et al.*, 2006, p. 445).

Além da característica de desenvolvimento de software iterativamente, a perspectiva prática sobre o RUP descreve as *boas práticas* da engenharia de softwares que são recomendadas para o uso no desenvolvimento de sistemas. Seguem as seis boas práticas cobertas pelo RUP: *desenvolver softwares iterativamente; gerenciar requisitos; usar arquiteturas baseadas em componentes; modelar visualmente o software; verificar continuamente a qualidade do software; controlar mudanças no software* (SOMMERVILLE, 2011).

A estrutura do processo RUP possui duas dimensões, relativas às suas perspectivas dinâmica e estática. A Figura 22 mostra a arquitetura global desse modelo de processo:

Figura 23 – Estrutura do processo – duas dimensões



Fonte: Kruchten (1999, p. 4).

A primeira dimensão (eixo horizontal) representa o tempo, da esquerda para a direita, ordenando o aspecto dinâmico do processo e é expresso em termos de ciclos, fases, iterações e marcos. A segunda dimensão (eixo vertical) representa os fluxos de trabalho (*workflows*) ou disciplinas do processo, agrupando logicamente as atividades por natureza. Essa dimensão representa o aspecto estático do processo, através de sua descrição em termos de componentes de processo, atividades, fluxos, artefatos e trabalhadores (KRUCHTEN, 2003).

Ainda sobre a estrutura do processo RUP (Figura 22), é possível perceber que ela contém informações visuais acerca da intensidade do trabalho da disciplina por momento do ciclo de vida do software. Essa característica é representada pelas espessuras dos gráficos correspondentes a cada disciplina, em que cada disciplina possui uma cor distinta das outras.

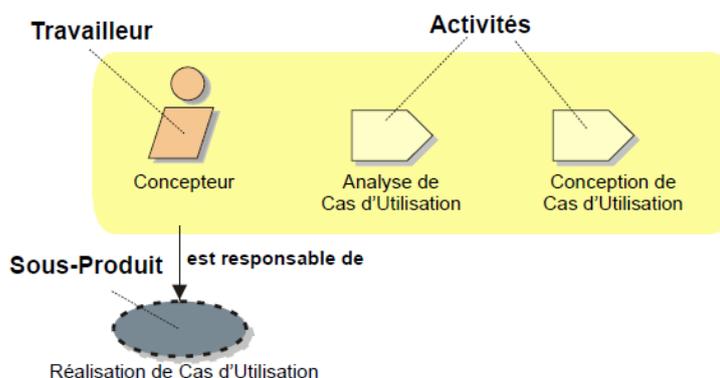
A estrutura do processo RUP também traz outra informação que vale destacar: as fases (dimensão horizontal) são compostas por iterações, representadas na parte inferior da estrutura. Percebemos, por exemplo, que enquanto as disciplinas de *Modelagem de negócios* (*Modélisations Métier*) e *Requisitos* (*Besoins*) são muito enfatizadas nas duas primeiras fases, o trabalho da disciplina de *implantação* (*Déploiement*) se concentra na fase final do processo. Outro aspecto importante é perceber que tanto a disciplina de *Requisitos* (*Besoins*) quanto a de *Gerenciamento de Configurações e Mudanças* (*Gestion de la Configuration et des Changements*) estão presentes ao longo de todo o ciclo de vida, mesmo que haja pequenas oscilações nessa ênfase.

Ao passar pelas quatro fases (iniciação, elaboração, construção e transição), realiza-se um ciclo de desenvolvimento e o software é gerado. No entanto, é importante destacar que esse processo pressupõe repetições, tendo em vista evolução do software, já que

A menos que a vida do produto seja interrompida, um produto existente evoluirá para sua próxima geração pela repetição da mesma sequência de fases de concepção, elaboração, construção e transição. Isso é a evolução do sistema, de modo que os ciclos de desenvolvimento posteriores aos ciclos iniciais são seus ciclos de evolução (BOOCH *et al.*, 2006, p. 447).

Voltando o olhar ao aspecto estático do processo, Khuchten (2003) afirma que “um processo descreve *quem está fazendo o quê, como e quando*, por meio de trabalhadores, artefatos, atividades e fluxos, respectivamente. Segue uma ilustração que exemplifica as relações existentes entre elementos do RUP:

Figura 24 – Trabalhadores, atividades e artefatos



Fonte: Kruchten (1999, p. 12).

Retomando às informações presentes na estrutura do processo RUP (Figura 22), o RUP é composto por nove *workflows*, em que os seis primeiros representam fluxos de *processo* e os outros três são fluxos centrados no *suporte* do sistema. Segue uma lista dos fluxos de trabalho do RUP acompanhados de uma breve descrição (SOMMERVILLE, 2011):

1. Modelagem de negócios: Os processos de negócio são modelados por meio de casos de uso de negócios;
2. Requisitos: Atores que interagem com o sistema são identificados e casos de uso são desenvolvidos para modelar os requisitos do sistema;
3. Análise e design: um modelo de design é criado e documentado com modelos de arquitetura, modelos de componentes, modelos de objetos e modelos de sequência;

4. Implementação: Os componentes do sistema são implementados e estruturados em subsistemas de implementação. A geração automática de código a partir de modelos de projeto ajuda a acelerar esse processo;
5. Teste: o teste é um processo iterativo que é feito em conjunto com a implementação. O teste do sistema segue a conclusão da implementação;
6. Implantação: Um *release* do produto é criado, distribuído aos usuários e instalado em seu local de trabalho;
7. Gerenciamento de configuração e mudanças: esse *workflow* de apoio gerencia as mudanças do sistema;
8. Gerenciamento de projeto: esse *workflow* de apoio gerencia o desenvolvimento do sistema;
9. Ambiente: esse *workflow* está relacionado com a disponibilização de ferramentas apropriadas para a equipe de desenvolvimento de software.

Em cada *workflow* há trabalhadores, atividades e artefatos correlacionados. Detalhando um pouco melhor o significado desses elementos do RUP, Booch *et al.* (2006) afirma que

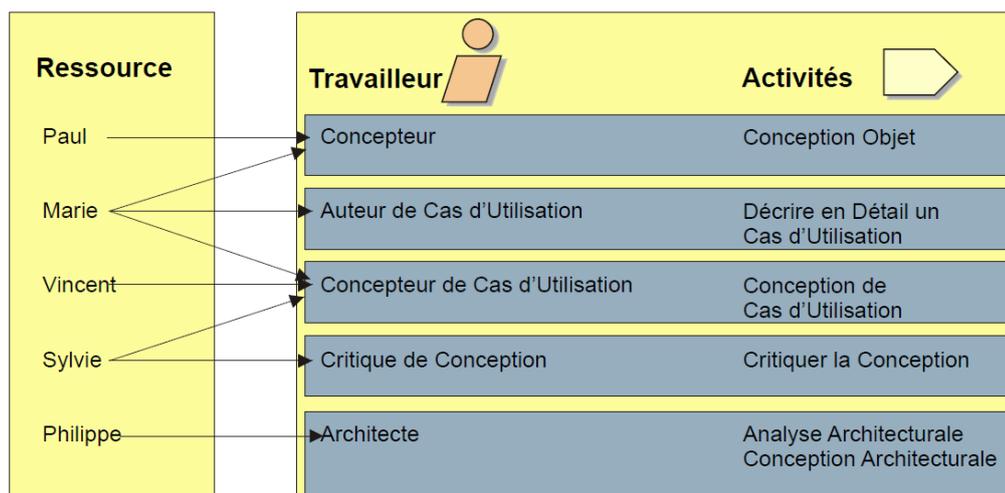
Um *artefato* é algum documento, relatório ou executável, que é produzido, manipulado ou consumido. Uma *atividade* descreve as tarefas – criando, realizando e verificando etapas – executada pelos trabalhadores para criar ou modificar artefatos, juntamente com as técnicas e diretrizes para a realização de tarefas. Possivelmente incluindo a utilização de ferramentas para ajudar a automação de algumas tarefas (BOOCH *et al.*, 2006, p. 448).

É importante destacar que, no RUP, a definição de *trabalhador* se distingue da de indivíduo. Na verdade, o significado trazido para trabalhador se aproxima ao de “papel”, pois os trabalhadores descrevem como os indivíduos deveriam se comportar no negócio e as responsabilidades de cada um. Isso pode ser mais bem compreendido pela citação a seguir:

Membros individuais das empresas de desenvolvimento de software usam chapéus diferentes ou executam partes ou papéis diferentes. O mapa do indivíduo para o trabalhador é desempenhado pelo gerente de projeto quando planejado e provendo de pessoal o projeto. Este mapa permite a um indivíduo atuar como vários trabalhadores e a um trabalhador ser executado por vários indivíduos (KRUCHTEN, 2003, p. 31).

Podemos ilustrar essa diferenciação entre trabalhadores e indivíduos por meio da ilustração a seguir:

Figura 25 – Exemplo de pessoas e trabalhadores no RUP



Fonte: Kruchten (1999, p. 13).

Ao apresentar o exemplo acima, Kruchten (2003) comenta que se observarmos o indivíduo Sylvie, por exemplo, ela poderia ser um *Trabalhador (Travailleur): Designer de caso de uso (Concepteur de Cas d'Utilisation)* pela manhã e atuar como um *Trabalhador (Travailleur): Revisor de design (Critique de Conception)* pela tarde. Da mesma forma, Paul e Marie são ambos *Designers (Concepteurs)*, embora sejam provavelmente responsáveis por classes diferentes ou pacotes de design diferentes (KRUCHTEN, 2003).

Para finalizar nossas considerações iniciais sobre o RUP, gostaríamos de enfatizar a característica flexível do uso de elementos desse processo em projetos diversos, já comentada no início dessa seção. Segundo seus criadores,

O Rational Unified Process é um processo *configurável*. Embora não exista um único processo adequado para todas as empresas de desenvolvimento de software, o Rational Unified Process pode ser ajustado e redimensionado para atender às necessidades de projetos que variam desde pequenas equipes até grandes empresas de desenvolvimento de software. O Rational Unified Process se fundamenta em uma arquitetura de processo simples e clara que proporciona uma base comum em uma família de processos e ainda pode ser modificada para acomodar várias situações. O Rational Unified Process contém uma orientação para se configurar o processo e atender às necessidades de uma empresa (BOOCH *et al.*, 2006, p. 444).

Na nossa pesquisa, buscamos considerar contribuições advindas dos princípios do modelo de processo RUP e da linguagem de modelagem UML para compor parte do modelo de processo da EDI, no caso particular da inserção de uma metodologia para a fase de Desenvolvimento da EDI na concepção de jogos matemáticos de simulação para situações didáticas.

Na seção seguinte, traremos um olhar mais específico para um desses *workflows* do modelo de processo RUP, que contribuirá para a construção de uma metodologia para a etapa de Desenvolvimento da EDI, voltado para a produção de jogos matemáticos de simulação para situações didáticas.

### 5.1.3 Um olhar para a disciplina *Análise e Design* do RUP

Ao propor uma metodologia para dar suporte à etapa de Desenvolvimento da EDI, nosso objetivo é aproximar o diálogo entre as áreas da Educação Matemática e a Engenharia de Softwares, como já mencionamos. A partir dos estudos envolvendo o modelo de processo unificado RUP, percebemos que nosso foco está no *workflow* Análise e Design (ou Projeto<sup>9</sup>), já que essa disciplina do RUP tem como proposta analisar e realizar o design e traduzir os requisitos numa especificação que escreva como implementar o sistema (KRUNCHTEN, 2003).

Segue uma citação que sintetiza a proposta do fluxo de trabalho Análise e Design do RUP apresentada acima:

Para fazer essa tradução, você tem que entender os requisitos e transformá-los num projeto do sistema, selecionando a melhor estratégia de implementação. Cedo no projeto, você tem que estabelecer uma arquitetura robusta, de forma que possa projetar um sistema fácil de entender, construir e evoluir. Então é necessário ajustar o projeto para combinar com o ambiente de implementação, projetando-o para desempenho, robustez, escalabilidade e teste, entre outras qualidades (KRUNCHTEN, 2003, p. 143).

Para que compreendamos um pouco a disciplina de Análise e Design e como considerá-la como parte integrante do modelo de processo da EDI, no caso dos jogos matemáticos de simulação, precisamos nos debruçar sobre alguns conceitos importantes, como os próprios “Análise” e “Design” do software.

Kruchten (2003) enfatiza que o propósito da Análise é transformar os requisitos do sistema numa forma que mapeie a área do designer do software de interesse, que seria um conjunto de classes e subsistemas. Segundo o autor, essa transformação é dirigida pelos casos de uso e amoldada posteriormente pelos requisitos não funcionais. O autor complementa, afirmando que

---

<sup>9</sup> Em algumas traduções, usa-se o termo “projeto”, em outras, sua versão na língua inglesa “design”. Decidimos explicitar as duas para evitar incompreensões do leitor em possíveis citações diretas com traduções distintas.

A análise focaliza-se em assegurar que os requisitos funcionais do sistema sejam controlados. Por causa da simplicidade, ignora muitos dos requisitos não funcionais do sistema e também as restrições do ambiente de implementação. Como resultado, a análise expressa uma linguagem quase ideal do sistema (KRUCHTEN, 2003, p. 142).

Ao realizar um comparativo entre a Análise e o Design, o autor ressalta que o Design tem o propósito de adaptar os resultados da Análise às restrições impostas pelos requisitos não funcionais, pelo ambiente de implementação, pelos requisitos de desempenho e assim sucessivamente. Portanto, o Design é o refinamento da Análise, focalizando-se no aperfeiçoamento do design do sistema através da cobertura completa dos requisitos (KRUCHTEN, 2003).

Um questionamento discutido por Kruchten (2003) e que, para a nossa proposta, se mostra muito pertinente é *até onde o design (ou projeto) tem que ir?* Considerando que estamos buscando aproximar cada vez mais o educador matemático do profissional da computação que irá implementar o software, sem que haja perdas consideráveis de qualidade no produto esperado por problemas de comunicação entre áreas, trataremos a seguir algumas considerações sobre esse questionamento que nos é adequado.

Segundo Kruchten (2003), o projeto tem que definir apenas o suficiente do sistema, de forma que sua implementação possa ser feita sem equívocos. Sobre essa suficiência, o autor comenta que

O que constitui o “suficiente” varia de projeto para projeto e de companhia para companhia. Em alguns casos, o projeto pode ser elaborado ao ponto que o sistema possa ser implementado por uma transformação direta e sistemática do projeto no código.

Em outros casos, o projeto se assemelha a um esboço elaborado somente o suficiente para assegurar que o implementador possa produzir um conjunto de componentes que satisfaça os requisitos. O grau de especificação então varia com as perícias do implementador, a complexidade do projeto e o risco do projeto poder ser mal interpretado (KRUCHTEN, 2003, p. 144).

O autor ainda apresenta outros níveis de precisão, como o da *engenharia reversa*, que ocorre quando o design é especificado precisamente, permitindo que o seja amarrado e mantido sincronizado ao projeto.

Tendo em vista que o RUP expressa cada *workflow* em termos de trabalhadores, artefatos e atividades envolvidas num fluxo de atividades, Kruchten (2003) afirma que os principais trabalhadores dessa disciplina são o *Arquiteto* e o *Designer* (ou *projetista*), mas que podem opcionalmente incluir o *Projetista de banco de dados*, o *Projetista de cápsula*, *Revisor de arquitetura* e *Revisor de Projeto*.

Sobre principais artefatos da disciplina Análise e Design do RUP, Kruchten (2003) destaca o *documento de arquitetura de software* e o *modelo de projeto*. Segundo o autor, o Documento de arquitetura de software capta várias visões arquitetônicas do software. Sobre modelo de projeto, gostaríamos de iniciar sua explanação a partir da citação que segue:

O modelo de projeto consiste em colaborações de classes que podem ser agregadas em pacotes e subsistemas para ajudar a organizar o modelo e fornecer blocos compostos da construção dentro do modelo. Uma *classe* é a descrição de um conjunto de objetos que compartilham as mesmas responsabilidades, relações, operações, atributos e semântica. Um *pacote* é um agrupamento lógico de classes, talvez para propósitos organizacionais, que reduzem a complexidade do sistema. Um *subsistema* é um tipo de pacote, consistindo de um conjunto de classes que agem em unidade para fornecer comportamentos específicos (KRUCHTEN, 2003, p. 146).

Como pode ser percebido acima, o *modelo de projeto* é o artefato mais completo da disciplina Análise e Design. Ele fornece o comportamento do sistema, por meio da abstração da implementação, representada por todas as classes de design, subsistemas, pacotes, colaborações e seus relacionamentos. É importante destacar que esse artefato é derivado principalmente do modelo de caso de uso e requisitos não funcionais (KRUCHTEN, 2003).

O autor traz uma fala sobre um *modelo de análise*, que, apesar de não ser, de fato um documento a ser mantido, representa uma etapa importante, como poderemos ver a seguir:

Geralmente, há um modelo de projeto do sistema; a análise produz um esboço grosseiro do sistema que mais adiante será refinado no projeto. As camadas superiores deste modelo descrevem os aspectos da própria aplicação, ou mais orientados à análise, do sistema. Usar um único modelo reduz o número de artefatos que devem ser mantidos em estado consciente (KRUCHTEN, 2003, p. 146).

Podemos perceber, portanto, que um percurso natural para a evolução dos requisitos é a elaboração do modelo de projeto, que parte de um Modelo de Casos de Uso e é esboçado por meio de um modelo de análise.

Ainda em relação à disciplina Análise e Design, é dado destaque ao papel das interfaces, que são usadas para especificar o comportamento oferecido por classes, subsistemas e componentes de um modo independente da implementação do comportamento (KRUCHTEN, 2003). Segundo o autor, na Análise e Design, as interfaces

Especificam um conjunto de operações executado pelos elementos do modelo, inclusive o tipo e número retornado e os tipos de parâmetros. Qualquer dos dois elementos do modelo que oferece a mesma interface é trocável. As interfaces melhoram a flexibilidade dos projetos, reduzindo dependências entre as partes do sistema e os tornando mais fáceis de mudar (KRUCHTEN, 2003, p. 146).

Uma das características do Design no RUP é que ele é baseado em componentes, que são elementos usados para expressar o modo que o sistema é implementado num conjunto de partes menores. Esse uso de componentes permite o projeto, a implementação, a entrega e a substituição de várias partes do sistema (KRUCHTEN, 2003). Explanando um pouco mais sobre o componente, o autor detalha que

O elemento do modelo de projeto que corresponde ao componente é o subsistema: os subsistemas são unidades separadas de funcionalidade que permitem que partes diferentes do sistema sejam projetadas e implementadas independentemente. Os subsistemas oferecem interfaces e são dependentes nas interfaces de outros elementos do modelo (KRUCHTEN, 2003, p. 147).

Em posse dos conceitos apresentados, apresentaremos os *Detalhes* do fluxo de trabalho *Análise e Design* na seção seguinte, que nos leva a construção da nossa metodologia de suporte à etapa de desenvolvimento da EDI na produção de jogos matemáticos de simulação para situações didáticas.

## 5.2 METODOLOGIA PARA A ETAPA DE DESENVOLVIMENTO DA EDI NA PRODUÇÃO DE JOGOS MATEMÁTICOS DE SIMULAÇÃO PARA SITUAÇÕES DIDÁTICAS

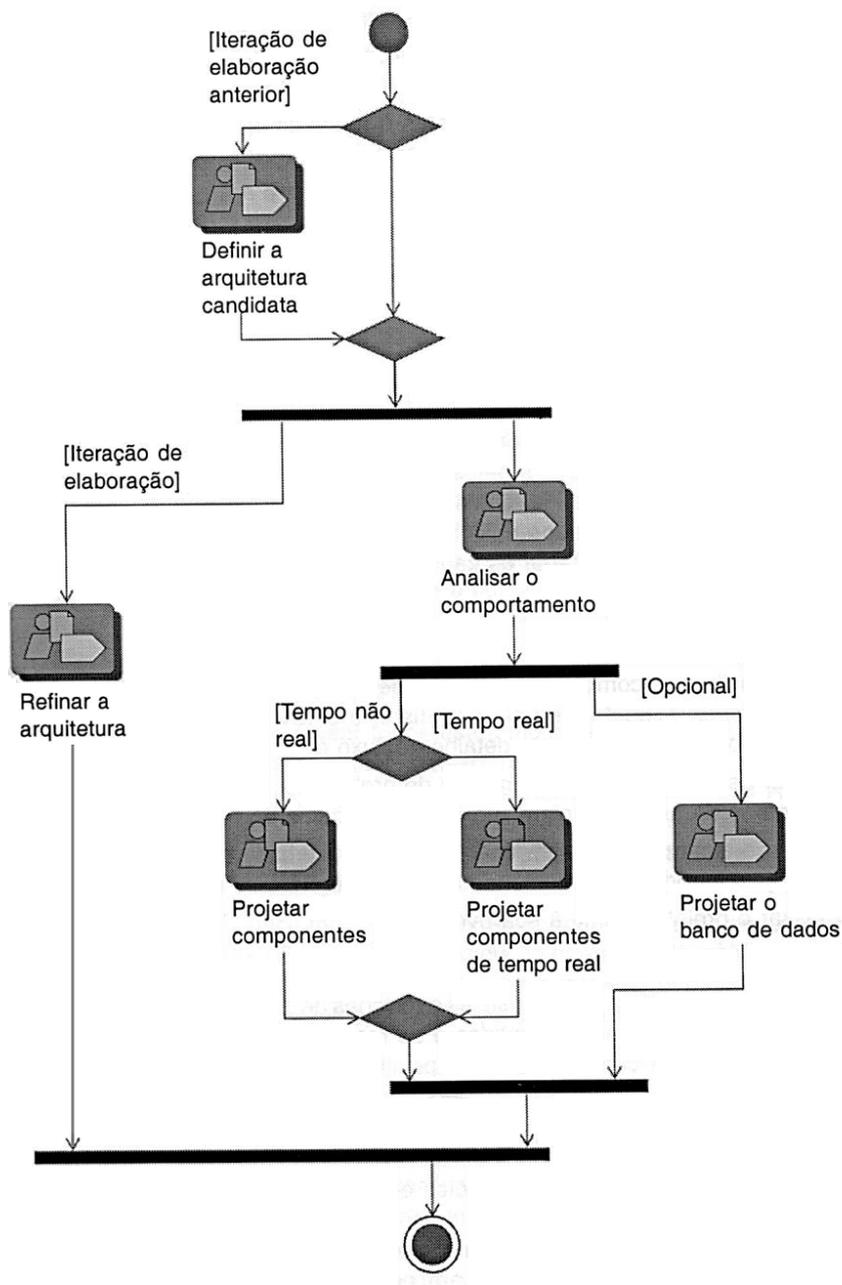
A seção anterior deste trabalho nos apresentou uma fundamentação sobre elementos da engenharia de software, no que se refere à modelagem com a Linguagem de Modelagem Unificada (UML) e ao modelo de processo de software do RUP.

Ao explicar sobre o RUP, trouxemos mais detalhes da disciplina Análise e Design, pelo fato de ela representar uma área da engenharia de software pouco detalhada na proposta do modelo de processo da Engenharia Didático-Informática e, a nosso ver, de poder contribuir bastante para essa metodologia de desenvolvimento de softwares educativos.

Esta seção trará a metodologia de suporte à etapa de desenvolvimento da EDI na produção de jogos matemáticos de simulação para situações didáticas, inspirada no fluxo de trabalho Análise e Design do RUP.

Gostaríamos de iniciar a construção de nossa metodologia por meio da análise da Figura 25, que contém o fluxo de Análise e Design proposto pelo RUP, que segue:

Figura 26 – Fluxo de Análise e Design



Fonte: Kruchten (2003, p. 149).

A figura acima apresenta uma iteração do fluxo Análise e Design, por meio de um diagrama de atividade da UML. Nesse diagrama, chamamos de *detalhe do fluxo* cada um dos estados de atividade. Cada detalhe do fluxo, por sua vez, possui pelo menos uma atividade. (KRUCHTEN, 2003).

Descreveremos, a seguir, de forma sucinta, características de cada um dos *detalhes de fluxo* do diagrama acima, com exceção do “Projetar componentes de tempo real”, por não se aplicar à nossa proposta de software a ser concebido.

O primeiro deles é o de “Definir uma arquitetura candidata”. Sua característica geral é de criar um esboço da arquitetura do sistema. Além da criação desse esboço inicial, as metas de suas atividades são de identificar classes de análise a partir dos casos de uso arquiteturalmente significativos e atualizar as realizações de caso de uso com as interações das classes de análise (KRUCHTEN, 2003).

O detalhe de fluxo “Refinar a arquitetura”, como a própria expressão sugere, conclui a arquitetura para uma iteração. Segundo Kruchten (2003), esse refinamento fornece a transição natural das atividades de análise para as atividades de design; descreve a organização da arquitetura de tempo de execução e de implementação do sistema; organiza o modelo de implementação para efetuar a transição entre o design e a implementação; e mantém a consistência e a integridade da arquitetura, assegurando que os novos elementos de design identificados para a iteração atual sejam integrados aos elementos de design preexistentes.

Em relação ao detalhe de fluxo “Analisar o comportamento”, seu papel é transformar as descrições comportamentais fornecidas pelos casos de uso em um conjunto de elementos, no qual o design possa se basear. Kruchten (2003) afirma que o interesse nessas atividades é muito maior na forma como entregar as capacidades necessárias para a aplicação do que nos requisitos não funcionais. Por isso, nesse detalhe de fluxo pode ser incluída a modelagem e a criação de protótipo da interface com o usuário, de modo a evitar o investimento em protótipos funcionais da interface com o usuário mais elaborados até que haja um consenso sobre o design geral dessa interface.

Sobre “Projetar componentes”, seu objetivo é refinar o design do sistema, que considera: refinar as definições de elementos de design por meio da elaboração de detalhes de como os elementos de design realizam o comportamento exigido deles; refinar e atualizar as realizações de casos de uso baseadas no novo elemento de design identificado (isto é, manter atualizadas as realizações de casos de uso); e revisar o design quanto à sua evolução (KRUCHTEN, 2003).

Por fim, sobre “Projetar o banco de dados”, sua característica é identificar as classes de design que precisam ser persistidas em um banco de dados e projetar as estruturas de banco de dados correspondente. Também faz parte desse detalhe de fluxo definir mecanismos e estratégias de armazenamento e recuperação de dados persistentes, de modo a atender aos critérios de desempenho do sistema (KRUCHTEN, 2003).

A seguir, apresentaremos como iremos considerar o cumprimento das atividades do fluxo de Análise e Design na proposição de nossa metodologia. Nossas escolhas foram inspiradas no trabalho de Dogboevi (2020), que realizou uma pesquisa que se assemelha à proposta deste capítulo, por meio da proposição de uma metodologia de desenvolvimento de softwares baseado no RUP e na UML.

### **5.2.1 Cumprimento das atividades dos detalhes de fluxo “Definir uma arquitetura candidata” e “Refinar a arquitetura”**

As atividades dos detalhes de fluxo “Definir uma arquitetura candidata” e “Refinar a arquitetura” serão consideradas na nossa metodologia guiando-se pela seguinte estrutura:

- Uso do Modelo de Casos de Uso do RUP como artefato de entrada, com os requisitos do sistema;
- Modelagem da relação estabelecida entre caso de uso e atores envolvidos;
- Identificação das classes de análise de Caso de Uso.

O uso do Modelo de Casos de Uso, como já foi mencionado, representa uma maneira bastante adequada para representar o documento de requisitos no processo da EDI, particularmente no nosso caso, da produção de jogos matemáticos de simulação para situações didáticas. Sua característica de ser orientada por suas funcionalidades, representando um processo dirigido por casos de uso, traz informações que pertencem à etapa de Concepção e Análise *a priori* da EDI, permitindo que esse documento seja único e representativo dos requisitos do jogo oriundos dos estudos das Análises Prévias.

A partir do momento em que os requisitos são declarados, aplicaremos uma técnica de modelagem dos casos de uso do sistema. A que escolhemos se baseia na proposta trazida por Dogboevi (2020) que, por sua vez, foram inspiradas nas obras dos criadores do RUP e da UML. Sendo assim, realizaremos a modelagem de um caso de uso a partir da relação estabelecida entre ele e os seus atores, por meio de um diagrama de Caso de Uso e através da representação do fluxo do Caso de uso, por meio do diagrama de Atividade da UML.

Seguindo com nossa proposta, o próximo passo é o de identificação das classes de análise capazes de realizar o comportamento do caso de uso em modelagem. Para isso, nos guiaremos pela proposta de Dogboevi (2020), que utilizou em sua pesquisa o padrão de concepção de software ECF para descobrir as classes de *fronteira*, de *controle* e de *entidade*

que compõem essa etapa de análise. Apresentaremos sucintamente os significados desses tipos de classe nos parágrafos seguintes.

As classes de *entidade* representam todos os elementos do sistema que precisam ser persistidos (se houver). As classes de *controle*, por sua vez, representam o desdobramento do fluxo de interações dos possíveis cenários do caso de uso em modelagem. Elas são responsáveis por dissociar os objetos de entidade e de fronteira, o que torna o sistema mais tolerante a mudanças. Além disso, como afirma Dogboevi, “geralmente encontramos apenas uma classe de controle por cada caso de uso na modelagem, exceto em situações em que esse caso de uso possui um fluxo complexo de eventos ou é manipulado por mais de um ator” (DOGBOEVI, 2020, p. 25, tradução nossa).

Por fim, as classes de *fronteira* se encontram na periferia de um subsistema, isolando-o das alterações em seu ambiente externo e impedindo que essas alterações afetem a estrutura da aplicação (DOGBOEVI, 2020). O autor afirma que existe pelo menos um objeto de fronteira para cada par ator-caso de uso, que será responsável por coordenar a interação entre esse ator e o caso de uso na modelagem.

Após a descoberta das classes necessárias para a realização do caso de uso em modelagem, precisamos descobrir suas propriedades e métodos, de modo a estabelecer relações entre as classes. Dogboevi (2020) apresenta uma maneira bastante intuitiva para essas descobertas, que é realizada a partir do estudo das descrições dos casos de uso, em que suas propriedades e métodos podem ser descobertos a partir, respectivamente, de alguns substantivos e verbos presentes no texto.

Munidos das classes de análise, com seus métodos e suas propriedades, estabeleceremos a relação entre as classes. Essa relação pode ser feita por várias técnicas, em que Dogboevi cita três, que também traremos a seguir como proposta nossa.

Em uma das técnicas, a relação é estabelecida a partir das precondições e pós-condições presentes nos casos de uso do sistema. Outra técnica parte da observação das regras que se aplicam à comunicação entre instâncias de classes organizadas de acordo com o padrão de projeto ECF:

“(1) Os atores do sistema só podem interagir com objetos de fronteira; (2) Objetos de fronteira só podem interagir com atores do sistema e objetos controladores; (3) Objetos de entidade só podem interagir com objetos de controlador, e; (4) Os objetos do controlador só podem interagir com objetos de fronteira, objetos de entidade e outros objetos do controlador (DOGBOEVI, 2020, p. 26, tradução nossa).

Uma última maneira de relacionar as classes de análise do sistema que trazemos como opção é a partir da utilização de diagramas de interação UML, que possibilita um melhor entendimento de como as instâncias de classes colaboram e trocam mensagens, ou transferem informações.

### **5.2.2 Cumprimento das atividades dos detalhes de fluxo “Analisar o comportamento”, “Projetar componentes” e “Projetar o banco de dados”**

Para as atividades relativas aos detalhes de fluxo “Analisar o comportamento”, “Projetar componentes” e “Projetar o banco de dados”, consideraremos as classes de análise obtidas na etapa anterior, contendo métodos, propriedades e relacionamentos entre classes, para avançar na reestruturação dessas classes em elementos de design preparados para serem programados na implementação.

Apresentaremos, por meio de tópicos, as atividades a serem desenvolvidas:

- Incorporar a persistência;
- Definir e detalhar novos atributos e métodos das classes;
- Desenvolver a estrutura da aplicação, por meio de reagrupamento de classes em pacotes lógicos e da transformação de classes de análise em elementos de projeto mais elaborados;
- Identificar oportunidades de reutilização, tanto internamente no sistema em modelagem, por meio da identificação de semelhanças entre classes em desenvolvimento e pré-existentes, quanto externamente, através da identificação de componentes de aplicação existentes em projetos anteriores, ou produtos e componentes desenvolvidos por outras instituições.

### **5.2.3 Considerações sobre a Metodologia**

É importante destacar um aspecto fundamental da nossa proposta de metodologia. Como vimos, o RUP possui uma característica de ser iterativo e incremental, enquanto a EDI não traz esse olhar em sua proposta de ciclo de vida, apesar de possibilitar certa flexibilidade entre suas fases, que busca diferir de um desenvolvimento produzido em *cascata*.

Nossa proposta de inserção de uma metodologia de suporte à etapa de Desenvolvimento da EDI, que se utilize de modelagem de sistema e que se destine à produção de jogos

matemáticos de simulação para situações didáticas, é a de que, nesta etapa específica do modelo de processo da EDI, as ações sejam realizadas iterativamente, ou seja, contamos com a evolução do software em desenvolvimento por etapas de refinamento da arquitetura, da análise, do design e da implementação do sistema, embora possamos considerar que essa proposta já seja contemplada pela EDI, tendo em vista que não há restrições de como os profissionais da engenharia de software (componentes da equipe de desenvolvimento) desenvolveriam suas atividades de programação do software, a partir dos requisitos educativos levantados.

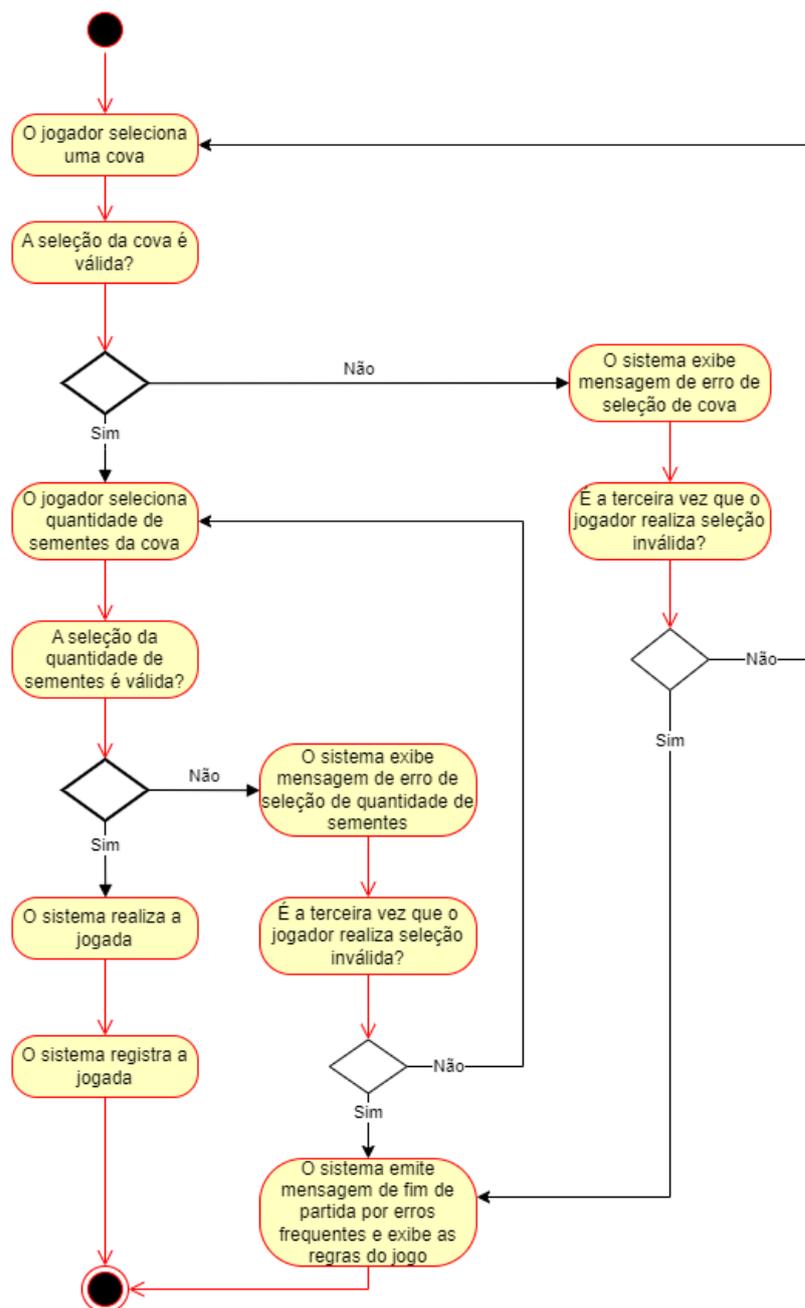
### 5.3 UTILIZAÇÃO DA METODOLOGIA PARA A ETAPA DE DESENVOLVIMENTO DA EDI NA MODELAGEM DO MANKALA COLHE TRÊS DIGITAL

Com o objetivo de exemplificar a metodologia que acabamos de propor, iremos realizar a modelagem do Mankala Colhe Três Digital seguindo as recomendações trazidas na seção anterior. Consideraremos, portanto, que daremos uma continuidade à concepção do Mankala Colhe Três Digital, que é resultado da passagem pelas etapas de Especificação, Análises Prévias, Levantamento de Requisitos, Concepção e Análise *a priori* e Prototipação da Engenharia Didático-Informática.

#### 5.3.1 Modelagem da relação Ator-Caso de Uso por meio do Diagrama de caso de uso

Tendo como artefato de entrada da utilização da Metodologia o documento de requisitos do software, representado pelo Modelo de Casos de Uso do Mankala Colhe Três Digital, modelaremos a relação estabelecida entre o ator “Jogador” e o caso de uso “Realizar Jogada”. Indicamos ao leitor a consulta da especificação do caso de uso “Realizar Jogada”, presente na subseção 4.1.1.3 deste trabalho, que será discutido ao longo de toda esta seção. De todo modo, para ilustrar o fluxo do caso de uso, apresentaremos a seguir um diagrama de atividades da UML:

Figura 27 – Diagrama de atividade do requisito “Realizar Jogada”

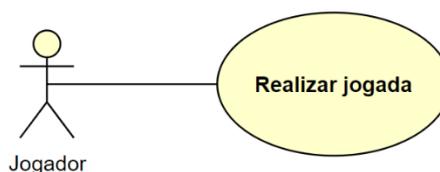


Fonte: elaborada pelo autor.

É importante ressaltar que o caso de uso “Realizar jogada” foi escolhido para ser modelado neste capítulo por se tratar de um caso de uso complexo do sistema, além do fato de representar uma funcionalidade fundamental do jogo, o que, para nós, trará elementos que auxiliarão na compreensão da nossa metodologia.

Apresentaremos a seguir o Diagrama de Casos de Uso, da UML, contendo o ator “Jogador” interagindo com o caso de uso “Realizar Jogada”.

Figura 28 – Diagrama de caso de uso para o requisito “Realizar Jogada”



Fonte: elaborada pelo autor.

### 5.3.2 Descoberta das Classes de análise (Entidade, Controle, Fronteira)

Após a produção do Diagrama de Casos de Uso, é possível identificar a única classe de fronteira de interface entre o ator e o caso de uso na modelagem: “TelaDoJogo”. Após a identificação da classe de fronteira, podemos pensar nas classes de controle e entidade. Chamaremos a classe de controle de “Middleware”, responsável por gerenciar o desencadeamento do fluxo de interações de cenários de casos de uso. Como classe de entidade, temos “TabuleiroEJogada”, que representa os elementos passivos cujas informações precisam ser persistidas de alguma maneira (em arquivos, em um banco de dados etc.).

Segue a representação das classes identificadas em diagrama de classes UML:

Figura 29 – Classes de análise do caso de uso “Realizar Jogada”, segundo o padrão ECF

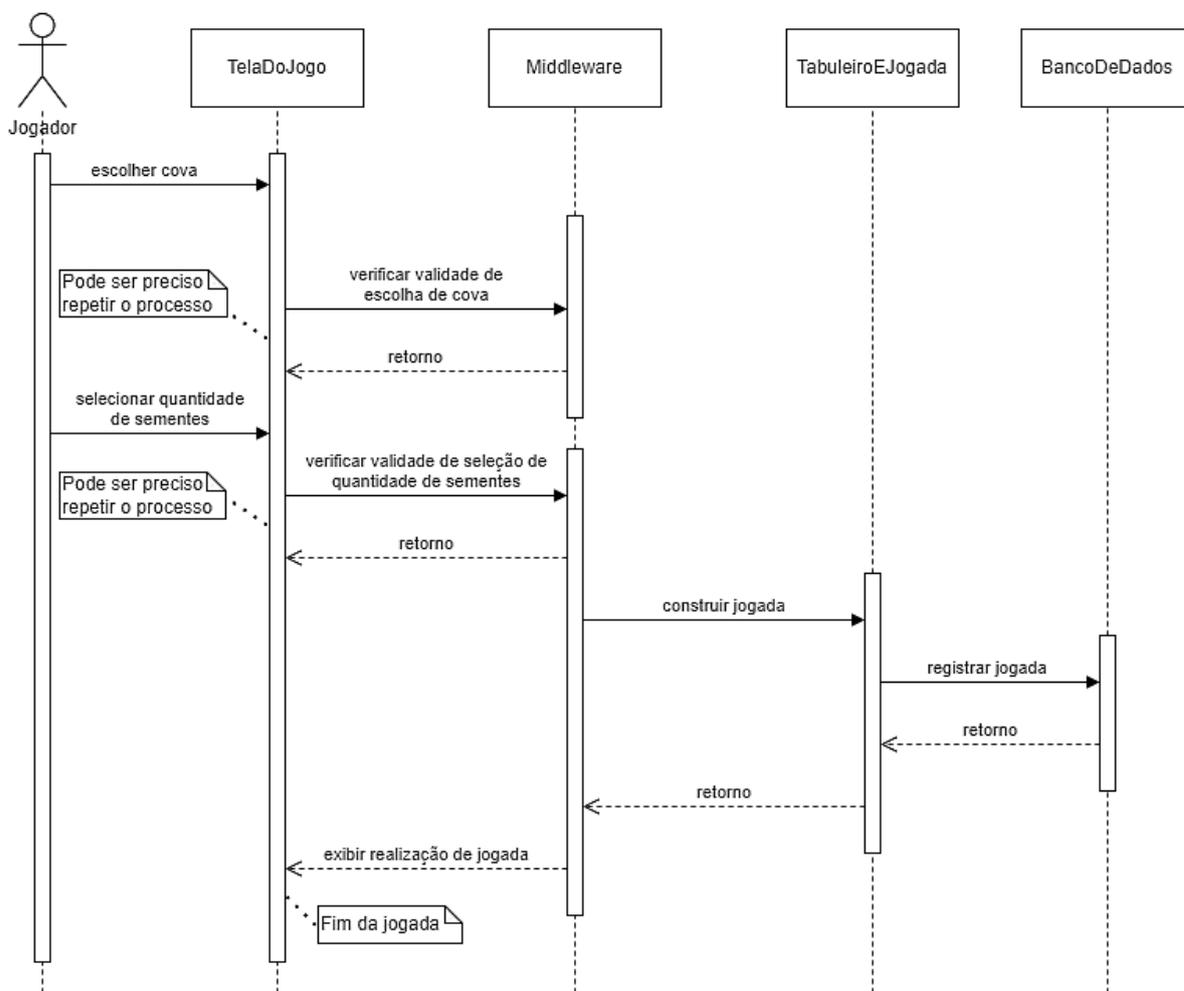


Fonte: elaborada pelo autor.

A partir do momento em que identificamos as classes de análise da modelagem do caso de uso, precisamos descobrir as propriedades e os métodos dessas classes, para que possamos estabelecer relações entre as classes.

Como afirmado na descrição da metodologia, na seção anterior, temos algumas técnicas de obtenção dos métodos e das propriedades das classes. Utilizaremos o diagrama de sequência da UML para representar a realização de uma jogada hipotética, apresentado na figura a seguir:

Figura 30 – Diagrama de sequência para a realização de uma jogada hipotética

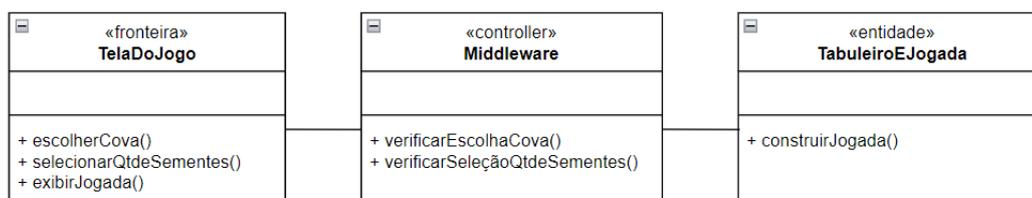


Fonte: elaborada pelo autor.

Podemos perceber que, ao escolher uma cova diretamente na tela do jogo, o Middleware realiza a atividade “Verificar a validade de escolha de cova” e retorna para a tela do jogo resultado dessa atividade. O mesmo acontece com a atividade “Verificar validade de seleção de quantidade de sementes”, no entanto, após a jogada ser validada, é preciso que haja um registro da jogada para que o Professor possa obter o relatório das jogadas, conforme descrito no caso de uso “Acessar relatório de jogadas”, do Modelo de Casos de Uso do Mankala Colhe Três Digital, presente na Concepção e análise *a priori* do jogo.

Como já temos as informações das propriedades e dos métodos das classes de análise, podemos construir o Diagrama de Classes de análise, apresentado a seguir:

Figura 31 – Diagrama de Classes (de análise) para o caso de uso “Realizar Jogada”



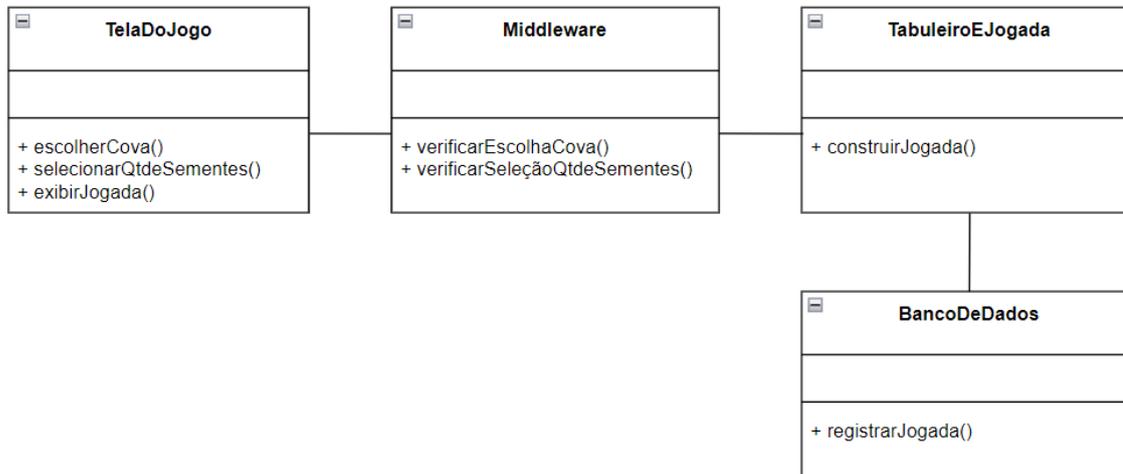
Fonte: elaborada pelo autor.

Concluimos, portanto, mais uma etapa para modelagem do Mankala Colhe Três Digital.

### 5.3.3 Produção das classes de design

Nessa etapa final, iremos incorporar ao nosso diagrama de classes apenas mais uma classe, que permitirá a realização da persistência das informações de jogada sobre uma base de dados.

Figura 32 – Diagrama de classes para o caso de uso “Realizar jogada”



Fonte: elaborada pelo autor.

Com o diagrama de classes de design do caso de uso “Realizar Jogada”, finalizamos nossa exemplificação de uso da Metodologia para a etapa de Desenvolvimento da EDI na modelagem do Mankala Colhe Três Digital. Como estamos propondo uma abordagem iterativa e incremental do uso dessa metodologia, as próximas iterações aprimorarão o design do sistema como um todo, permitindo que sejam elaborados o design de pacotes, subsistemas e, por fim, no Modelo de Design (ou Projeto), contendo todas as classes de design do sistema.

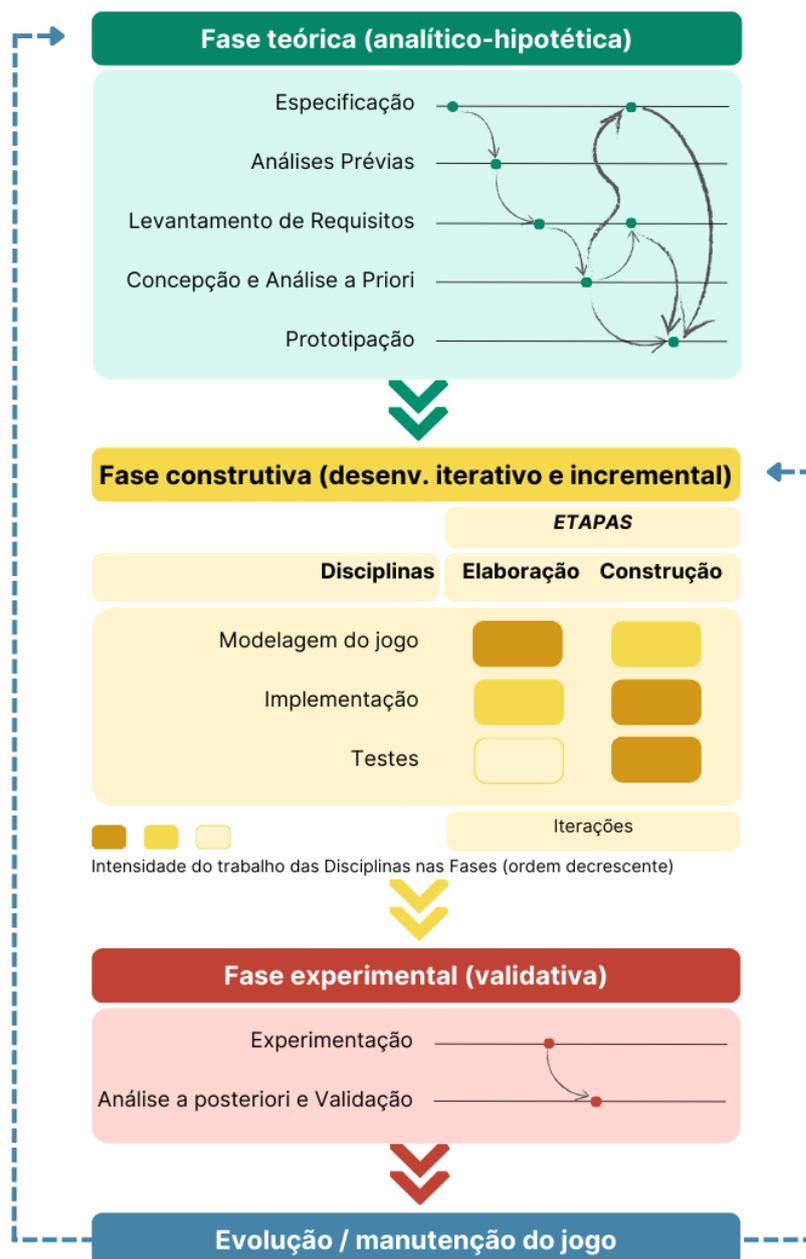
## **6 MODELO DE PROCESSO DA ENGENHARIA DIDÁTICO-INFORMÁTICA DESTINADO À PRODUÇÃO DE JOGOS MATEMÁTICOS DE SIMULAÇÃO PARA SITUAÇÕES DIDÁTICAS**

A realização das etapas da Engenharia Didático-Informática para a concepção do Mankala Colhe Três Digital nos rendeu muitas reflexões ao longo do desenvolvimento da investigação. Considerando nosso objetivo geral da pesquisa, apresentaremos como um dos resultados deste trabalho o modelo de processo de software da Engenharia EDI destinado à produção de jogos matemáticos de simulação para situações didáticas.

Acreditamos que algumas construções que traremos neste capítulo poderão ser consideradas para o modelo de processo geral da EDI. No entanto, reforçamos que, por meio do viés que tomamos ao longo de toda a nossa investigação, centralizaremos nosso olhar para os jogos matemáticos de simulação para situações didáticas.

Como já tivemos algumas oportunidades de discutir os elementos do modelo de processo da EDI, formalizado por Tiburcio (2020), o presente capítulo focará em apresentar a nossa proposta de modelo de processo e detalhar seus elementos, que foram embasados por todas as reflexões realizadas ao longo dessa investigação. Iniciaremos por meio da apresentação do modelo, que segue:

Figura 33 – Modelo de Processo da EDI destinado a jogos matemáticos de simulação para situações didáticas



Fonte: elaborada pelo autor.

Inicialmente, traremos informações sobre como interpretar graficamente nosso modelo. Ele é dividido em três fases e uma etapa adicional de transição, todas relacionadas entre si. Cada fase possui etapas específicas e um fluxo de realização dessas etapas internamente à fase.

É possível perceber que o fluxo das fases teórica e experimental, direcionado por setas, se dá temporalmente da esquerda para a direita e permite revisitações de etapas internas à fase. O comportamento da fase construtiva será detalhado mais adiante.

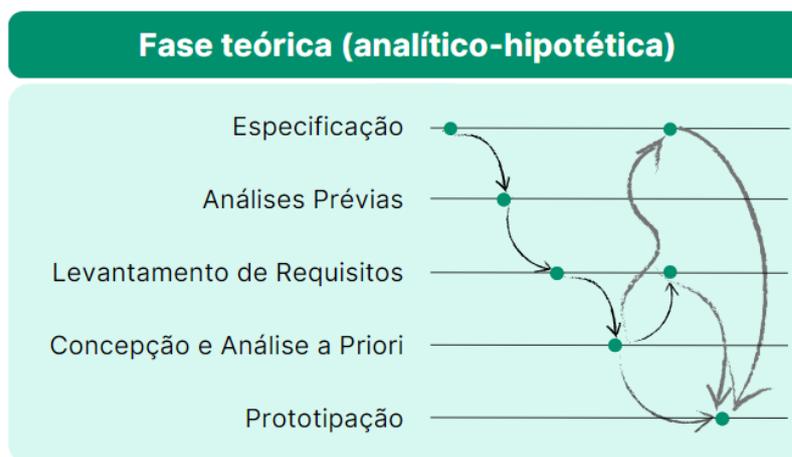
As setas externas do modelo informam que a realização da *fase teórica* conduz à realização da *fase construtiva* que, por sua vez, subsidia a realização da *fase experimental*. Por meio resultados da fase experimental, obtém-se informações para a *evolução e manutenção do software*, que poderá propor mudanças de cunho teórico e/ou construtivo.

Ainda em relação às características de cada fase, veremos que elas possuem marcos, representados por artefatos que servirão de suporte à realização da fase seguinte.

## 6.1 FASE TEÓRICA (ANALÍCO-HIPOTÉTICA)

A fase teórica do nosso modelo de processo, que possui etapas de natureza analítico-hipotética, é composta pelas etapas: Especificação, Análises Prévias, Levantamento de Requisitos, Concepção e Análise *a priori* e Prototipação, como podemos verificar na figura a seguir.

Figura 34 – Fase teórica (analítico-hipotética)



Fonte: elaborada pelo autor.

Consideramos as etapas de Especificação, Análises Prévias similares a proposta de Tiburcio (2020), destacando certa flexibilidade na composição da equipe transdisciplinar de desenvolvimento e a importância do olhar para as noções *matemáticas*, *paramatemáticas* e *protomatemáticas* ao longo desse estudo inicial. Quanto à etapa de Levantamento de Requisitos, como vimos a necessidade de realizar uma distinção entre os requisitos educativos e os computacionais, essa etapa fornece o leque de requisitos educativos, mesmo que possuam um olhar para a dimensão informática. Sendo assim, propomos que seja realizada uma

sistematização dos requisitos educativos organizados por dimensões, conforme proposto por Tiburcio (2020).

A etapa de Concepção e Análise *a priori* carrega o mesmo significado presente na versão atual do modelo de processo geral da EDI. Contudo, propomos que ela seja realizada por meio da produção do Modelo de Casos de Uso do jogo. Esse documento se mostra bastante adequado no detalhamento dos requisitos funcionais do jogo digital, além de conter os primeiros requisitos não funcionais, fornecendo, desta forma, um primeiro olhar computacional ao sistema.

A última etapa da fase teórica é a de Prototipação. Como discutimos bastante na seção referente à realização desta etapa, há um duplo significado dessa expressão ao longo do texto de Tiburcio (2020), sendo, em alguns momentos, compreendido como uma sistematização dos requisitos educativos representados por documentos que ilustram como seriam as telas do software (“prototipação em telas”), e em outros momentos, representando uma versão inicial do software minimamente desenvolvido a ser experimentado por usuários em potencial.

Utilizaremos o termo prototipação, como etapa da fase teórica, no sentido de sistematização dos requisitos educativos “em telas”, pois acreditamos que a realização de uma versão beta do sistema precisa ser realizada nas etapas da fase seguinte (construtiva), nas primeiras iterações do projeto. É importante destacar que, apesar de acreditarmos que o Modelo de Casos de Uso e os diagramas UML podem se mostrar suficientes na sistematização e apresentação dos requisitos educativos do software, acreditamos que o aspecto visual da interface do usuário possui um papel importante para os jogos matemáticos de simulação para situações didáticas, principalmente no que se refere à dimensão lúdica do jogo digital.

Finalizando a análise da fase teórica, iremos nos debruçar sobre o fluxo das atividades presente no nosso modelo. É possível perceber que existe um caminho natural iniciado pela especificação do jogo, seguindo com os estudos das Análises Prévias, que fornecerá um leque de aspectos importantes para o levantamento de requisitos educativos.

Após o levantamento de requisitos, na idealização das situações de uso e levantamento das hipóteses referentes ao atendimento das questões envolvendo ensino e aprendizagem, Tiburcio (2020) enfatiza que pode haver ajustes na especificação do software ou nos requisitos levantados. Concordando com Tiburcio, ilustramos essas possibilidades no modelo com o intuito de esclarecer os caminhos mais prováveis de retornos às etapas anteriores na fase teórica.

O marco da realização da fase teórica é a produção do Modelo de Casos de Uso do jogo, contendo o Diagrama de Casos de Uso da UML, e a produção da “Prototipação em telas”.

## 6.2 FASE CONSTRUTIVA (DESENVOLVIMENTO)

A fase construtiva, cujo enfoque é o desenvolvimento computacional do jogo, contém as etapas Elaboração e Construção do jogo, realizadas por meio das disciplinas de Modelagem do jogo, Implementação e Testes. Essa fase é iniciada a partir dos artefatos oriundos da fase teórica, para subsidiar o início etapa de elaboração.

A modelagem do jogo é realizada a partir da metodologia apresentada no Capítulo 5 deste trabalho, em que, por meio de desenvolvimento iterativo e incremental, realiza-se uma sequência de atividades de modo a transformar os requisitos advindos do Modelo de Casos de Uso do jogo em um modelo de projeto, contendo todas as classes de design do sistema.

A Implementação e os Testes também são realizados de forma iterativa e incremental e representam disciplinas inspiradas no modelo de processo RUP. É possível perceber, na Figura 34, que o desenvolvimento do jogo seguirá a ordem das fases de elaboração e construção por meio da realização das atividades das disciplinas, que, por sua vez, possuem ênfases diferentes em cada etapa do processo.

Figura 35 – Fase construtiva (desenvolvimento iterativo e incremental)



Fonte: elaborada pelo autor.

Analisando a realização das atividades das disciplinas no processo da Figura 34, percebemos que o início dessa fase possui maior ênfase nas atividades de Modelagem do jogo, que continuam acontecendo na etapa de construção, mas de maneira menos intensa. Já a disciplina de Implementação, mesmo que possua um foco mediano na etapa de elaboração,

principalmente no desenvolvimento de um protótipo focado na interface do usuário, sua maior ênfase é dada na etapa de construção.

Quanto à disciplina de Testes, podemos perceber que ela não se faz presente durante a etapa de Elaboração, já que, nesse momento, o foco maior é dado à modelagem do sistema, e tem seu lugar marcado na etapa de Construção do jogo.

É importante destacar que consideramos as atividades da disciplina de Testes como sendo testes computacionais realizados pela equipe de desenvolvimento e por usuários em potencial, com o objetivo de verificar possíveis falhas de implementação e de funcionalidades.

O marco da realização da fase construtiva é a produção de uma versão beta do sistema, a ser experimentada didaticamente na fase seguinte.

### 6.3 FASE EXPERIMENTAL (VALIDATIVA) E ETAPA DE EVOLUÇÃO

A fase Experimental possui as etapas de Experimentação, *Análise a posteriori* e Validação.

Figura 36 – Fase experimental (validativa)



Fonte: elaborada pelo autor.

Consideramos a realização dessas etapas de forma similar ao proposto por Tiburcio (2020) apresentando apenas duas ressalvas. A primeira é que escolhemos por considerar a *Análise a posteriori* e a Validação em uma única etapa, devido ao grau de dependência entre elas. A outra ressalva que destacamos é a distinção entre os testes do software, computacionalmente falando, e o experimento nos termos da Engenharia Didática, para tentar evitar possíveis incompreensões sobre o que chamaríamos de experimentação.

Sobre essa segunda ressalva, no nosso modelo, consideramos que os testes do software, presente na fase construtiva, considera a equipe de desenvolvedores de usuários em potencial, como professores e alunos, e que, através dos *feedbacks* de uso, realiza ajustes a cada nova iteração. O que consideramos como experimentação, presente nesta fase experimental, é o uso

do jogo digital, minimamente validado em uma versão beta do sistema, em situações de ensino e aprendizagem preparadas pelo professor, com o fim de analisar os diferentes usos do recurso digital e, com isso, auxiliar os educadores matemáticos por meio de propostas validadas de uso do jogo digital.

O marco da realização da fase Experimental é o conjunto de informações advindas da Análise *a posteriori* e Validação com características de modificações do jogo e de seu uso.

Por fim, a etapa de Evolução/Manutenção foi mantida conforme a proposta apresentada na versão da EDI de Tiburcio (2020).

Como podemos verificar no Modelo de Processo da EDI destinado a jogos matemáticos de simulação para situações didáticas, apresentado na Figura 32, destacamos que a Evolução do software pode considerar aspectos tanto da fase teórica quanto da fase construtiva.

## 7 CONSIDERAÇÕES FINAIS

Ao buscar contribuir com debate acerca da produção de softwares educativos de qualidade, nos motivamos a dar seguimento ao trabalho de Tiburcio (2020), que tem centrado suas pesquisas em torno da formalização e aperfeiçoamento da Engenharia Didático-Informática (EDI).

A EDI surgiu a partir da busca pela articulação entre os princípios teórico-metodológicos da Engenharia Didática (ARTIGUE, 1996) e a área da Engenharia de Software, com o intuito de construir uma metodologia de desenvolvimento de softwares educativos que considere os aportes da Engenharia Didática na Engenharia de Software.

Em sua pesquisa de doutorado, Tiburcio (2020) apresentou uma versão atualizada da Engenharia Didático-Informática, contendo um modelo de processo de software educativo. Essa pesquisa gerou alguns encaminhamentos para pesquisas futuras, dentre as quais estavam o estímulo à utilização dessa nova versão da EDI e a motivação do uso da EDI para a produção de softwares educativos de tipologias diferentes dos ambientes de simulação/micromundos, já que as primeiras experiências da EDI focaram com esse tipo de software.

Vimos a oportunidade de dar seguimento aos encaminhamentos levantados por Tiburcio (2020) a partir da concepção de uma versão digital do jogo matemático Mankala Colhe Três. A escolha do jogo a ser utilizado se deu, dentre outros motivos, pelo fato de que Santos (2014) havia desenvolvido uma pesquisa de mestrado que buscou investigar o potencial didático do Mankala Colhe Três e, em seus encaminhamentos para pesquisas futuras, ele apresentou vários aspectos do jogo e de seu uso em situações didáticas que, para nós, poderiam ser considerados em uma versão digital do jogo.

Partimos, portanto, dessa motivação e demos início a esta pesquisa, buscando responder a questões como: “*quais as primeiras impressões da utilização dos fundamentos da EDI em sua nova versão?*”, “*há sugestões de atualizações da EDI?*”, “*quais elementos do modelo de processo de software da EDI precisarão ser reconsiderados para o caso da produção de jogos matemáticos de simulação para situações didáticas?*”.

Como hipótese, acreditamos que a concepção de uma versão digital do jogo Mankala Colhe Três, baseada nos princípios teórico-metodológicos da Engenharia Didático-Informática, forneceria elementos suficientes para a proposição de um novo modelo de processo de software da EDI destinado à produção de jogos matemáticos de simulação para uso em situações didáticas.

Traçamos, como objetivo geral da pesquisa, a proposição de um modelo de processo de software da Engenharia Didático-Informática destinado à produção de jogos matemáticos de simulação a serem utilizados como instrumentos centrais na vivência de situações didáticas.

Sobre os objetivos específicos da pesquisa, trouxemos: *realizar um estudo do Mankala Colhe Três e das noções que podem ser mobilizadas por meio do jogo, considerando aspectos de natureza epistemológica, cognitiva, didática e informática, de modo a estabelecer requisitos educativos para uma versão digital do jogo; identificar as particularidades da utilização dos princípios da Engenharia Didático-Informática na concepção de uma versão digital do Mankala Colhe Três; e apresentar contribuições metodológicas para a Engenharia Didático-Informática.*

Para atender aos objetivos levantados para a pesquisa, nossa metodologia foi conduzida por meio da concepção de uma versão digital do jogo Mankala Colhe Três, seguindo a realização das fases da EDI, de modo a obter informações sobre particularidades do processo de desenvolvimento de jogos matemáticos de simulação para situações didáticas e, assim, propor um modelo de processo específico para a produção desse tipo de software educativo.

Durante a realização das etapas da EDI para a concepção do Mankala Colhe Três Digital, realizamos reflexões acerca do uso dessa metodologia etapa por etapa. Iniciamos pela Especificação, seguimos pelas Análises Prévias, realizamos o Levantamento de Requisitos, a Concepção e Análise *a priori* e discutimos acerca da Prototipação e do Desenvolvimento.

Sobre a realização da etapa de Especificação, destacamos a existência de algumas particularidades na consideração dessa etapa em comparação ao modo como ela foi tratada nos demais softwares educativos que fizeram o uso da EDI na produção de ambientes de simulação/micromundos.

No caso da concepção de uma versão digital do Mankala Colhe Três, percebemos inicialmente que os conhecimentos que pretendíamos abordar por meio do jogo deveriam se basear não apenas nas análises de conhecimentos matemáticos delimitados, mas também em uma análise que incluía as noções relacionadas ao jogo como recurso pré-existente fisicamente, que deveria considerar importantes aspectos educativos que não estão relacionados, necessariamente, aos conhecimentos matemáticos, como o aspecto lúdico e cultural do jogo.

Além disso, tendo em vista os resultados apresentados por pesquisadores que se debruçaram sobre o Mankala Colhe Três, vimos a necessidade de refinar o olhar para as noções suscetíveis de serem mobilizadas através de situações com o jogo por meio da distinção entre noções matemáticas, paramatemáticas e protomatemáticas.

Ainda em relação à etapa de Especificação, destacamos que o diferencial do Mankala Colhe Três Digital em comparação ao ambiente papel-lápis, ou à sua versão física, está associado diretamente aos resultados da pesquisa de Santos (2014), que mostrava tanto o potencial educativo do jogo físico quanto fornecia um leque de informações que, para nós, se enquadrava em um esboço inicial dos requisitos para uma versão digital, com seu potencial educativo ampliado.

Sobre o público-alvo a quem se destina o Mankala Colhe Três Digital, nossa posição foi de não limitar o uso do jogo para estudantes de anos escolares específicos, pois consideramos que as adaptações presentes na versão digital do jogo, em termos estruturais do jogo e metodológicas de seu uso, certamente ampliarão seu público-alvo em comparação à versão física, que, por sua vez, já possui resultados consolidados no uso por estudantes dos anos iniciais e de 6º ano do Ensino Fundamental (ANDRADE *et al.*, 2013; SANTOS, 2014). Além disso, nossa proposta de jogo permitirá seu uso *offline* por jogadores em situações em que não haja intenções didáticas, focando em seu aspecto lúdico a um público-alvo ainda mais abrangente.

Sobre a composição da equipe de desenvolvimento, a ser detalhada na etapa de Especificação, consideramos que o olhar transdisciplinar na produção dos jogos matemáticos de simulação para situações didáticas poderá ser focado não necessariamente em termos da presença de uma ampla equipe de profissionais envolvidos, mas em uma equipe que domine com competência os temas das diversas áreas envolvidas no projeto, propiciando uma integração sólida dessas áreas. Essa escolha se atrela à ampliação da viabilidade do uso da EDI.

Em relação às Análises Prévias, por nos basearmos nas considerações sobre a Especificação do Mankala Colhe Três Digital, acreditamos que a base do levantamento deveria se focar tanto nos vários aspectos que envolvem o uso do jogo como recurso para o ensino e a aprendizagem de matemática quanto em seu potencial educativo como um todo.

Decidimos, portanto, realizar nosso estudo analítico por meio de seções temáticas, que apresentam de forma integrada encaminhamentos das dimensões epistemológica, cognitiva, didática, informática, entre outras. As seções temáticas trazidas nas Análises Prévias foram: *Os jogos matemáticos na Educação Matemática; O Mankala Colhe Três; A Teoria das Situações Didáticas e a análise do potencial educativo do Mankala Colhe Três; A Transposição Didático-Informática do Mankala Colhe Três.*

A realização da etapa “Levantamento de Requisitos” foi direcionada pelas quatro fases propostas por Tiburcio (2020). A primeira delas é conduzida por questionamentos relacionados ao favorecimento do ensino e da aprendizagem de conhecimentos matemáticos por meio do uso do jogo. Sobre essa fase do levantamento de requisitos, nossas considerações foram que o

ensino e a aprendizagem das noções a serem mobilizadas pelo uso do jogo podem ser favorecidos ao considerarmos os aspectos trazidos nos resultados das pesquisas envolvendo o Mankala Colhe Três em sua versão original (física) (ANDRADE *et al.*, 2013; SANTOS, 2014).

Continuando nas considerações sobre a primeira fase do levantamento de requisitos, concluímos que uma versão digital do Mankala Colhe Três que leve em consideração os encaminhamentos de pesquisas futuras apontados por Santos (2014) permitirá uma ampliação do potencial educativo do jogo, já que a produção do jogo por um viés computacional trará uma diversidade de novas possibilidades de situações de uso.

A segunda fase do levantamento de requisitos do software se refere à uma análise externa. Nesse caso, como o Mankala Colhe Três Digital se baseou em sua versão física para a realização dessa análise, afirmamos que o que propomos de diferencial da versão digital deveria vir dos encaminhamentos futuros de Santos (2014), tendo em vista que consideramos que cada sugestão levantada pelo autor seria atendida na versão digital do jogo.

A terceira fase, que busca sistematizar os requisitos do software no Documento de Requisitos conforme descrito por Tiburcio (2020), nos fez refletir sobre a necessidade de uma distinção entre requisitos educativos e requisitos computacionais para a produção do jogo. Sendo assim, elaboramos dois documentos de requisitos distintos: o Documento de Requisitos em forma de quadro que traz a categorização dos requisitos educativos por dimensões; e o Modelo de Casos de Uso do Mankala Colhe Três, baseado nos princípios propostos pelo Processo Unificado da *Rational* (RUP), que apresenta detalhadamente os requisitos funcionais e não funcionais do sistema.

Ao analisar as contribuições advindas das Análises Prévias para a produção do quadro de requisitos educativos, achamos mais adequado categorizar esses requisitos em três grupos: o bloco *cognitivo-epistemológico*; o bloco *didático-informático*; e a *dimensão sociocultural*. Consideramos dessa forma por percebermos uma conexão entre os requisitos oriundos das dimensões cognitiva e epistemológica e das dimensões didática e informática. Além disso nos deparamos com a necessidade da criação de uma nova dimensão que considere os aspectos das origens do Mankala Colhe Três.

Sobre a última fase do levantamento de requisitos, que trata da verificação dos requisitos quanto à pertinência, consistência e integralidade, concluímos que as ações referentes à essa fase devem ser realizadas ao longo do processo de produção do jogo, concordando com as orientações de Tiburcio (2020) acerca da necessidade de que o gerenciamento de requisitos seja realizado durante todo o ciclo de vida do software.

A etapa de Concepção e Análise *a priori* foi realizada por meio da produção do Modelo de Casos de Uso do Mankala Colhe Três Digital. Esse documento se mostrou muito adequado para a sistematização dos requisitos funcionais e não funcionais do sistema além de apresentar muita afinidade com o que preconiza esta etapa da EDI. Neste documento, foram apresentados: o histórico de alterações; o Diagrama de Casos de Uso da Linguagem de Modelagem Unificada (UML); os requisitos funcionais; os requisitos não funcionais.

Tendo em vista que a etapa de Prototipação da EDI representa uma etapa de transição entre as fases hipotética e experimental, percebemos que ela se relaciona fortemente com as etapas de Concepção e Análise *a priori* e de Desenvolvimento e Experimentação do software. Ao analisar a forma como Tiburcio (2020) propõe a realização da Prototipação, percebemos certa ambiguidade do termo para representar tanto uma sistematização “em telas” das idealizações de uso do software, advindas da etapa de Concepção e Análise *a priori*, quanto uma versão inicial do software desenvolvido, suscetível de experimentações por usuários em potencial.

Decidimos, portanto, considerar a realização da etapa de Prototipação pelas duas perspectivas: a da sistematização das funcionalidades do sistema, que, no nosso caso, ao invés de produzirmos a “prototipação em telas”, apresentamos o Modelo de Casos de Uso do jogo; e a que representa a construção de uma versão inicial do sistema a ser detalhada nas discussões acerca da etapa de Desenvolvimento, tendo em vista que a produção de uma versão inicial do jogo minimamente desenvolvido requer, da equipe de desenvolvimento, um olhar voltado aos aspectos computacionais de desenvolvimento de softwares.

A realização da etapa de Desenvolvimento da EDI, que representa o design e a arquitetura do software, nos redirecionou à problemática envolvendo a composição da equipe de desenvolvimento em uma perspectiva transdisciplinar, iniciada na Especificação do Mankala Colhe Três Digital.

Consideramos que a difusão e a utilização da EDI podem estar sendo dificultadas pela carência de profissionais para compor essa equipe transdisciplinar, principalmente no que se refere à disponibilidade, em ambientes científicos, de se contar com profissionais da Engenharia de Software que possuam um perfil profissional que favoreçam o diálogo com os profissionais da Educação Matemática, área na qual a EDI tem se destacado como fundamentação metodológica para a produção de softwares educativos.

Diante dessa problemática, nos deparamos com a necessidade de contribuições para a fase de Desenvolvimento da EDI, de modo a melhorar a viabilidade do uso dessa metodologia de desenvolvimento de software, especialmente por educadores matemáticos. Essa necessidade

nos gerou um novo objetivo de pesquisa, que descrevemos como “*construir uma metodologia para a etapa de Desenvolvimento da EDI que forneça um percurso para transformação dos requisitos em código a ser implementado no desenvolvimento de jogos matemáticos de simulação para situações didáticas*”.

A partir de então, construímos uma metodologia de suporte à etapa de Desenvolvimento da EDI para a produção de jogos matemáticos de simulação para situações didáticas, apoiada no modelo de Processo Unificado da *Rational* (RUP) e na Linguagem de Modelagem Unificada (UML), que auxilie a transformação dos requisitos elencados nas etapas iniciais da EDI em artefatos oriundos da UML, que se aproximem ao máximo do código final a ser implementado.

Nossa proposta de metodologia se baseia nas atividades pertencentes à disciplina *análise e design* do RUP para a modelagem do jogo, por meio do uso de diagramas visuais da UML em procedimentos detalhados, de modo a partirmos do Modelo de Casos de Uso do jogo como artefato de entrada e chegarmos ao Diagrama de Classes de Design do sistema, se aproximando ao máximo da implementação.

Após a construção da metodologia, exemplificamos seu uso por meio da modelagem parcial do Mankala Colhe Três Digital. Essa metodologia possui a característica de considerar aspectos importantes do RUP, como o fato de ser iterativa e incremental. Sendo assim, a modelagem parcial realizada no Mankala Colhe Três Digital foi feita por meio da modelagem de uma funcionalidade do sistema, representando a realização de uma iteração inicial da modelagem do sistema como um todo.

A realização de novas iterações na modelagem do Mankala Colhe Três Digital, que considere as demais funcionalidades do sistema e os relacionamentos entre suas classes de design, comporão a modelagem completa do jogo. Essas próximas etapas representam o primeiro encaminhamento para futuras pesquisas desta investigação.

Ao considerar todos os elementos discutidos ao longo desta pesquisa, tanto a partir da realização das etapas iniciais da EDI na concepção do Mankala Colhe Três Digital, quanto por meio das contribuições trazidas pela proposição da metodologia de auxílio à etapa de Desenvolvimento da EDI, apresentamos, no último capítulo do trabalho, o Modelo de Processo da Engenharia Didático-Informática destinado à produção de jogos matemáticos de simulação para situações didáticas.

Acreditamos que a realização dessa pesquisa trouxe muitas contribuições para a Engenharia Didático-Informática, tanto para as futuras pesquisas envolvendo a produção de jogos matemáticos de simulação para situações didáticas, a partir do modelo de processo que propusemos, quanto para a EDI como um todo, pois acreditamos que muitos resultados obtidos

no âmbito da presente pesquisa poderão ser considerados nas investigações envolvendo a produção de softwares educativos em geral.

Partindo do que foi dito no parágrafo anterior, destacamos um segundo encaminhamento para futuras pesquisas, que é a proposição de atualizações do Modelo de Processo geral da EDI advindas das contribuições apresentadas pelo nosso modelo de processo destinada aos jogos matemáticos de simulação para situações didáticas, principalmente no que se refere à etapa de Desenvolvimento, visto que a presente pesquisa trouxe muitas contribuições teóricas do campo da Engenharia de Software ao longo da produção e exemplificação de uso da Metodologia para a etapa de Desenvolvimento da EDI na produção de jogos matemáticos de simulação para situações didáticas.

Gostaríamos de finalizar nossas considerações trazendo mais um encaminhamento para futuras pesquisas, que é a utilização do modelo de processo apresentado neste trabalho em outros jogos matemáticos de simulação para situações didáticas.

Percebemos, ao longo da realização desta pesquisa, que essa experiência trouxe muitas reflexões que, por sua vez, geraram outras tantas contribuições para fomentar o debate acerca da qualidade da produção de softwares educativos para o ensino e a aprendizagem de Matemática, especialmente em torno da Engenharia Didático-Informática. Por isso, acreditamos que o uso do modelo de processo proposto nesta investigação em investigações futuras poderá fortalecer tanto o próprio modelo, por meio de indicações de atualizações, quanto a Educação Matemática, tendo em vista que, como discutimos amplamente, carecemos de uma diversidade de softwares matemáticos de qualidade para o favorecimento do ensino e da aprendizagem.

## REFERÊNCIAS

- ALMOULOUD, S. A. Didática e concepção de dispositivos informáticos educacionais. **Revista de Informática Aplicada**, São Caetano do Sul, v. 3, n. 1, 03-10, jan./jun. 2007. Disponível em: [https://seer.uscs.edu.br/index.php/revista\\_informatica\\_aplicada/article/view/270/249](https://seer.uscs.edu.br/index.php/revista_informatica_aplicada/article/view/270/249). Acesso em: 29 jul. 2023.
- ALMOULOUD, S. A.; COUTINHO, C. Q. S. Engenharia Didática: características e seus usos em trabalhos apresentados no GT-19/ANPEd. **Revemat: Revista Eletrônica de Educação Matemática**, v. 3, n. 1, 2008.
- ANDRADE, J. LEAL, Y. MONTEIRO, A. ANDRÉ, R. MACLYNE, D. TELES, R. GITIRANA, V. Projeto Rede: Jogos na Educação Matemática, UFPE. **Vídeo de confecção do jogo**, 2011. Disponível em: <http://www.authorstream.com/Presentation/veronica.gitirana-955051-confecao-da-mankala-colhe-tr-s-com-sucata/>. Acesso em: 20 mar. 2022.
- ANDRADE, J. P. G.; LEAL, Y.; MONTEIRO, A.; ANDRE, R.; MACLYNE, D.; TELES, R.; GITIRANA, V. Mankala Colhe Três. In: GITIRANA, V.; TELES, R. A. M.; BELLEMAIN, P. M. B.; CASTRO, A.T.; ALMEIDA, I.A.C.; LIMA, P. F.; BELLEMAIN, F. (Org.). **Jogos com Sucata na Educação Matemática**. 1. ed. Recife: Editora UFPE, 2013. Cap 2, p. 25-38.
- ARTIGUE, M. Engenharia Didática. In: BRUN, J. **Didáctica das Matemáticas**. 1. ed. Lisboa: Instituto Piaget, 1996. v. 1. cap. 4, p. 193-217.
- BALACHEFF, N. La transposition informatique. Note sur un nouveau problème pour la didactique. In : ARTIGUE, M.; GRAS, R.; LABORDE, C.; TAVIGNOT, P. (eds) **Vingtans de didactique des mathématiques en France**. Grenoble : La pensée sauvage éditions, 1994. p. 364-370.
- BELLEMAIN F., **Conception, réalisation et expérimentation d'un logiciel d'aide à l'enseignement de la géométrie, Cabri-géomètre**. 1992. Tese (Doutorado em Didática da Matemática) - Université Joseph Fourier, Grenoble, 1992.
- BELLEMAIN, F. A transposição informática na engenharia de softwares educativos. In: SEMINÁRIO INTERNACIONAL DE PESQUISA EM EDUCAÇÃO MATEMÁTICA, 1., 2015, Serra Negra. **Livro de resumos [...]**. Serra Negra: Sociedade Brasileira de Educação Matemática, 2000.
- BELLEMAIN, F.; RAMOS C.; dos SANTOS, R. Engenharia de Softwares Educativos, o caso do Bingo dos Racionais. In: VI SIPEM - Seminário Internacional de Pesquisa em Educação Matemática, Pirenópolis. **Anais do VI SIPEM**. Brasília: Sociedade Brasileira de Educação Matemática, v. 1, p. 1-12, 2015.
- BÍBLIA, N. T. Colossenses. In BÍBLIA. Português. **Bíblia de Estudo de Genebra**. 2ª ed. Texto bíblico: Almeida Revista e Atualizada. Barueri, SP: Sociedade Bíblica do Brasil; São Paulo: Cultura Cristã, 2009. p. 1591-1601.

BITTAR, M. Contribuições da teoria das situações didáticas e da engenharia didática para discutir o ensino de Matemática. *In: TELES, R. A. M.; BORBA, R. E. S. R, MONTEIRO, C. E. F. (org) Investigações em Didática da Matemática*. Recife: Editora UFPE, 2017.

BOOCH, G.; RUMBAUGH, J.; JACOBSON, I. **UML – Guia do usuário**. 2. ed. (traduzida). Rio de Janeiro: Campus, 2006.

BRASIL. Ministério da Educação. **Secretaria de Educação Fundamental. Parâmetros Curriculares Nacionais: Matemática (3º e 4º ciclos do Ensino Fundamental)**. Brasília: SEF/MEC, 1998.

BRASIL. **Lei nº 10.639, de 9 de janeiro de 2003**. Altera a Lei nº 9.394, de 20 de dezembro de 1996, que estabelece as diretrizes e bases da educação nacional, para incluir no currículo oficial da Rede de Ensino a obrigatoriedade da temática "História e Cultura Afro-Brasileira", e dá outras providências. Publicado no DOU de 10/1/2003. Disponível em: <http://www3.dataprev.gov.br/SISLEX/paginas/42/2003/10639.htm>. Acesso em: 20 fev. 2022.

BRASIL **Lei nº 11.645, de 10 março de 2008**. Altera a Lei no 9.394, de 20 de dezembro de 1996, modificada pela Lei nº 10.639, de 9 de janeiro de 2003, que estabelece as diretrizes e bases da educação nacional, para incluir no currículo oficial da rede de ensino a obrigatoriedade da temática – História e Cultura Afro-Brasileira e Indígena. Publicado no DOU de 11.3.2008. Disponível em: [http://www.planalto.gov.br/ccivil\\_03/\\_Ato2007-2010/2008/Lei/L11645.htm](http://www.planalto.gov.br/ccivil_03/_Ato2007-2010/2008/Lei/L11645.htm). Acesso em: 20 fev. 2022.

BRITO, R. S. **Uma Proposta par Modelagem de Requisitos Não-Funcionais em Projetos Ágeis**. 2010. Dissertação (Mestrado em Ciência da Computação) - Universidade Federal de Pernambuco, Recife, 2010.

BROUSSEAU, G. **La théorie des situations didactiques**. Curso ministrado na atribuição do título de Doutor *Honoris Causa* da Université de Montréal. 1997.

BROUSSEAU, G. Fundamentos e Métodos da Didática da Matemática. *In: BRUN, J. (Org.). Didática das Matemáticas*. Lisboa: Instituto Piaget, 1996. p. 35-113.

BROUSSEAU, G. **Introdução ao estudo das Situações Didáticas**: conteúdos e métodos de ensino. Tradução de Camila Bogéa. São Paulo: Ática, 2008.

CÂMARA, L. T. **Mancala, um jogo milenar, contribuindo na alfabetização matemática de jovens e adultos**. 2006. Monografia (Graduação em Matemática) - Universidade Católica de Brasília, 2006. Disponível em: <https://repositorio.ucb.br:9443/jspui/bitstream/10869/1774/1/Luciene%20Tavares%20da%20Camara.pdf>. Acesso em: 24 mar. 2022.

CHEVALLARD, Yves. **La tranposición didáctica: Del saber sabio al saber enseñado**. Tradução: GILMAN, C. Buenos Aires: Editora Aique, 1991.

CONFREY, J.; SMITH, E. Function Probe: Multi-Representational Software for Learning about Functions. *In: MALCOM, S.; ROBERTS, L.; SHEINGOLD, K. (orgs.). New York State Association for Computers and Technology in Education Journal*. Washington, DC: American Association for the Advancement of Science. 1992.

D'AMBROSIO, U. Sociedade, cultura, matemática e seu ensino. **Educação e Pesquisa**, São Paulo, v. 31, n. 1, p. 99-120, jan./abr. 2005.

DOGBOEVI, K. H. **MeCCIAH – Méthodologie pour la Conception de Contrats Intelligents pour des Architectures Hybrides**. 2020. Dissertação (Mestrado em Ciências e Tecnologia da Informação) - Université du Québec en Outaouais, Gatineau, 2020.

DOMINGUES, H. H. **Fundamentos da Aritmética**. São Paulo: Atual, 1991. 297 p.

GALVIS, A. H. **Ingeniería de Software Educativo**. Santafé de Bogotá: Ediciones Uniandes, 1992.

GITIRANA, V.; TELES, R.; BELLEMAIN, P.; CASTRO, A.; CAMPOS, I.; LIMA, P.; BELLEMAIN, F. (Orgs.). **Jogos com sucata na Educação Matemática**. Recife: Editora UFPE, 2013.

GRANDO, R. C. **O conhecimento matemático e o uso de jogos na sala de aula**. 2000. Tese (Doutorado em Educação) - Faculdade de Educação, Universidade Estadual de Campinas (UNICAMP), Campinas, 2000.

HEFEZ, A. **Curso de Álgebra**. v. 1. Rio de Janeiro: IMPA, 1993.

KAMII, Constance. **A criança e o número: Implicações educacionais da teoria de Piaget para a atuação junto a escolares de 4 a 6 anos**. 23. ed. Campinas (SP): Papirus, 1997. 124 p.

KRUCHTEN, P. **Introduction au Rational Unified Process**. Editions Eyrolles, 1999.

KRUCHTEN, P. **Introdução ao RUP - Rational Unified Process**. Rio de Janeiro: Ciência Moderna, 2003.

LAGRANGE, J.-B. Curriculum, classroom practices and tool design in the learning of functions through technology-aided experimental approaches. **International Journal of Computers in Mathematics Learning**, 10: p. 143-189, mai. 2005.

LOPES, V. N. Mankala: Jogo de tabuleiro de origem africana explora valores e habilidades. *In: Revista do Professor*, ano 24, n. 96, out./dez. 2008.

MAGELA, R. **Engenharia de Software Aplicada: Princípios**. 1. ed. Rio de Janeiro: Alta Books Ltda, 2006.

MELO, M. S. L.; MONTENEGRO, G. M. M.; SANTOS, L. S.; MORAIS, M. D.; BELLEMAIN, P. M. B. Bingo dos Números Racionais. *In: GITIRANA, V.; TELES, R. A. M.; BELLEMAIN, P. M. B.; CASTRO, A.T.; ALMEIDA, I.A.C.; LIMA, P. F.; BELLEMAIN, F. (Org.). Jogos com Sucata na Educação Matemática*, 1. ed. Recife: Editora UFPE, 2013. Cap 6, p. 103-147.

NUNES, T.; BRYANT, P. **Crianças fazendo matemática**. 1. ed. Porto alegre: Artmed, 1997. 244 p.

PAIVA, M. A. V.; PEREIRA, R. C. Análise da Transposição Didática do Conceito de Divisibilidade em uma Turma do 6º Ano do Ensino Fundamental. *In: I Simpósio Latino-Americano de Didática da Matemática*, 2016, Bonito. **Anais do I Simpósio Latino-Americano de Didática da Matemática**. Bonito: LADIMA, 2016.

PANIZZA, Mabel (org). **Ensinar Matemática na educação infantil e nas séries iniciais**. Porto Alegre: Artmed, 2006. 188 p.

PERNAMBUCO. Secretaria de Educação de Pernambuco. **Parâmetros para a Educação Básica do estado de Pernambuco: Parâmetros Curriculares de Matemática para o Ensino Fundamental e Médio**. Recife: SEDUC-PE, 2012.

PROJETO REDE. Jogos na Educação Matemática. **Apresentação**. 2011. Disponível em: <https://docplayer.com.br/12476944-Projeto-rede-formacao-docente-interdisciplinaridade-e-acao-docente.html>. Acesso em: 29 jul. 2023.

RAMOS, C. S. **Princípios da engenharia de Software educativo com base na engenharia didática: uma prototipação do Bingo dos Racionais**. 2014. 111 p. Dissertação (Mestrado em Educação Matemática e Tecnológica) - Universidade Federal de Pernambuco, Recife, 2014.

RÉZEAU, J. **Médiatisation et médiation pédagogique dans un environnement multimédia**. 2001. Tese (Doutorado em Educação) - Université Victor Segalen Bordeaux 2, Bordeaux, 2001. Disponível em: <https://theses.hal.science/edutice-00000222/document>. Acesso em 29 jul. 2023.

SANTOS, C. J. **Limites e potencialidades do uso dos mankalas na educação matemática e nas relações etnico-raciais no ambiente escolar**. Maringá, 2008a. Disponível em: [http://www.gestaoescolar.diaadia.pr.gov.br/arquivos/File/producoes\\_pde/artigo\\_celso\\_jose\\_santos.pdf](http://www.gestaoescolar.diaadia.pr.gov.br/arquivos/File/producoes_pde/artigo_celso_jose_santos.pdf). Acesso em: 29 jul. 2023.

SANTOS, C. J. **Jogos africanos e a educação matemática: semeando com a família mancala**. Secretaria de Estado da Educação - Universidade Estadual do Paraná. Maringá, 2008b. Disponível em: <http://www.diaadiaeducacao.pr.gov.br/portals/pde/arquivos/121-2.pdf>. Acesso em: 29 jul. 2023.

SANTOS, G. L. Alguns princípios para situações de Engenharia de Software Educativos. **Interação**, Goiás: v. 34, n. 1, 2009. Disponível em: <http://www.revistas.ufg.br/index.php/interacao/article/view/6540/4801>. Acesso em: 29 jul. 2023.

SANTOS, T. R. **Mankala Colhe Três: jogando e explorando conhecimentos matemáticos por meio de situações didáticas**. 2014. Dissertação (Mestrado em Educação Matemática e Tecnológica) - Universidade Federal de Pernambuco, Recife, 2014.

SANTOS, L. G.; CUNHA, H. A utilização do jogo mancala como ferramenta para o desenvolvimento de ensino, aprendizagem e de habilidades em matemática. **Anais do X-ENEM**. Salvador, 2010.

SILVA, C. T. J. **A Engenharia Didático-Informática na prototipação de um software para abordar o conceito de taxa de variação**. 2016. Dissertação (Mestrado em Educação Matemática e Tecnológica) – Universidade Federal de Pernambuco, Recife, 2016.

SILVA, A. D. **Prototipação, desenvolvimento e validação de um micromundo com suportes para o ensino de área e perímetro**. 2019. Tese (Doutorado em Educação Matemática e Tecnológica) - Universidade Federal de Pernambuco, Recife, 2019.

SIQUEIRA, J. E. M. **Articulando os registros de representação semiótica das curvas cônicas através da integração de recursos computacionais**. 2019. Tese (Doutorado em Educação Matemática e Tecnológica) - Universidade Federal de Pernambuco, Recife, 2019.

SMOLE, K. S.; DINIZ, M. I.; MILANI, E. **Cadernos do Mathema: Jogos de Matemática de 6º a 9º ano**. Porto Alegre: Artmed, 2007.

SOMARIVA, J. F. G.; ZWIEREWICZ, M.; CANDIOTTO, W. C.; & MENEGALI, B. T. **Uso de jogos e o ensino da cultura africana na metodologia dos projetos criativos ecoformadores**. Revista Electrónica de Investigación y Docencia (REID), n. Monográfico, p. 25-40, out. 2011.

SOMMERVILLE, I. **Engenharia de Software**. 9. ed. São Paulo: Pearson Education, 2011.

SOSIO, A. Traditional mancalas. **Galeria de andrea.sosio**. Flickr: andrea.sosio. Disponível em: <https://www.flickr.com/photos/andsos/galleries/72157626345718730/>. Acesso em: 29 mar. 2022.

TCHOUNIKINE, P. **Platon-1: quelques Dimensions pour l'analyse des travaux de recherche en conception d'EIAH**. Rapport de l'Action Spécifique. Fondements théorique set méthodologiques de la conception des EIAH. Département STIC du CNRS. 2004. 19 p.

TCHOUNIKINE, P.. **Précis de recherche de Ingénieriedes EIAH**. 2009. Disponível em: <http://hal.archives-ouvertes.fr/docs/00/41/36/94/PDF/PrecisV1.pdf>. Acesso em: 06 mar. 2022

TEODORO. V. D. **Modellus: Learning Physics with Mathematical Modelling**. 2002. p. 248. Tese (Doutorado em Ciências da Educação) - Faculdade de Ciências e Tecnologia, Universidade Nova de Lisboa, Lisboa, 2002.

TIBURCIO, R. S. **Processo de Desenvolvimento de Software Educativo: um estudo da prototipação de um software para o ensino de função**. 2016. Dissertação (Mestrado em Educação Matemática e Tecnológica) - Universidade Federal de Pernambuco, Recife, 2016.

TIBURCIO, R.; BELLEMAIN, F.; RODRIGUES, A. Concepção e evolução da engenharia didático-informática: uma proposta de metodologia para a produção de software educativo. **Revista de Educação Matemática e Tecnológica Iberoamericana**, v. 12, n. 4, p. 1-24, set. 2021.

TIBURCIO, R. S. **A Engenharia Didático-Informática: uma metodologia para a produção de software educativo**. 2020. Tese (Doutorado em Educação Matemática e Tecnológica) - Universidade Federal de Pernambuco, Recife, 2020.

VERGNAUD, G. **El niño, las matemáticas y la realidad: problemas de la enseñanza de las matemáticas en la escuela primaria**. 1. ed. México: Trillas, 1991.