



# JOGOS EDUCATIVOS NA ÁREA DE ENGENHARIA DE REQUISITOS

**Trabalho de Conclusão de Curso**

**Engenharia da Computação**

**Daniel Negreiros Araujo**

**Orientador: Prof<sup>a</sup>. Dr<sup>a</sup>. Maria Lencastre Pinheiro de Menezes e Cruz**

**Co-Orientador: Prof. Dr. João Henrique Correia Pimentel**



UNIVERSIDADE  
DE PERNAMBUCO

**Universidade de Pernambuco  
Escola Politécnica de Pernambuco  
Graduação em Engenharia de Computação**

**DANIEL NEGREIROS ARAUJO**

**JOGOS EDUCATIVOS NA ÁREA DE  
ENGENHARIA DE REQUISITOS**

Monografia apresentada como requisito parcial para obtenção do diploma de Bacharel em Engenharia de Computação pela Escola Politécnica de Pernambuco – Universidade de Pernambuco.

### MONOGRAFIA DE FINAL DE CURSO

#### Avaliação Final (para o presidente da banca)\*

No dia 7 de julho de 2016, às 14:00 horas, reuniu-se para deliberar a defesa da monografia de conclusão de curso do discente DANIEL NEGREIROS ARAUJO, coorientado pelo professor João Henrique Correia Pimentel, sob título Jogos educativos na área de engenharia de requisitos, a banca composta pelos professores:

**Fernanda Maria Ribeiro Alencar**  
**João Henrique Correia Pimentel**

Após a apresentação da monografia e discussão entre os membros da Banca, a mesma foi considerada:

Aprovada       Aprovada com Restrições\*       Reprovada

e foi-lhe atribuída nota: 9,5 (nove e meio)

\*(Obrigatório o preenchimento do campo abaixo com comentários para o autor)

O discente terá 15 dias para entrega da versão final da monografia a contar da data deste documento.

FERNANDA MARIA RIBEIRO ALENCAR

JOÃO HENRIQUE CORREIA PIMENTEL

\* Este documento deverá ser encadernado juntamente com a monografia em versão final.

*Aos meus pais Ivson e Sufia, e ao meu irmão Guilherme,  
minha eterna gratidão.*

# Agradecimentos

Agradeço, primeiramente, aos meus pais, por terem me ensinado valores importantes e a ser sempre honesto.

Ao meu irmão Guilherme por insistir em me tornar uma pessoa melhor, mesmo que do seu jeito peculiar.

Aos meus orientadores Maria Lencastre e João Pimentel, e a Mariana, que dedicaram tempo e esforço para me ajudar no decorrer da monografia, e que foram extremamente importantes para minha conclusão.

Aos ótimos professores de Graduação, como Carmelo, Sérgio Campello e Gustavo, que foram compreensivos em certas situações e ajudaram muito na minha formação.

Aos meus amigos de Universidade Toinho, Yuri, Pitágoras, John, Kinho, Diogo, Eric, Myllena, Renan, Chico e tantos outros que fiz durante o decorrer do curso, que apoiaram e foram excelentes companheiros.

Aos também amigos de Universidade Marcelo, Ingrid e Juliana, que me deram um apoio fundamental ao fazer a monografia.

# Resumo

Com o avanço da tecnologia, têm surgido novas propostas de configurações de ensino. Os *Softwares* Educacionais, por exemplo, têm sido bastante utilizados como forma de motivar e consolidar conhecimentos nas instituições de ensino. Porém, estes mesmos jogos têm sido desenvolvidos de uma forma pouco alinhada com os aspectos pedagógicos presentes na teoria do ensino. Na indústria de *software*, as deficiências na atividade da Engenharia de Requisitos (ER) são preocupantes, uma vez que são consideradas uma das principais causas de falhas de projeto de *software*; pesquisas apontam que 85% dos problemas de *software* têm origem na atividade de elicitação de requisitos. Há várias formas experimentais de ensino da ER; instruir usando jogos parece ser uma tendência inovadora, que promove a participação ativa dos alunos na dinâmica do ensino. Este trabalho apresenta um conjunto de jogos, disponíveis na literatura, e faz uma análise dos mesmos avaliando se estes contemplam requisitos considerados obrigatórios ou desejáveis em termos de *softwares* educacionais, de acordo com um catálogo de requisitos, chamado EduCatalog4RE, voltado para o auxílio ao desenvolvimento de *softwares* educacionais. Dos seis jogos analisados, três mostraram serem bons complementos para o ensino da ER. Porém, de um modo geral, os resultados demonstraram que os jogos voltados para o ensino podem ser ainda melhor desenvolvidos, contemplando características de ensino e aprendizado já consolidadas pela pedagogia, assim como se tratarem a questão de acessibilidade aos deficientes visuais.

**Palavras-chave:** jogos educativos, engenharia de requisitos, educatalog4re

# Abstract

With technology improvement, new proposals for educational settings have appeared. The Educational Softwares, for example, have been widely used as a way to motivate and consolidate knowledge in educational institutions. However, these games have been developed in a way that have a low alignment with pedagogical aspects present in educational theory. In the software industry, the deficiencies in the activity of Requirements Engineering (RE) are worrying, since they are considered a major cause of software project failures; researches show that 85% of software problems are in the elicitation activity requirements. There are several experimental forms of teaching the ER; to teach using games seems to be an innovative trend, which promotes the active participation of students in the dynamics of teaching. This monography presents a set of games available in the literature and an analysis of the same evaluating whether these include requirements regarded as required or desirable in terms of educational software, according to a catalog of requirements called EduCatalog4RE, turned to assist a educational software development. Of the six games analyzed, three had to be good additions to RE teaching. However, in general, the results demonstrated that games that focus on teaching can be developed even better, considering teaching and learning features already consolidated by pedagogy, so as to address the issue of accessibility to the visually impaired.

**Keywords:** educational games, requirements engineering, educatalog4re

# Sumário

<b>Capítulo 1 Introdução</b>	<b>1</b>
1.1 Motivação e caracterização do problema	1
1.2 Objetivos e Metas	1
1.3 Metodologia e estratégias de ação	2
1.3.1 Revisão da literatura	2
1.3.2 Análise de jogos	3
1.4 Estrutura da Monografia	3
<b>Capítulo 2 Fundamentação Teórica</b>	<b>5</b>
2.1 Engenharia de Requisitos	5
2.2 Tipos de Requisitos	7
2.3 Jogos Educacionais	9
2.4 EduCatalog4RE	10
<b>Capítulo 3 Avaliação dos Jogos</b>	<b>12</b>
3.1 Ilha dos Requisitos	13
3.2 EAReq-Game	22
3.3 <i>Software Quantum</i>	25
3.4 UbiRE	28
3.5 iStar Game	30

3.6	RE-O-Poly	33
3.7	Síntese dos Dados Coletados	36
<b>Capítulo 4 Análise de Requisitos nos Jogos</b>		<b>38</b>
4.1	Análise das Tabelas de Requisitos	43
<b>Capítulo 5 Considerações Finais</b>		<b>46</b>
5.1	Conclusão	46
5.2	Trabalhos Futuros	48
<b>Bibliografia</b>		<b>49</b>
<b>Anexo A Requisitos não Utilizados</b>		<b>52</b>

# Índice de Figuras

<b>Figura 1.</b>	Processo de Engenharia de Requisitos.....	6
<b>Figura 2.</b>	Tipos de Requisitos Não Funcionais. ....	8
<b>Figura 3.</b>	Exemplo de requisito abordado no catálogo. ....	11
<b>Figura 4.</b>	Tela Inicial do jogo Ilha dos Requisitos. ....	15
<b>Figura 5.</b>	Mapa inicial do Ilha de Requisitos. ....	15
<b>Figura 6.</b>	Desafio um do Ilha de Requisitos. ....	16
<b>Figura 7.</b>	Desafio dois do Ilha de Requisitos. ....	17
<b>Figura 8.</b>	Desafio três do Ilha de Requisitos. ....	17
<b>Figura 9.</b>	Desafio quatro do Ilha de Requisitos. ....	18
<b>Figura 10.</b>	Desafio cinco do Ilha de Requisitos (parte 1). ....	19
<b>Figura 11.</b>	Desafio cinco do Ilha de Requisitos (parte 2). ....	20
<b>Figura 12.</b>	Desafio seis do Ilha de Requisitos.....	21
<b>Figura 13.</b>	Último desafio do Ilha de Requisitos. ....	21
<b>Figura 14.</b>	Tela dos resultados do Ilha de Requisitos. ....	22
<b>Figura 15.</b>	Tela inicial do EAReq-Game. ....	23
<b>Figura 16.</b>	Explicação do EAReq-Game. ....	23
<b>Figura 17.</b>	Interação entre avatares para coleta de requisitos no EA-Req-Game. ....	24
<b>Figura 18.</b>	Gabarito das respostas do EA-Req-Game. ....	24
<b>Figura 19.</b>	Mensagem de parabenização no EA-Req-Game. ....	25

<b>Figura 20.</b>	Tela inicial do <i>Software Quantum</i> .....	26
<b>Figura 21.</b>	Nível um do <i>Software Quantum</i> .....	26
<b>Figura 22.</b>	Nível dois do <i>Software Quantum</i> , com o cliente falando diretamente com o Designer, e ele, falando diretamente com o programador.....	27
<b>Figura 23.</b>	Nível três do <i>Software Quantum</i> , com o cliente pulando todas as etapas e falando diretamente com o programador.....	27
<b>Figura 24.</b>	Tela dos resultados do <i>Quantum</i> , quando o jogador não consegue uma boa pontuação.....	28
<b>Figura 25.</b>	Tela inicial do UbiRE. ....	29
<b>Figura 26.</b>	Conexões possíveis entre objetos no UbiRE. Ao clicar em um dos objetos à direita, pode-se criar uma conexão com o selecionado previamente. .	29
<b>Figura 27.</b>	Simulação após acerto do desafio no UbiRE. ....	30
<b>Figura 28.</b>	Modelo do i* Game.....	31
<b>Figura 29.</b>	Exemplo de carta no i* Game.....	32
<b>Figura 30.</b>	Tabuleiro do RE-O-Poly .....	34
<b>Figura 31.</b>	Exemplo de cartas de cenário e de tarefa do RE-O-Poly.....	35

# Índice de Tabelas

<b>Tabela 1.</b>	Jogos vs. Artigos vs Acesso ao jogo. ....	12
<b>Tabela 2.</b>	Design instrucional vs. Desafio relacionado no jogo Ilha dos Requisitos	14
<b>Tabela 3.</b>	Síntese dos dados coletados .....	37
<b>Tabela 4.</b>	Requisitos obrigatórios x tipo do requisito. ....	38
<b>Tabela 5.</b>	Requisitos desejáveis x Tipo de Requisito. ....	40
<b>Tabela 6.</b>	Jogo vs. requisito obrigatório.....	41
<b>Tabela 7.</b>	Jogo vs. requisito desejável .....	42
<b>Tabela 8.</b>	Jogo vs. Quantidade de requisitos atendidos.....	45
<b>Tabela 9.</b>	Jogo vs. Quantidade de requisitos pedagógicos .....	45

# Tabela de Símbolos e Siglas

(Dispostos em ordem alfabética)

ER – Engenharia de Requisitos

E/S – entrada/saída

i\* – iStar

IEEE – Instituto de Engenheiros Eletricistas e Eletrônicos

NPC – *non-playable character* (personagem não-jogável)

PHP – PHP: *Hypertext Preprocessor*

SSP – *Stakeholder Satisfaction Points* (Pontos de satisfação do *stakeholder*)

# Capítulo 1

## Introdução

Neste capítulo estão descritos a motivação para a pesquisa, os objetivos deste trabalho, a metodologia empregada e a descrição da estrutura do trabalho.

### 1.1 Motivação e caracterização do problema

Deficiências na Engenharia de Requisitos (ER) são consideradas uma das causas comuns de falhas de projeto na indústria do *software* [9]. Pesquisas apontam que 85% dos problemas de *software* têm origem na atividade de elicitación de requisitos [8]. Analisando estas afirmações, observa-se que o método tradicional de ensino, por meio de aulas e exercícios talvez não esteja passando aos alunos o conteúdo de forma tão eficaz, seja pela desatenção dos mesmos na hora das aulas, seja pela falta de ensino diferente do tradicional.

Nos cursos relacionados a Computação, as aulas sobre Engenharia de Requisitos (ER) geralmente são na forma tradicional de aulas e exercícios [1].

Há várias formas experimentais de ensino da ER; instruir usando jogos parece ser uma tendência inovadora, que promove a participação ativa dos alunos na dinâmica do ensino [10]. Porém, a maior parte dos jogos ainda não está preparada e organizada efetivamente para um uso efetivo em sala de aula.

Esta monografia tem como foco a busca de jogos educativos que possam ser bons complementos de ensino na área de ER.

### 1.2 Objetivos e Metas

Este trabalho tem como objetivo fazer uma avaliação dos jogos na área de ER em termos de ensino.

As metas foram:

1. Levantar os jogos existentes na área de ER.
2. Baixar e classificar os jogos como pertinentes ou não à área de ER.
3. Definir o critério de avaliação dos jogos em termos de ensino.
4. Avaliar os jogos de acordo com os critérios estabelecidos.

## 1.3 Metodologia e estratégias de ação

Inicialmente foi realizado um levantamento, através de uma revisão bibliográfica, dos principais trabalhos sobre Engenharia de Requisitos; em seguida foi realizada uma revisão dos trabalhos encontrados, excluindo protótipos.

### 1.3.1 Revisão da literatura

Essa fase consistiu no levantamento e estudo de trabalhos sobre jogos educativos, mais especificamente sobre os que focam em Engenharia de Requisitos

- Na pesquisa de artigos e trabalhos acadêmicos, foram feitas buscas no site do Instituto de Engenheiros Eletricistas e Eletrônicos (IEEE) e no Springer entre as datas 20/03/2016 e 02/06/2016. Para tal, foi utilizada a *String* de busca *((requirements OR software) AND engineering) OR ((engenharia AND (requisitos OR software))) AND (game OR games OR jogo) AND (teaching OR learning)*.
- Além da pesquisa, houve uma busca em referências bibliográficas de artigos encontrados para encontrar mais jogos.
- Foram baixados 11 jogos, e para alguns houve necessidade de entrar em contato diretamente com os autores/criadores a fim de se ter acesso. Dos 11 jogos apenas 6 foram selecionados para análise (considerando critério de pertinência ao domínio da ER, e se erao jogos que não eram apenas protótipos.

### 1.3.2 Análise de jogos

Inicialmente cada jogo foi executado detalhadamente, para se aprofundar o conhecimento no mesmo. Em seguida, foi utilizado o catálogo de requisitos EduCatalog4RE [11], que lista requisitos importantes que devem estar presentes em ferramentas educativas, para se fazer uma avaliação de cada jogo em termos de requisitos educacionais usados, de usabilidade e gerais. Apenas 48 requisitos dos 108 propostos no catálogo foram considerados, por estes serem os mais representativos.

## 1.4 Estrutura da Monografia

Este documento está dividido em 5 capítulos, cujos respectivos resumos estão apresentados abaixo:

- **Capítulo 1: Introdução**

Neste capítulo estão inseridos os textos introdutórios deste trabalho: a motivação e a caracterização do problema, os objetivos, a metodologia empregada e a descrição da organização do documento.

- **Capítulo 2: Fundamentação Teórica**

Neste capítulo é fornecido o embasamento teórico neste trabalho. São descritos os conceitos de Engenharia de Requisitos, as classificações dos requisitos, uma descrição sobre jogos educacionais e a descrição do catálogo EduCatalog 4RE utilizado neste trabalho.

- **Capítulo 3: Descrição dos Jogos**

Este capítulo apresenta um resumo e uma análise dos jogos avaliados.

- **Capítulo 4: Avaliação dos Jogos**

Neste capítulo, é apresentada a análise dos jogos em relação aos requisitos apresentados do EduCatalog4RE.

- **Capítulo 5: Considerações Finais**

Neste capítulo são apresentadas as conclusões do trabalho, suas contribuições e possíveis trabalhos futuros.

# Capítulo 2

## Fundamentação Teórica

Este capítulo tem como objetivo apresentar os principais conceitos dos temas relacionados a este trabalho, que são: Engenharia de Requisitos, Tipos de Requisitos, Jogos Educativos e o EduCatalog4RE.

### 2.1 Engenharia de Requisitos

A Engenharia de Requisitos é uma abordagem sistemática e disciplinada para a especificação e gerenciamento de requisitos, com os seguintes objetivos [18]:

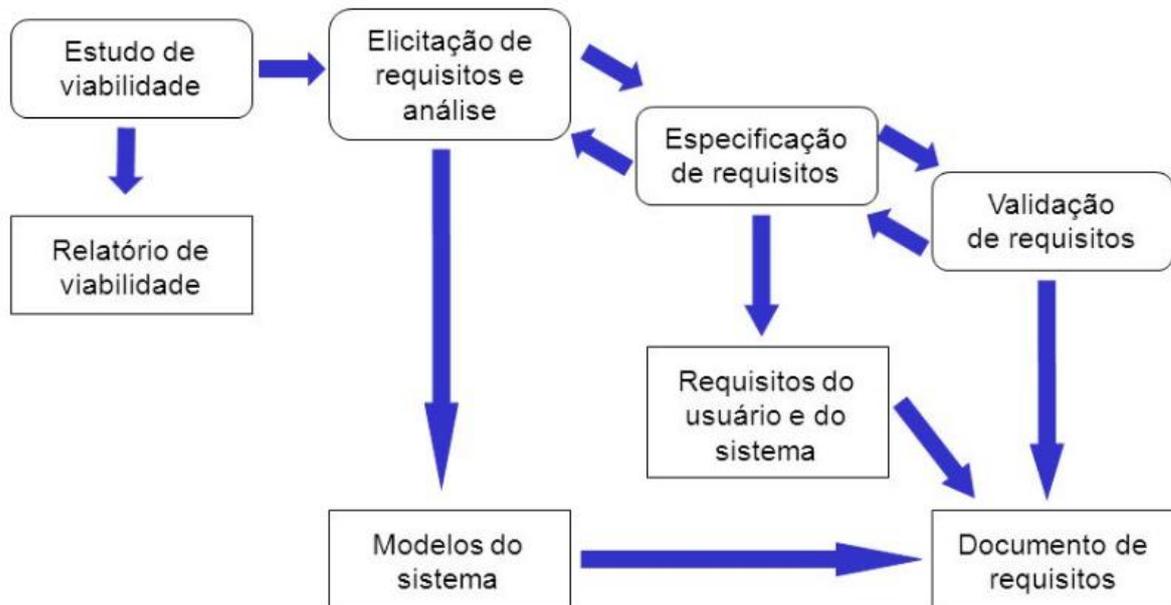
- 1- Conhecer os requisitos relevantes, estabelecer um consenso entre os *stakeholders* a respeito de tais requisitos, documentar os requisitos de acordo com determinados padrões e gerenciar os requisitos de forma sistemática.
- 2- Compreender e documentar as expectativas e necessidades dos *stakeholders*, especificar e gerenciar os requisitos para minimizar o risco de entregar um sistema que não atenda às suas expectativas e necessidades.

Segundo os autores Pohl e Rupp [18], as quatro atividades centrais para atingir esses objetivos são: elicitacão, documentacão, validacão e negociaçao, e gerenciamento de requisitos.

Porém, de acordo com o autor Pressman [20], a Engenharia de Requisitos abrange sete tarefas: concepção, levantamento, elaboracão, negociaçao, especificacão, validacão e gestao. Segundo o autor, as etapas podem ocorrer em paralelo e se adaptam de acordo com a necessidade do projeto.

Para o autor Sommerville [24], o objetivo do processo de ER é criar e manter um documento de requisitos de sistema. O processo geral inclui quatro

subprocessos de alto nível de Engenharia de Requisitos, que são: estudo de viabilidade, elicitacão e análise, especificacão e validacão. A Figura 1, além de demonstrar o relacionamento entre essas atividades, apresenta os documentos produzidos em cada estágio na Engenharia de Requisitos.



**Figura 1.** Processo de Engenharia de Requisitos.

Fonte: Sommerville (2008), p. 96

Um estudo de viabilidade é um estudo breve e focado que procura responder a uma série de questões [24]:

1. O sistema contribui para os objetivos gerais da organizacão?
2. O sistema pode ser implementado com tecnologia atual e dentro das restriçoes definidas de custo e prazo?
3. O sistema pode ser integrado a outros sistemas já implantados?

A elicitacão e análise dos requisitos têm como atividades a obtençao de requisitos, a classificacão e organizacão de requisitos, a priorizacão e negociaçao de requisitos, e a documentacão de requisitos [24].

Na validação de requisitos devem ser realizadas verificações no documento de requisitos, bem como as verificações de validade, de consistência, de completeza e de realismo, e além disso ter facilidade para verificação. Após as verificações, devem ser realizadas revisões de requisitos, prototipação e geração de casos de teste [24].

Conforme Sommerville, “O gerenciamento de requisitos é um processo para compreender e controlar as mudanças dos requisitos de sistema, isto é, manter o acompanhamento dos requisitos individuais e manter as ligações entre os requisitos dependentes, de modo que seja possível avaliar o impacto das mudanças de requisitos” [24].

Fica notório que cada autor tem uma visão distinta em relação às etapas que constituem o processo de Engenharia de Requisitos, porém elas são bem parecidas.

Nesta monografia foi utilizada a classificação de Sommerville, que divide a ER em quatro: estudo de viabilidade, elicitação e análise de requisitos, especificação de requisitos e validação de requisitos.

## 2.2 Tipos de Requisitos

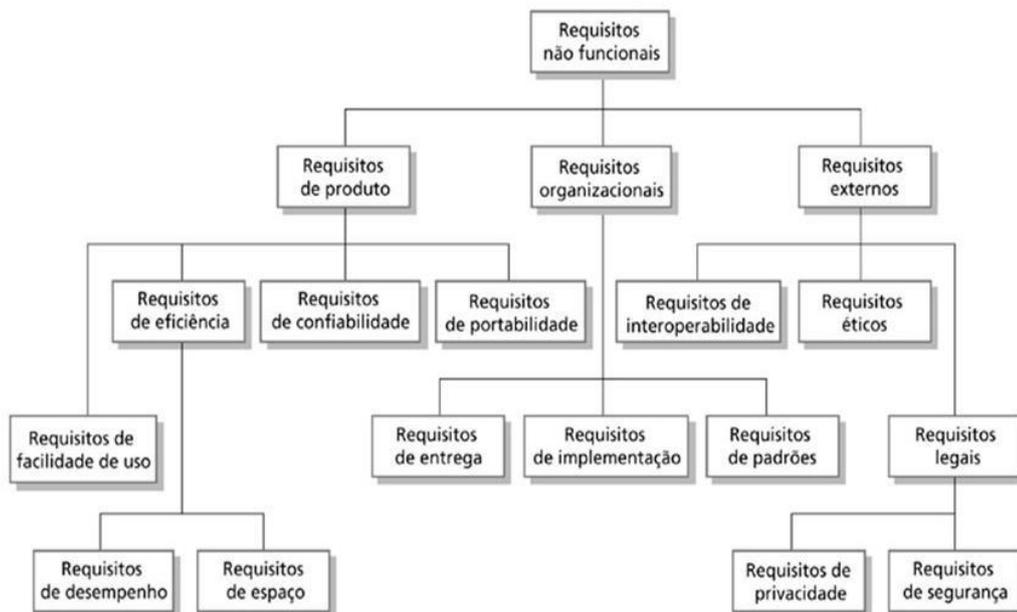
Os requisitos de sistema de software são classificados em: funcionais, não funcionais ou requisitos de domínio [24]. Os requisitos funcionais são descrições do que o sistema deve fazer, como o sistema deve reagir a entradas específicas e como o sistema deve se comportar em determinadas situações. Já os requisitos não funcionais são restrições sobre os serviços ou as funções oferecidos pelo sistema. Eles podem estar relacionados às propriedades emergentes do sistema, como confiabilidade, ou podem definir restrições, como a capacidade dos dispositivos de entrada e saída (E/S) do sistema. Os requisitos de domínio são provenientes do domínio da aplicação do sistema, e refletem as características e restrições deste domínio, podem ser tanto requisitos funcionais quanto não funcionais.

De acordo com o autor Sommerville [24], a distinção entre os diferentes tipos de requisitos não é tão clara. O autor exemplifica que um requisito de usuário que

esteja relacionado a segurança pode parecer um requisito não funcional, porém, ao desenvolvê-lo minuciosamente, esse requisito pode gerar outros requisitos funcionais, como a necessidade de incluir recursos de autenticação de usuário.

Idealmente, a especificação de requisitos funcionais de um sistema deve ser completa e consistente. Completa quer dizer que “de modo que todos os serviços exigidos pelo usuário devem ser definidos” e consistente quer dizer que “os requisitos não devem ter definições contraditórias” [24]. Há de se ressaltar que em sistemas mais complexos é praticamente impossível atender aos dois aspectos, visto que há uma maior quantidade de *stakeholders* que possuem necessidades distintas.

Os requisitos não funcionais surgem devido às necessidades do usuário, às restrições de orçamento, às políticas organizacionais, à necessidade de interoperabilidade com outros sistemas de software ou hardware ou a fatores externos, como regulamentos de segurança [24]. A Figura 2 apresenta uma classificação de requisitos não funcionais. Os requisitos não funcionais podem vir como requisitos de produto, requisitos organizacionais ou de fontes externas.



**Figura 2.** Tipos de Requisitos Não Funcionais.

Fonte: Sommerville (2008), p. 82.

Os requisitos de produto especificam o comportamento do produto. Como exemplos estão requisitos de desempenho, de confiabilidade, de portabilidade ou de usabilidade. Os requisitos organizacionais são derivados de políticas e procedimentos da organização do cliente e do desenvolvedor. Alguns exemplos são a linguagem de programação ou o método de projeto usado no desenvolvimento do sistema. Os requisitos externos podem incluir requisitos de interoperabilidade com outros fatores externos ao sistema, requisitos legais e éticos. [24]

A imprecisão na especificação de requisitos é o motivo de muitos problemas de engenharia de software. É natural que um desenvolvedor interprete um requisito ambíguo de forma a simplificar sua implementação. Porém, pode não ser exatamente o que o cliente quer. Novos requisitos precisam ser definidos e mudanças devem ser feitas no sistema. Então, isso atrasa a entrega do projeto e aumenta os custos.

## 2.3 Jogos Educacionais

O propósito dos jogos educacionais ou educativos é a interação do jogador com o conteúdo de uma forma dinâmica, envolvente, multimídia e interativa. Alguns jogos podem ir além, explorando estratégias de aprendizagem e resolução de problemas complexos, além de familiarizar o estudante com a tecnologia [19].

Para que um jogo educativo cumpra estes objetivos, alguns fatores não devem ser ignorados, um deles é o fator diversão. Um jogo educativo deve ser tão divertido quanto um jogo comercial, pois além de envolver o jogador, a diversão cria um estado propício à aprendizagem [19].

Segundo o autor Piaget [16], existem quatro categorias diferentes de jogos. São eles: jogos de exercícios, jogos simbólicos, jogos de acoplagem (ou construção), e jogos de regras. Jogos de exercícios são, como o próprio nome denota, jogos focados em atividades básicas, podendo ser jogos de exercício de pensamento como jogos sensório-motores, sendo que ambos estão relacionados ao prazer funcional ou à tomada de consciência de novos poderes. Os jogos simbólicos são voltados a crianças de dois a seis anos, por serem importantes para o

desenvolvimento cognitivo. Caracterizam-se por recriar a realidade usando sistemas simbólicos e estimulam a imaginação e a fantasia da criança, favorecendo a interpretação e ressignificação do mundo real. Jogos de acoplagem são aqueles em que a pessoa reproduz objetos ou situações que representam circunstâncias reais ou situações cotidianas. Já os jogos de regras são os que definem um escopo de regras, podendo ser regras simples ou complexas, para o jogador se basear.

A partir da análise dos jogos descritos nesta monografia, observou-se que eles contemplam três categorias: jogos de exercícios, de acoplagem e de regras.

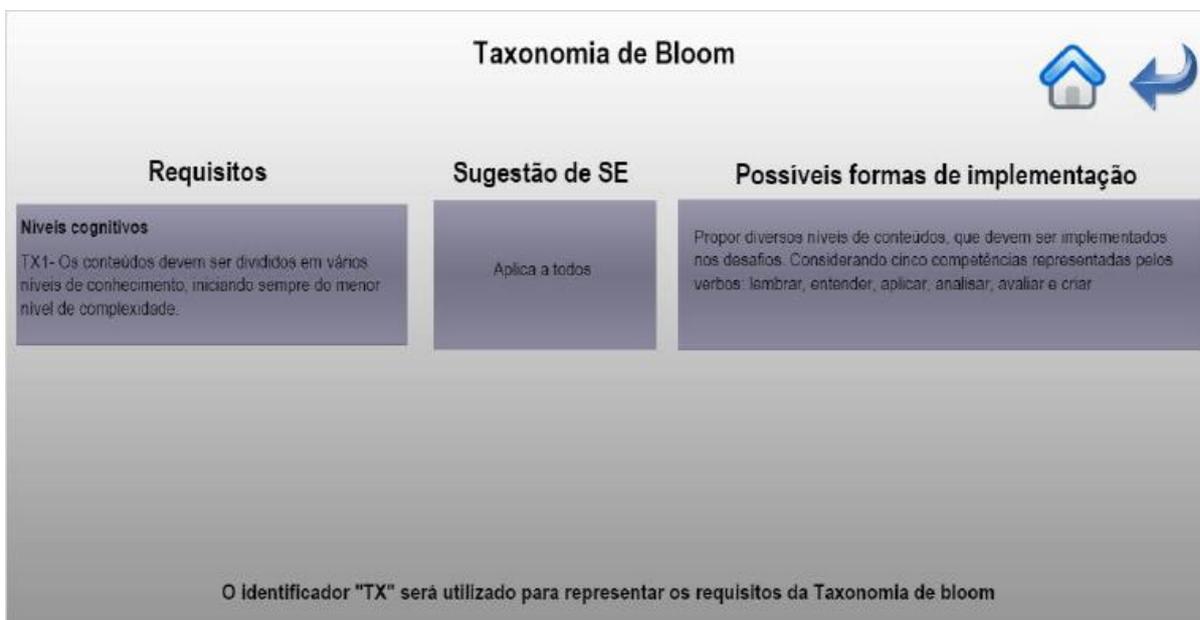
## 2.4 EduCatalog4RE

O catálogo de requisitos EduCatalog4RE [11] foi desenvolvido a partir de uma revisão sistemática da literatura, que buscou *softwares* educativos (Tutoriais, Exercício e prática, Aplicativos, Multimídia e Internet, Programação, Simulação e Modelagem, e Jogos). Este catálogo define uma lista de requisitos para *softwares* educacionais.

O EduCatalog4RE possui duas versões. A primeira versão foi realizada a partir de um questionário avaliativo com diversos docentes do país, assim como com representantes de empresas que produzem *softwares*, como o Manifesto Games. A segunda versão foi criada com base no *feedback* dos vinte e nove participantes. Houve uma redução de cento e doze requisitos na primeira versão, para cento e oito na segunda versão, além de melhorias em relação a usabilidade. Para a segunda versão foi criado um protótipo de uma ferramenta visando auxiliar na organização e busca de requisitos; nela o usuário pode selecionar sugestões de uso do catálogo, assim como ver os requisitos (que estão divididos em pedagógicos, de usabilidade e requisitos gerais); os requisitos pedagógicos são ainda divididos de acordo com as Teorias de Aprendizagem [11]. O usuário também pode optar por ver os tipos de *softwares* educativos abordados pela ferramenta.

O catálogo procura enumerar requisitos para *softwares* educativos. A Figura 3 mostra um exemplo de apresentação de requisitos na ferramenta, onde na parte superior da Figura é abordada a Teoria de Aprendizagem, os requisitos exclusivos

àquela Teoria, a sugestão de *softwares* educacionais que devem possuir o requisito, e dicas para possíveis formas de implementação deste requisito.



**Figura 3.** Exemplo de requisito abordado no catálogo.

Fonte: EDUCATALOG4RE

Esta monografia considera os requisitos do catálogo listados que são encontrados em todos os tipos de *software* educativo, além daqueles que são voltados para jogos. Vale lembrar que nem todos os requisitos presentes no catálogo são aqui avaliados, pois, alguns são estéticos ou opcionais, não adicionando muito ao objetivo da análise. Dos 108 requisitos da versão 2 do catálogo, foram selecionados 48, divididos em 30 requisitos obrigatórios e 18 requisitos desejáveis.

## Capítulo 3

# Descrição dos Jogos

Este capítulo resume alguns dos jogos disponíveis na literatura e faz uma síntese dos seus dados. Através da pesquisa realizada, foram encontrados cerca de vinte jogos potencialmente relacionados a Engenharia de Requisitos. A Tabela 1 apresenta os artigos correspondentes a cada jogo, e indica se o acesso ao jogo foi realizado com sucesso.

SE•RPG [1], The Incredible Manager [7] e Elicit@ção [3] não foram avaliados, ou por fugirem muito do escopo da monografia, ou pelo fato de não estarem disponíveis para o uso, como demonstra a Tabela 1, ou por serem protótipos.

**Tabela 1.** Jogos vs. Artigos vs Acesso ao jogo.

Nome do jogo	Artigo do jogo	Acesso?
Ilha dos Requisitos [27]	Promovendo a Aprendizagem de Engenharia de Requisitos de Software Através de um Jogo Educativo	Sim
EAREq-Game [5]	EAREq-Game: Um Jogo Educacional para o Ensino de Elicitação e Análise de Requisitos	Sim
Quantum [12]	<i>A Game for Taking Requirements Engineering More Seriously</i>	Sim
UbiRE [13]	<i>UbiRE: A Game for Teaching Requirements in the Context of Ubiquitous Systems</i>	Sim
iStar [17]	<i>i* Game</i>	Sim. Jogo de tabuleiro
RE-O-POLY [23]	<i>RE-O-Poly: A Customizable Game to Introduce and Reinforce Requirements Engineering Good Practices</i>	Sim. Jogo de tabuleiro
SE•RPG [1]	Utilização de um RPG no ensino de gerenciamento e processo de desenvolvimento de <i>software</i>	Sim
The Incredible Manager [7]	Treinamento Experimental com Jogos de Simulação para Gerentes de Projeto de <i>Software</i>	Sim
Elicit@ção [3]	Elicit@ção: Ferramenta de Apoio ao Ensino de Elicitação de Requisitos de <i>Software</i> baseada em Instituições Eletrônicas	Apenas protótipo.

SimuIES [9]	SimuES: Um Jogo para o Ensino de Engenharia de <i>Software</i>	Sim. Jogo de tabuleiro
ELEMENT [22]	Desenvolvimento e avaliação de jogo de treinamento para ensino de técnicas de elicitação de requisitos de <i>software</i>	Não
PMK [28]	<i>Project Management Knowledge Learning Environment: Ambiente Inteligente de Aprendizado para Educação em Gerenciamento de Projetos</i>	Não
SimSE [14]	<i>Teaching Software Engineering Using Simulation Games</i>	Não
Jogo sem nome [21]	<i>Simulating the Software Engineering Interview Process using a Decision-Based Serious Computer Game</i>	Não
Jogo sem nome [4]	<i>Diseño e Implementación de un Videojuego para la Enseñanza de Requisitos No Funcionales</i>	Não
SPIAL [15]	<i>Challenges and Issues in the Development of a Software Engineering Simulation Game</i>	Não
SECONDS e ref Quest [30]	<i>Collecting End User Requirements Playfully</i>	Não
Requirements Game [29]	<i>Requirements Game: Teaching Software Project Management</i>	Não

Nas próximas seções serão detalhados os jogos Ilha dos Requisitos, EAReq-Game, *Software Quantum*, UbiRE, i\* Game e o RE-O-Poly, uma vez que eles são os jogos com foco na área de ER que conseguiu-se o acesso.

### 3.1 Ilha dos Requisitos

Ilha dos Requisitos [27], “envolve a concepção da sua estória e de seus personagens, assim como a especificação de regras, desafios, e a definição das maneiras como o jogo irá fornecer o feedback ao jogador.” Uma primeira análise deste jogo demonstra uma interface simples, desenvolvida na plataforma Flash, para navegadores WEB. O jogo fornece sete desafios acessíveis sobre áreas relacionadas a ER, separadas por tópicos.

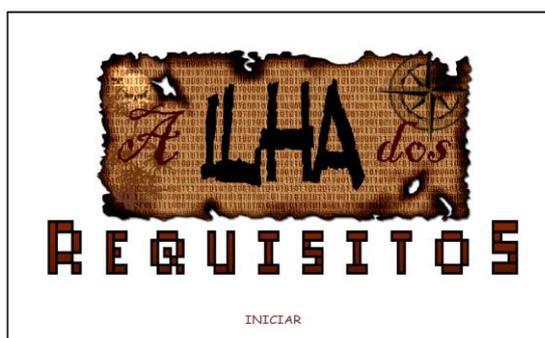
Os autores disponibilizam a Tabela 2, que faz uma associação dos tópicos que o jogo reforça o aprendizado e o desafio (1 a 7) que será responsável pelo ensino do conteúdo. [27]

**Tabela 2.** Design instrucional vs. Desafio relacionado no jogo Ilha dos Requisitos

<b>Tópico da Engenharia de Requisitos</b>	<b>Desafio relacionado</b>	<b>Breve descrição do desafio</b>
Processo de Engenharia de Requisitos	1	Ordenar as fases do processo de Engenharia de Requisitos.
Validação dos requisitos	5	Levar os requisitos ao “cliente” para que estes sejam validados antes do início da execução do projeto.
Papel do analista de requisitos	2	Identificar as habilidades do analista de requisitos.
Análise do problema	3	Distinguir o problema de possíveis soluções.
Especificação dos requisitos	4	Classificar requisitos apresentados como funcionais ou não funcionais.
Classificação dos requisitos	4	Classificar requisitos apresentados como funcionais ou não funcionais.
Gerência dos requisitos	6 e 7	Ordenar o processo de mudança dos requisitos; Identificar as atividades que compõe a gerencia de requisitos.

Fonte: Thiry et al. (2009).

Dadas estas informações, pode-se acessar o jogo a partir do link: <http://www.incremental.com.br/ilhadosrequisitos>. A plataforma utilizada para acesso é pelo *desktop*, com acesso a internet. A tela inicial é observada na Figura 4.



**Figura 4.** Tela Inicial do jogo Ilha dos Requisitos.

. Após selecionar INICIAR, há uma referência a história do jogo, que afirma que um avião com o avatar do jogador chamado Jack caiu em uma ilha, e logo em seguida abre uma tela, que mostra as regras e pede para informar o apelido do usuário. As regras determinam que o jogador tem vinte e dois “dias” restantes, onde um dia representa uma tentativa de resposta ou uma reinicialização do desafio realizada pelo jogador, e quarenta e cinco minutos, para escapar da ilha. Essas grandezas são diretamente proporcionais, isto é, a cada tentativa errada a pessoa terá menos tempo para terminar o jogo, e vice-versa. Então, o jogo abre a tela com o mapa principal (Figura 5).



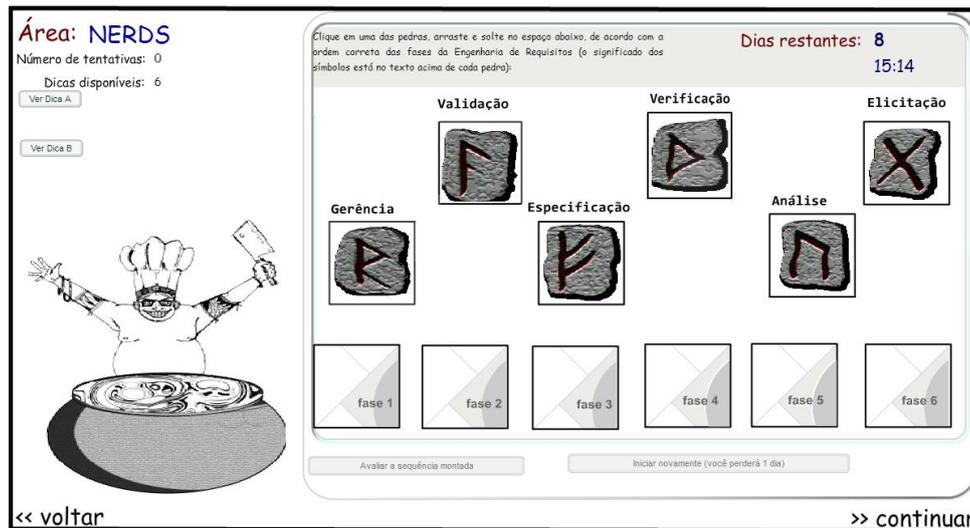
**Figura 5.** Mapa inicial do Ilha de Requisitos.

Se o jogador escolher ir ao avião, ele recebe uma bateria para o notebook dele, que serve para fornecer dicas (podendo ser utilizadas até seis vezes) durante os desafios. Porém, esta ação não é necessária para o decorrer do jogo, apenas o torna mais fácil. Os desafios começam quando se escolhe um local que corresponda a etapa do jogo. Há também a opção de visualizar o Diário de Jack, apenas para saber mais sobre a história do jogo. Após a escolha do lugar do desafio, o jogo exibe informações sobre o desafio subsequente, e quando o usuário opta por continuar, a tela com o desafio proposto é inicializada. Os locais que não correspondem ao desafio, ou:

- 1- Mostram um pequeno relato do local eleito, sem mostrar informações relevantes sobre o andamento do jogo.

2- Apresentam um pequeno texto sobre o assunto do último desafio realizado no lugar.

No primeiro desafio, após a escolha da opção Tribo dos Nerds, há a opção de utilizar até duas dicas, como demonstrado na Figura 6, e a proposta é ordenar as etapas da Engenharia de Requisitos. As etapas são seis, e a atividade consiste em arrastar o bloco que representa a etapa para a sua fase correspondente.



**Figura 6.** Desafio um do Ilha de Requisitos.

Caso o jogador escolha a sequência incorreta, aparece uma mensagem “Resposta incorreta”, e o usuário é obrigado a escolher a opção Iniciar novamente para continuar tentando. Caso acerte a sequência (Elicitação, Análise, Especificação, Verificação, Validação e Gerência), ele pode selecionar a opção “Continuar”, que fica disponível para escolha. A próxima tela explana um pouco sobre as etapas da ER, e o jogador pode optar por continuar o jogo. O jogador volta ao mapa, e o ciclo de escolha do local do mapa se repete. Com o jogador escolhendo o lugar Pedra Sagrada, o segundo desafio é iniciado.



Figura 7. Desafio dois do Ilha de Requisitos.

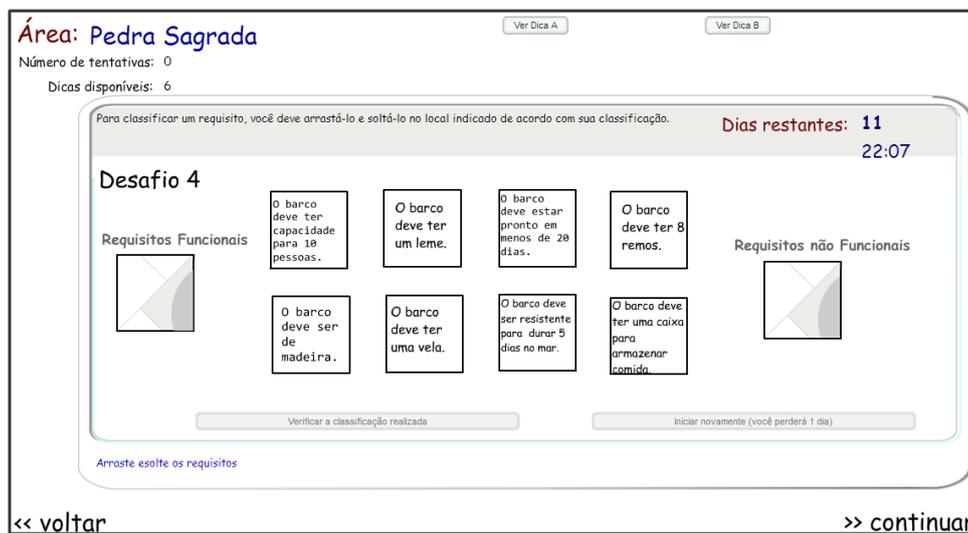
Como visto na Figura 7, o desafio dois consiste em marcar as habilidades associadas com um analista de requisitos. O jogador pode escolher receber até duas dicas, como o primeiro desafio. O jogador tem nove alternativas para escolha, podendo selecionar no mínimo uma, e no máximo as nove. Ao escolher as assertivas corretas (Ser bom ouvinte, ser bom observador, Analisar o problema do usuário e identificar suas necessidades e Mediar discussões e conflitos), o usuário pode avançar no jogo indo ao desafio três, antes passando pela tela que reafirma as importâncias de um analista de requisitos.



Figura 8. Desafio três do Ilha de Requisitos.

Após o jogador selecionar a área “Floresta”, o jogo começa o desafio três, que é uma atividade para esclarecer sobre quais são os problemas do personagem na história, como a Figura 8 demonstra. A forma do desafio é a mesma do desafio dois, onde a pessoa seleciona as alternativas, que são onze. Pergunta-se quais as frases que identificam o problema do jogador. O jogador também pode pedir até duas dicas.

Ao escolher as assertivas corretas (O vulcão irá explodir a ilha, A erupção ocorrerá em menos de 30 dias, e A tribo dos Nerds tem 15 membros, incluindo homens, mulheres e crianças), o jogo fala um pouco sobre o que é um problema na ER, e como fazer para discernir problema de solução. Ao continuar, o jogador pode voltar ao local Pedra Sagrada, para assim realizar o desafio quatro (Figura 9).



**Figura 9.** Desafio quatro do Ilha de Requisitos.

O desafio quatro seria equivalente no conceito ao desafio um, que é arrastar os blocos para os seus locais indicados, só que este desafio constitui-se em classificar os requisitos em requisitos funcionais e não funcionais. O jogador também pode pedir para visualizar, se preferir, duas dicas, caso não tenha usado todas anteriormente. Há oito requisitos envolvidos, dois quais quatro são funcionais, e quatro são não funcionais. Após a resposta exata (O barco ter um leme, O barco deve ter 8 remos, O barco deve ter uma vela, O barco deve ter uma caixa para armazenar comida em requisitos funcionais, e os requisitos restantes como não Funcionais), o jogador é agraciado com mais uma tela de explicação, dessa vez

perguntando ao usuário o que ele deve fazer agora. Após isso, o jogador pode selecionar no mapa para continuar, indo em direção a Tribo dos Nerds, para realizar o desafio cinco (Figura 10).



**Figura 10.** Desafio cinco do Ilha de Requisitos (parte 1).

O desafio cinco é composto de dois desafios pequenos; primeiro, o jogo questiona o que o jogador foi fazer na Tribo dos Nerds, com seis alternativas, podendo escolher no mínimo uma e no máximo as seis. Ao selecionar as alternativas corretas (Avaliar se existem novos requisitos e Validar os requisitos documentados na Pedra Sagrada com UbaapNerd), o jogador abre a segunda parte do desafio cinco (Figura 11), que pergunta o porquê de ser importante validar os requisitos com o UbaapNerd (fazendo referência à etapa de validar requisitos com o cliente).



**Figura 11.** Desafio cinco do Ilha de Requisitos (parte 2).

Como os desafios dois e três, o jogo oferece algumas assertivas para o jogador marcar. Desta vez, há cinco. Ao selecionar a resposta correta (Confirmar se o trabalho que será desenvolvido corresponde ao que o cliente deseja), o jogador conclui o desafio cinco, e o jogo relembra que no desafio quatro havia um requisito para construir um barco para até dez pessoas, e a tribo possui quinze pessoas, enfatizando a importância da área de validação.

Com a conclusão do desafio citado, o jogador agora é capaz de realizar o penúltimo desafio, localizado na Tribo Geek. Este desafio foca no fluxo de mudança de requisito, isto é, depois da validação, que se observa a necessidade de alteração de requisitos, propõe-se algumas mudanças em requisitos que não correspondem exatamente ao escopo do jogo. O jogo oferece ao jogador ver uma dica, caso o jogador ainda possua dicas disponíveis, e um desafio semelhante ao primeiro desafio, onde tem que arrastar os cinco blocos que representam as fases para sua sequência correta. (Figura 12)

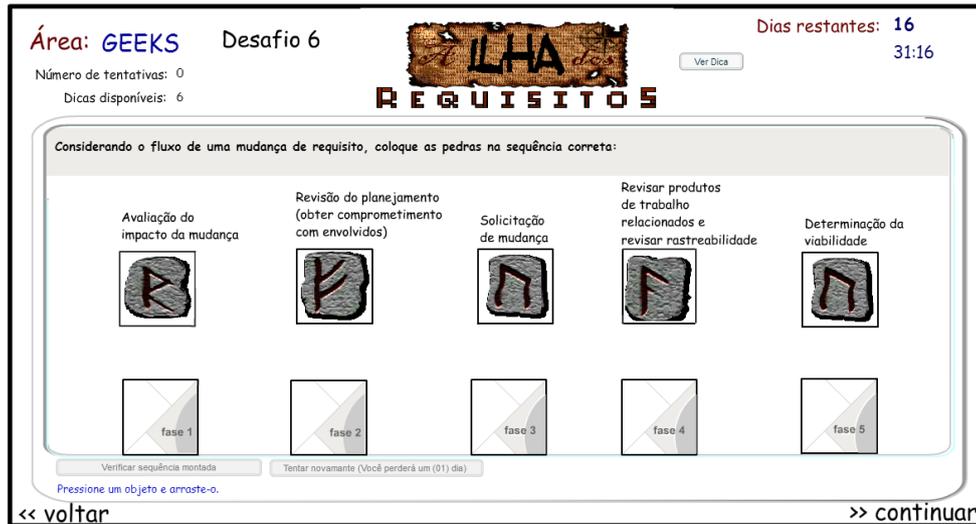


Figura 12. Desafio seis do Ilha de Requisitos.

Ao selecionar a sequência correta (Solicitação de mudança, Avaliação do impacto da mudança, Determinação da viabilidade, Revisão do planejamento e Revisar produtos de trabalho relacionados e revisar rastreabilidade), o jogador tem acesso a tela que explica um pouco mais sobre as etapas, e agora pode ir ao último desafio, que é realizado no único local que não teve desafio ainda (Praia). Na seleção, o jogo abre a tela representada na Figura 13 o último desafio.



Figura 13. Último desafio do Ilha de Requisitos.

O último desafio consiste em o que o jogador deve fazer para sair da ilha com todas as pessoas das duas tribos. Semelhante a parte 2 do quinto desafio, há cinco

assertivas para se escolher. Ao selecionar as corretas e apertar no botão para avaliação (Verificar quais os requisitos que foram afetados pela mudança, Incluir requisitos para acomodar as mudanças solicitadas, e Informar a UbaapNerd sobre o impacto de tempo e de custo para realizar a mudança solicitada), o jogo finaliza, mostrando a pontuação do jogador, a quantidade de dicas restantes, e uma tabela com os desafios, o tempo de entrada e de saída, e o número de tentativas (ver Figura 14 como exemplo).

Jogador:  
Dicas restantes: 5  
Pontuação: 2546

Desafio	Entrada	Saída	Tentativas
Desafio 1	44:57	44:49	1
Desafio 2	44:48	44:40	1
Desafio 3	44:37	44:12	1
Desafio 4	44:11	43:59	1
Desafio 5	43:58	31:34	2
Desafio 6	31:28	24:55	1
Desafio 7	24:14	19:57	3

**Figura 14.** Tela dos resultados do Ilha de Requisitos.

Nota-se que o jogo tenta trabalhar com conceitos mais específicos da Engenharia de Requisitos, como suas etapas e as funções de analistas, além de ter um bom foco em definir a diferença entre requisitos funcionais e não funcionais. O jogo se enquadra nas categorias de jogos de exercícios e jogos de regras.

## 3.2 EAReq-Game

O EAReq-Game [5] é um jogo que tem como objetivo simular uma situação ao jogador, “no qual atuará como um engenheiro de requisitos coletando, organizando e priorizando os requisitos em um cenário simulado, buscando progredir nas fases do jogo”. O jogo foi desenvolvido no *RPG Maker*. O jogo foi enviado para análise pelos desenvolvedores, então, não está disponível para *download*. A plataforma utilizada para acesso é o *desktop*.



**Figura 15.** Tela inicial do EReq-Game.

Ao abrir o executável, o usuário depara-se com a tela inicial (Figura 15), onde possui a opção para começar um jogo novo. Apertando a tecla 'espaço' no teclado, o jogo começa, oferecendo uma pequena história, para o jogador se situar. (Figura 16)



**Figura 16.** Explicação do EReq-Game.

Após a explicação, o jogador pode controlar o seu avatar com as setas do teclado. Após entrar no colégio, um *non-playable character* (NPC) interage com o jogador, falando qual caminho o usuário deve fazer para prosseguir. O jogador pode optar por rever o que o NPC falou, ao sair do colégio e entrar novamente. Ao seguir para a sala da secretaria, o avatar do jogador explica a um novo NPC que gostaria de coletar alguns requisitos. (Figura 17)



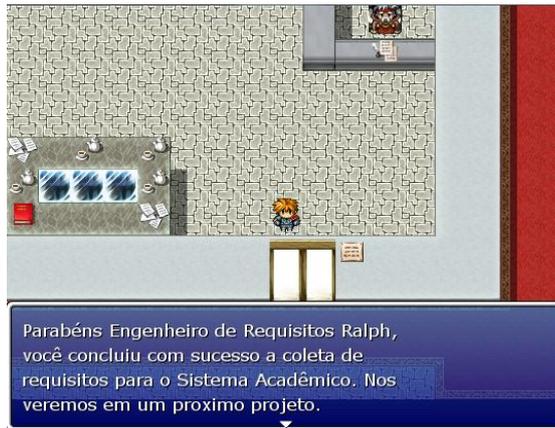
**Figura 17.** Interação entre avatares para coleta de requisitos no EA-Req-Game.

A partir daí, o jogo é bem simples: ao passar por alguns papéis dispostos pela sala, o jogador recolhe um requisito e o classifica. Alguns dos requisitos podem ser especificados como obrigatório ou desejável, enquanto outros podem ser rotulados como obrigatório, desejável ou fora do escopo. O jogador deve coletar todos os requisitos, que são oito, da secretaria para poder avançar para a próxima sala, a da direção. Após o término da primeira área, o jogo mostra as respostas corretas dos requisitos, e se o jogador acertou, como observado na Figura 18.



**Figura 18.** Gabarito das respostas do EA-Req-Game.

Ao se dirigir para a próxima área, o jogador recebe mais seis requisitos para análise. Finalizando todos os requisitos, se o jogador acertar no mínimo onze requisitos somando as duas salas, o jogo congratula o jogador (Figura 19); se não, o jogo dá uma mensagem de incentivo ao jogador para melhorar nesta área, e em ambos os casos, dá *Game Over*.



**Figura 19.** Mensagem de parabenização no EA-Req-Game.

O jogo apresenta um foco apenas em definir se os requisitos são obrigatórios ou desejáveis, não aprofundando em conhecimentos mais detalhados na Engenharia de Requisitos. O jogo se enquadra nas categorias de jogos de exercícios e de acoplagem.

### 3.3 Software Quantum

*Software Quantum* [12] é um jogo criado em Java, disponível no site [http://www.se.uni-hannover.de/pages/en:projekte\\_re\\_qgame](http://www.se.uni-hannover.de/pages/en:projekte_re_qgame). A plataforma utilizada para execução do jogo é o *desktop*, com acesso a Internet.

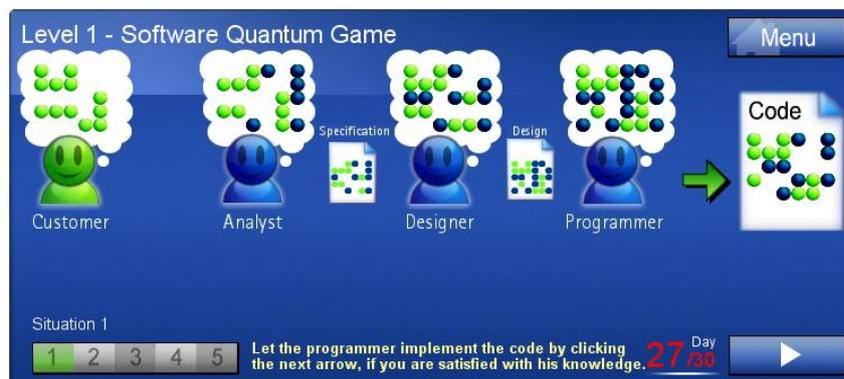
O jogo é baseado no conceito de *quantums* de *software*. O nome “*quantum* de *software*” alude a unidades atômicas. Uma unidade de informação de projeto, que podemos chamar de requisito, é chamado de *quantum* de *software*. O jogo visualiza requisitos como um saco de bolas. Requisitos do cliente são todos postulados a serem legítimos e válidos. Eles são demonstrados na cor verde no jogo. Quando o analista entrevista o cliente, ele “pega uma quantidade” destas bolas e as coloca no seu próprio “saco mental”. Nesse saco, as bolas que chegam são adicionadas às bolas presentes. As bolas podem ser pegadas durante cada passo da simulação. Porém, alguns requisitos falsos também são adicionados. Eles não representam os requisitos dos clientes, e sim, mal-entendidos ou suposições erradas (bolas azuis). Requisitos falsos e inválidos também são modelados usando este mecanismo. Quanto mais o analista interage com o cliente, mais bolas ele coleta.

A Figura 20 mostra a tela inicial do jogo.



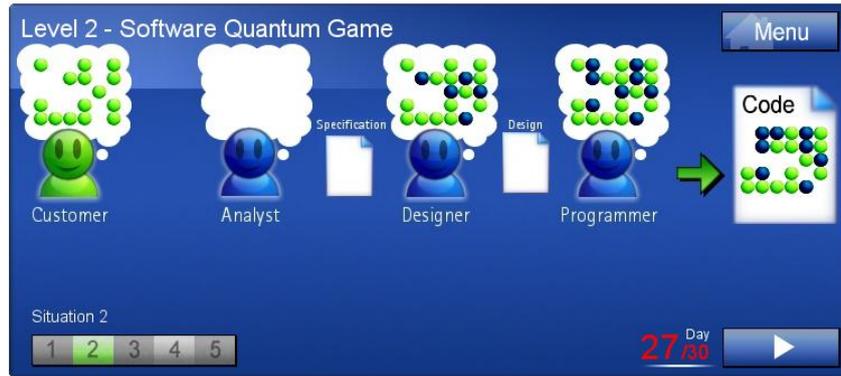
**Figura 20.** Tela inicial do *Software Quantum*.

O jogo possui três níveis. Cada nível possui cinco situações distintas. O primeiro nível familiariza os jogadores a operações básicas, como exemplos avançar o tempo de simulação e deixar os quantum de software fluírem da fonte para um alvo. Neste nível, quantum só podem fluir por um caminho pré-definido, seguindo um processo de cascata. A Figura 21 demonstra esse nível.



**Figura 21.** Nível um do *Software Quantum*.

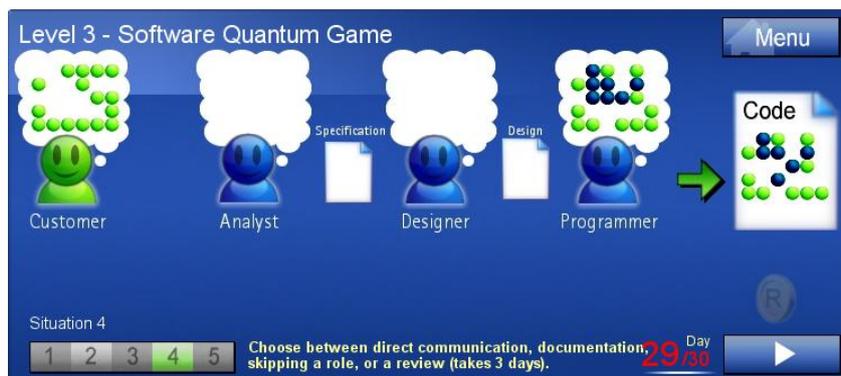
O segundo nível adiciona ao primeiro fluxos alternativos. Jogadores podem não apenas escolher quando ativar o próximo fluxo, como também podem decidir se este fluxo deve usar documentação (como no primeiro nível) ou comunicação oral entre as pessoas. No último caso, setas conectadas diretamente entre sacos mentais podem ser ativadas ao invés de fluxos de documentação. (Figura 22)



**Figura 22.** Nível dois do *Software Quantum*, com o cliente falando diretamente com o Designer, e ele, falando diretamente com o programador.

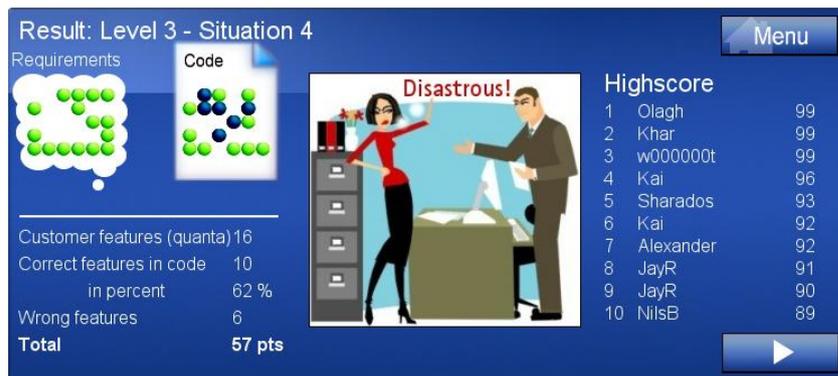
O terceiro nível adiciona mais variantes. Os clientes podem escolher falar com os programadores diretamente (Figura 23). Isso ajuda a evitar atrasos e passos intermediários. Mas ao mesmo tempo, toda especificação e documentos de design são pulados. Um projeto assim possui menos manutenibilidade, e depende muito mais de clientes individuais – e sofre quando eles mudam as suas ideias.

O terceiro nível também introduz a opção de revisões agendadas. Dois documentos são selecionados pelo jogador. Uma revisão compara os documentos ou sacos de quânticos e remove algumas das diferenças. Esta operação custa tempo de simulação. Quando é feita com frequência, não haverá tempo suficiente para fazer os requisitos fluírem. Sem revisões, porém, os documentos ficam poluídos com requisitos falsos.



**Figura 23.** Nível três do *Software Quantum*, com o cliente pulando todas as etapas e falando diretamente com o programador.

Ao fim de cada nível, há uma tela com a pontuação do jogador, e a pontuação dos usuários que utilizaram o jogo previamente. Caso o jogador consiga uma elevada pontuação de 80% do código correto, o jogo congratula o jogador. Caso contrário, o jogo fala que a performance do jogador foi desastrosa, como visto na Figura 24.



**Figura 24.** Tela dos resultados do *Quantum*, quando o jogador não consegue uma boa pontuação.

Observa-se que o foco do jogo é apenas enfatizar a importância da Engenharia de Requisitos, sem trazer ensinamentos contundentes sobre a área. O jogo se enquadra nas categorias de jogos de exercícios, e jogos de acoplagem.

### 3.4 UbiRE

UbiRE [13] é um jogo desenvolvido na ferramenta *GameMaker*. Ele é disponibilizado em <http://www.cos.ufrj.br/~thaiana/ubire>. Após baixar o jogo no *link*, é necessário renomeá-lo para a extensão *.exe*, para poder utilizá-lo em *desktop*. Ao abrir o arquivo, o jogador pode, além de escolher para jogar, na opção *Jogar*, selecionar o nível de dificuldade, entre fácil, médio e difícil, na opção *Nível*, visualizar os créditos do jogo, na opção *Créditos*, consultar as instruções, em *Instruções*, e sair do jogo, na opção *Sair*. Além disso, o usuário pode optar por remover a música no ícone de música (Figura 25).



**Figura 25.** Tela inicial do UbiRE.

Ao iniciar o jogo, o jogador depara com um personagem explicando mais um pouco do contexto do jogo, e após avançar, aparecem quatro cômodos da casa, e apenas um desbloqueado. As outras dependências são acessadas sucessivamente ao concluir os desafios. Ao acessar um quarto, o morador faz alguns comentários sobre seus requisitos no quarto. Como um dos exemplos, ele menciona que quer que a luz se acenda sozinha ao entrar no quarto, criando uma conexão entre a lâmpada, que é um objeto do quarto, com o sensor de movimento. Ao apertar sobre a lâmpada, o jogo abre uma tela com os outros objetos do cômodo, e assim o jogador consegue fazer as conexões entre os objetos (Figura 26).



**Figura 26.** Conexões possíveis entre objetos no UbiRE. Ao clicar em um dos objetos à direita, pode-se criar uma conexão com o selecionado previamente.

Então, cabe ao jogador realizar as conexões corretas, como pedido pelo morador da casa, para poder avançar no jogo. O usuário possui seis tentativas de erro para acertar cinco conexões que atendam aos requisitos. Ao término da atividade, o jogo faz uma simulação dos requisitos que foram pedidos, e assim, pode-se avançar ao próximo cômodo, que nada mais é que uma repetição do

primeiro desafio, com novos requisitos para serem validados, por meio de conexões entre objetos, com uma simulação ao fim (Figura 27), e assim sucessivamente.



**Figura 27.** Simulação após acerto do desafio no UbiRE.

Observou-se que a diferença entre níveis é que, enquanto que na categoria fácil o morador detalhou exatamente como queria as conexões, nos níveis mais difíceis, ele não especifica tão minuciosamente (como exemplo, o morador fala, no nível mais difícil e no primeiro cômodo, que “gostaria de assistir filmes como no cinema”), ficando como atividade extra decifrar o que o cliente realmente deseja para aquele cômodo, além do jogador possuir menos tentativas para errar.

Então, conclui-se que o jogo simula uma interação entre cliente e analista, onde o cliente pode tanto detalhar o máximo seu pedido, como pode dar apenas um desejo particular de como o sistema funcione, e a partir disso, o analista tem que desenvolver requisitos que atendam os desejos do cliente. Isto emula a situação cotidiana de engenheiros de requisitos, que precisam elicitare requisitos de acordo com as vontades daqueles que os contratam. O jogo se enquadra na categoria de exercício.

### 3.5 iStar Game

iStar Game (i\* Game) [17] é um jogo de tabuleiro não digital que, como os autores destacam, “consiste em se mover por um modelo de i\*, puxando cartas que contém ações de análise.” O jogo, isto é, os modelos utilizados, seu livro de regras,

e as cartas de análise, estão disponíveis diretamente no site: <https://sites.google.com/site/istargameufpe/home>.

A Figura 28 traz um dos três modelos disponíveis.

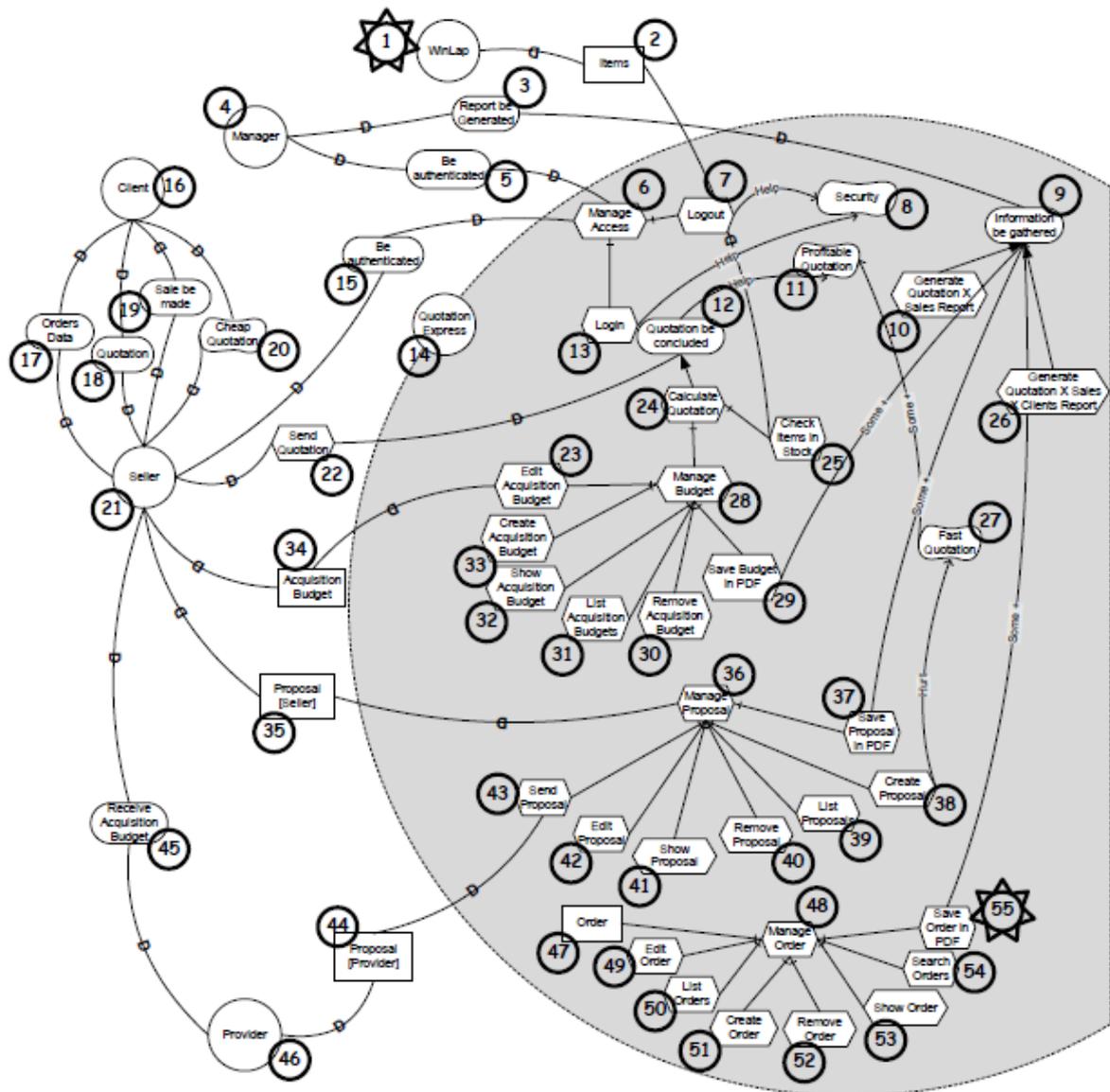


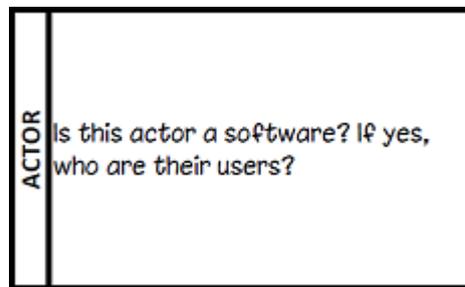
Figura 28. Modelo do i\* Game.

As regras do jogo são as seguintes:

Sobre o jogo: O modelo a ser analisado será utilizado como tabuleiro do jogo, no qual os jogadores irão avançando. Com os jogadores andando pelo tabuleiro, eles puxarão cartas contendo perguntas relacionadas a análise de modelo do i\*.

Destes questionamentos, os jogadores podem identificar melhorias a serem feitas no modelo, ou dúvidas a serem esclarecidas. Além do que é perguntado nas cartas, os jogadores estão livres para discutir qualquer problema que apareça durante a partida. Estes problemas e dúvidas devem ser gravadas, ou até escritas no modelo, em ordem a permitir a criação de uma versão revisada do mesmo.

*Setup* do jogo: Antes de jogar, é necessário criar o tabuleiro, do modelo  $i^*$  que será analisado. Para criar o tabuleiro, o necessário é somente numerar os elementos do modelo, em sequência. E, as cartas devem ser divididas em sete conjuntos: objetivos, sub-objetivos, tarefas, recursos, atores, dependências e desafios (Figura 29)



**Figura 29.** Exemplo de carta no  $i^*$  Game.

*Início do jogo:* Cada jogador escolhe um *token*, e o coloca no primeiro espaço. Todos jogam o dado uma vez, e começa quem tirar o maior número.

*Como jogar:* Jogar um dado de quatro ou de seis lados para saber quantos espaços o jogador irá se mover. Puxar uma carta de acordo com o tipo do elemento que a pessoa terminou após o movimento: objetivo, objetivo *soft*, tarefa, recurso ou ator. Se o jogador está em um *dependum*, ele irá puxar não apenas uma carta do tipo do elemento, mas também uma carta de dependência. Após isso, ele deve fazer a análise sugerida pela(s) carta(s). Então, é a vez do próximo jogador.

Se o jogador tirar o maior número do dado, antes de se mover ele deve retirar uma carta desafio. Após seguir as instruções da carta, a pessoa saberá quantos espaços que irá andar de fato. Após o término, o jogo segue normalmente.

O vencedor: O primeiro a chegar no último espaço do tabuleiro é o vencedor. Após isso, é a hora de discutir os resultados da análise de modo a identificar o que precisa ser alterado no modelo.

Constata-se que o jogo procura trabalhar os aspectos de análise e melhoria dos modelos feitos em i\*. Então, é necessário que o jogador já tenha familiaridade com os modelos. Ao ter tarefas, sub-tarefas e objetivos, o jogo trabalha alguns aspectos da Engenharia de Requisitos. O jogo pertence a categoria de acoplagem, e a categoria de regras.

### 3.6 RE-O-Poly

RE-O-Poly [23] é um jogo de tabuleiro não digital baseado no jogo *Monopoly* (conhecido como Banco Imobiliário no Brasil), e possui como intenção ensinar boas práticas de ER para engenheiros de requisitos principiantes. Os autores selecionaram, de uma lista de sessenta e seis boas práticas listadas no *Requirements Engineering Good Practice Guide* [25], oito para trabalharem em cima deles, que são: estrutura padrão, dividir em versões, possuir identificadores, prudência, fazer uso de *templates*, ser escrito em linguagem concisa, fazer inspeção de requisitos, e ter um *checklist* de validação e análise.

O jogo não encontra-se disponível para acesso e *download* do tabuleiro no site original <http://www.i-home.com.es/REgames/reopoly.html>, sendo necessário fontes alternativas para poder usá-lo. O tabuleiro é dividido em quatro quadrantes, cada um representando uma das fases do ER (Elicitação, Documentação, Validação e Gerenciamento). O tabuleiro é visto na Figura 30:

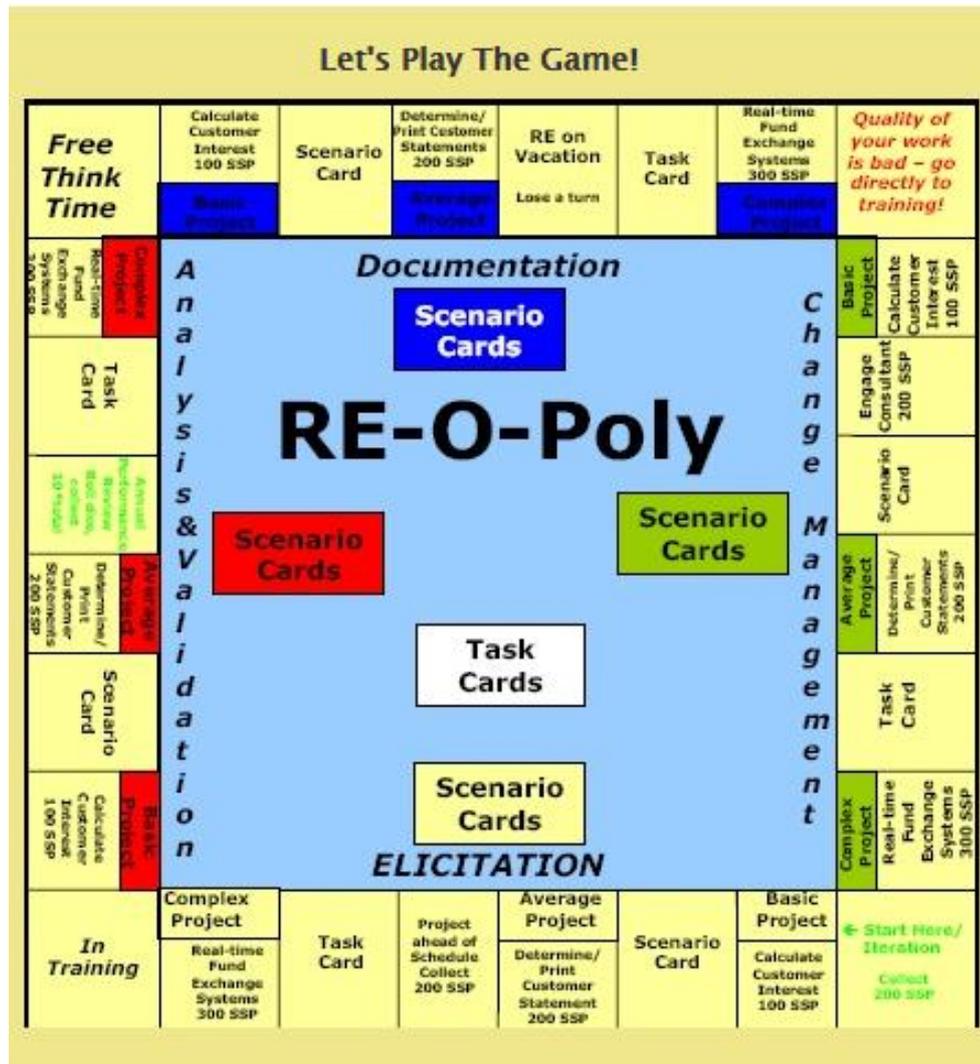


Figura 30. Tabuleiro do RE-O-Poly

RE-O-Poly pode ser jogado por até oito jogadores, e requer habilidades matemáticas, saber negociar e ser mente aberta. O jogo substitui o dinheiro utilizado no *Monopoly* original por *Stakeholder Satisfaction Points* (SSP), e os jogadores os coletam demonstrando conhecimentos sobre ER, gerenciando projetos no jogo. Além do tabuleiro, necessita-se de dois dados, *tokens* para representar os jogadores, e um cronômetro de trinta ou mais segundos. O jogo também possui cinquenta e duas cartas de cenário, trinta e duas cartas de tarefas e doze cartas de projeto, sendo quatro destas para cada projeto. Os projetos são divididos em básico, médio e complexo. A Figura 31 mostra algumas das cartas utilizadas no jogo.



**Figura 31.** Exemplo de cartas de cenário e de tarefa do RE-O-Poly.

As cartas de cenário possuem trabalho com apenas uma fase do processo de ER. Cada jogador começa com 1000 SSP, dadas pelo Facilitador. Todos os outros SSP e cartas de projeto são mantidas pelo Facilitador. O Facilitador é a pessoa que monitora os recursos do jogo, e geralmente não joga. Ele é o responsável por pagar e coletar SSP dos jogadores, vender e dar cartas de projeto, e nunca pode “falir”. Se o Facilitador estiver sem SSP disponível, ele pode criar mais.

Sobre o jogo: começa quem tirar o maior número nos dados. Então, o primeiro joga os dados e move a sua peça no sentido horário. Dependendo do espaço que a peça cair, o jogador pode comprar um projeto, ou ser obrigado a pagar SSP, pagar uma multa, puxar uma carta de tarefa ou de cenário, ser dirigido a um espaço particular do tabuleiro, e assim sucessivamente. Se tirar dois números iguais, o jogador lança os dados de novo, recebendo os benefícios ou prejuízos da casa que ficou. Se lançar três vezes consecutivas, ele vai para o espaço “*In RE Training*” (em treinamento de ER), e só consegue sair ao: esperar três rodadas, pagar 50 SSP pelas próximas três jogadas, ter a carta que libere (Get Out of Training Free Card), tirar números iguais no dado. Ainda, cada vez que o *token* passar pelo “Comece aqui”, seja jogando os dados ou puxando uma carta, o Facilitador paga 200 SSP.

Comprando projetos: O jogador pode comprar um projeto pelo preço indicado, desde que não pertença a alguém, ao cair sobre ele no tabuleiro. Caso não queira

comprar, o projeto entra em leilão, e todos, inclusive o que o recusou anteriormente, podem participar.

Desafios de projeto e condição de especialista: Se um jogador cair em um projeto que pertença a outro, e este não possua condição de especialista naquele projeto, o primeiro deve desafiar o segundo questionando uma das três perguntas listadas na carta de projeto. A premissa do desafio é que o dono do projeto deveria saber as respostas para as perguntas das suas cartas de projeto. Os gerentes do projeto devem também explicar o motivo por trás de suas respostas, e os outros jogadores decidem se suas respostas foram aceitáveis.

As três perguntas diferem em dificuldade, e se o jogador falhar em responder corretamente, ele precisa pagar ao desafiante o correspondente como descrito na carta, e vice-versa. Se o proprietário errar duas das três perguntas, a carta de projeto retorna ao Facilitador. Caso acerte todas as três perguntas, ele se torna especialista.

Fim do jogo: Os jogadores decidem o quanto eles irão jogar. No fim, aquele com mais SSP vence o jogo.

O que se percebe é que o jogo contempla boas práticas de ER para todas as atividades inclusas (isso é, elicitación, análise e validación, documentação e gerência). O jogo se enquadra nas categorias de acoplagem e de regras.

### **3.7 Síntese dos Dados Coletados**

Após uso dos jogos, foi possível coletar alguns dados, que são representados na Tabela 3. Todos os jogos que foram contemplados neste trabalho estão incluídos na tabela.

**Tabela 3.** Síntese dos dados coletados

Jogo	Estudos relacionados	Descrição do jogo	Quantidade de Jogadores	Ambiente de execução	Área de conhecimento	Atividade de aprendizagem
Ilha dos Requisitos	[Thiry, Zoucas e Gonçalves, 2011]	O estudante é apresentado a conceitos de Engenharia de Requisitos solucionando desafios, que representam várias etapas na ER.	Individual	Computador com conexão a Internet	Engenharia de Requisitos	O jogador se torna um sobrevivente de queda de avião em uma ilha, e deve resolver desafios referentes a Engenharia de Requisitos para conseguir escapar dela.
EAREq-Game	[Chiavegatti e Petri, 2014]	EAREq-Game traz atividades de priorização de requisitos, onde o jogador deve escolher se os requisitos são obrigatórios, desejáveis ou fora do escopo.	Individual	Computador em ambiente local	Priorização de requisitos	O jogador assume um papel de analista, e deve coletar requisitos e especificá-los de acordo com sua importância no projeto.
Software Quantum	[Knauss, Schneider e Stapel, 2008]	Jogo que simula uma interação entre cliente, analista, designer e programador. O cliente dá os requisitos e o código do programa no fim tem que ser equivalente aos requisitos pedidos.	Individual	Computador com conexão a Internet	Elicitação de requisitos	O jogador realiza interações entre os desejos do cliente com analista, designer e programador, estimulando simulações de comunicação oral entre eles, e documentação dos requisitos.
UbiRE	[Lima, Campos, Santos e Werner, 2012]	UbiRE simula uma casa inteligente, onde o jogador deve atender aos requisitos pedidos pelo morador da residência.	Individual	Computador em ambiente local	Análise de requisitos	O jogador realiza conexões entre determinados objetos em cômodos da casa, respeitando os pedidos feitos pelo morador, que podem ser ou não detalhados.
i* Game	[Pimentel, Santos e Pereira, 2012]	Jogo que utiliza modelos em i* como tabuleiro, cujo objetivo é melhorar estes modelos.	2-4 jogadores	Jogo de Tabuleiro	Análise de requisitos	Os jogadores realizam atividades de acordo com as cartas e regras fornecidas, de modo a melhorar os modelos disponibilizados de i*.
RE-O-Poly	[Smith e Gotel, 2008a] [Smith e Gotel, 2008b]	Jogo de tabuleiro baseado em <i>Monopoly</i> , trocando o dinheiro pelo SSP ( <i>Stakeholder Satisfaction Points</i> ) e adaptando o tabuleiro e as cartas para ensinar ER.	2-8 jogadores	Jogo de Tabuleiro	Boas práticas de Engenharia de Requisitos	Os jogadores são gerentes de projeto, e devem coletar SSP respondendo perguntas sobre boas práticas de ER e sobre os projetos que o jogo possui.

# Capítulo 4

## Avaliação dos Jogos

Este capítulo apresentará uma análise dos jogos considerando os requisitos do EduCatalog4RE [11]. Os requisitos foram divididos em obrigatórios e desejáveis, seguindo critérios de relevância, de conteúdo e técnicos. Alguns requisitos foram unificados ou removidos por serem ambíguos. O Anexo 'A' mostra os requisitos que não foram avaliados por esta monografia. São trinta requisitos obrigatórios avaliados e dezoito opcionais (Tabela 4 e Tabela 5, respectivamente). Destes trinta obrigatórios, dezesseis são pedagógicos, oito são de usabilidade e seis são requisitos gerais. Já dos dezoito opcionais, nove são pedagógicos, um de usabilidade, e oito requisitos gerais. Enfatiza-se que os requisitos são válidos para *Softwares*, em geral; logo, em algumas exceções eles não se aplicam aos jogos de tabuleiros analisados.

**Tabela 4.** Requisitos obrigatórios x tipo do requisito.

Requisito	Tipo de Requisito
R1- O sistema deve exibir feedback construtivo.	Pedagógico
R2- O sistema deve possuir vários desafios.	Pedagógico
R3- O sistema deve propor reflexões críticas ao trabalhar os conteúdos.	Pedagógico
R4- O sistema deve permitir que o usuário realize escolhas durante a interação com o ambiente.	Pedagógico
R5- O sistema deve propor ao usuário construir o máximo de soluções possíveis para cada problema proposto.	Pedagógico
R6- O conteúdo proposto deve se relacionar com os conhecimentos prévios do aluno.	Pedagógico
R7- É preciso criar um ambiente no qual os símbolos e signos do cotidiano dos alunos sejam transcritos visualmente.	Pedagógico
R8- Os objetos do ambiente devem mudar de estado quando o usuário interagir com eles, e devem ser manipuláveis, desta forma o usuário terá uma percepção maior durante a interação e terá uma maior interação no geral.	Pedagógico

R9- O sistema deve perceber as possíveis dificuldades do usuário e oferecer ajuda, através das interações do usuário com outros personagens.	Pedagógico
R10- Os personagens devem interagir entre si e com objetos distribuídos pelo ambiente.	Pedagógico
R11- O sistema deve permitir que o usuário possa interagir com o ambiente de várias formas.	Pedagógico
R12 -Os conteúdos devem ser planejados de forma hierárquica e sequencial.	Pedagógico
R13- O software deve penalizar o usuário, que serão consequências das suas ações.	Pedagógico
R14- O sistema deve prover feedback imediato para o usuário de acordo com as suas ações.	Pedagógico
R15- O sistema deve beneficiar o usuário caso ele acerte os desafios ou pegue itens importantes no cenário.	Pedagógico
R16- Os conteúdos devem ser divididos em vários níveis de conhecimento, iniciando sempre do menor nível de complexidade.	Pedagógico
R17- O sistema deve ser acessível ao público para o qual foi projetado.	Usabilidade
R18- O sistema deve disponibilizar ajuda através de menu, formulário, ou através de notificações para o usuário.	Usabilidade
R19- A interface do sistema deve ser fácil e bem intuitiva.	Usabilidade
R20- O sistema deve exibir instruções de uso sempre que solicitado pelo usuário.	Usabilidade
R21- O sistema deve disponibilizar um manual/tutorial que possua informações básicas sobre o sistema. Os usuários devem ter acesso ao material.	Usabilidade
R22- As instruções para uso do sistema devem estar visíveis e facilmente encontradas quando necessárias.	Usabilidade
R23- O sistema deve tornar a navegação fácil e eficiente.	Usabilidade
R24- O sistema deve exibir regras sempre que solicitado pelo usuário.	Usabilidade
R25- O sistema deve disponibilizar um avatar para facilitar a interação do usuário com o ambiente.	Requisito Geral
R26- O software deve possuir várias fases.	Requisito Geral
R27- O ambiente deve ser motivador e atraente, oferecendo ludicidade em sua interface e desafios.	Requisito Geral
R28- O sistema deve possuir vários níveis de dificuldade.	Requisito Geral
R29- O sistema deve exibir a pontuação dos usuários de forma individual e/ou coletiva.	Requisito Geral
R30- O sistema deve exibir o tempo dos desafios e das partidas, se houver.	Requisito Geral

Fonte: Henrique (2016)

**Tabela 5.** Requisitos desejáveis x tipo de requisito.

Requisito	Tipo de Requisito
D1- Disponibilizar o trabalho em rede (Internet)	Pedagógico
D2- O sistema deve permitir que o usuário crie desafios para outros jogadores.	Pedagógico
D3- O sistema deve prover feedback sonoro para auxílio aos usuários.	Pedagógico
D4- Organizar e disponibilizar os conteúdos (desafios e aulas) em ordem cronológica.	Pedagógico
D5- O sistema deve propor um ambiente que simule situações reais onde o problema será inserido.	Pedagógico
D6- O sistema deve ser capaz de detectar o perfil do usuário, isso deve acontecer antes dele começar a interagir com o sistema.	Pedagógico
D7- O sistema deve se adaptar as características do usuário.	Pedagógico
D8- O sistema deve levar em consideração a personalidade e/ou sentimentos do usuário, para propor estilos de aprendizagem.	Pedagógico
D9- Prover recursos que permitam que diferentes canais de processamento do usuário venham ser estimulados durante a aprendizagem.	Pedagógico
D10- O sistema deve apresentar de forma adequada os menus que proporcionarão a interação do usuário com o sistema.	Usabilidade
D11- O sistema deve disponibilizar um espaço para exibir os nomes dos desenvolvedores.	Requisito Geral
D12- O personagem (Player) e o ambiente devem possuir vários itens e características.	Requisito Geral
D13- O sistema deve possuir vários níveis de interação.	Requisito Geral
D14- O usuário poderá personalizar o avatar através da customização de novos elementos.	Requisito Geral
D15- O sistema deve calcular a pontuação dos usuários e gerar um ranking.	Requisito Geral
D16- O sistema deve gerar relatórios com dados estatísticos sobre o desempenho dos alunos.	Requisito Geral
D17- O sistema deve possuir uma versão “desktop ou off-line no caso do m-learning”, para os mais diversos sistemas operacionais (com ênfase no Windows)	Requisito Geral
D18- O sistema deve propor problemas menores com um curto prazo para resolução.	Requisito Geral

Fonte: Henrique (2016)

O requisito R8 era originalmente dividido em dois, que eram:

- 1- Os objetos do ambiente devem mudar de estado quando o usuário interagir com eles, desta forma o usuário terá uma percepção maior durante a interação.
- 2- Os objetos do ambiente devem ser manipuláveis.

Eles foram unificados por se assemelharem bastante em suas propostas.

A Tabela 6 mostra se o jogo atende ao requisito obrigatório apresentado. A Tabela 7, por sua vez, expõe se o jogo em questão atende ao requisito desejável apresentado.

**Tabela 6.** Jogo vs. requisito obrigatório

Jogo x Requisito	Ilha dos Requisitos	EAREq-Game	Software Quantum	UbiRE	I* Game	RE-O-Poly
R1	Sim	Não	Não	Sim	Sim	Sim
R2	Sim	Não	Sim	Sim	Sim	Sim
R3	Sim	Não	Não	Não	Sim	Sim
R4	Sim	Sim	Sim	Sim	Sim	Sim
R5	Não	Não	Sim	Não	Sim	Sim
R6	Sim	Sim	Não	Sim	Sim	Sim
R7	Sim	Sim	Sim	Sim	Sim	Sim
R8	Sim	Não	Sim	Sim	Sim	Sim
R9	Não	Não	Não	Não	N/A	N/A
R10	Não	Sim	Não	Não	Sim	Sim
R11	Não	Não	Sim	Sim	Sim	Sim

R12	Sim	Não	Não	Não	Não	Sim
R13	Sim	Não	Não	Sim	Sim	Sim
R14	Sim	Não	Sim	Sim	Sim	Sim
R15	Sim	Não	Não	Não	Sim	Sim
R16	Sim	Não	Não	Não	Sim	Sim
R17	Não	Não	Não	Não	Não	Não
R18	Sim	Não	Não	Sim	N/A	N/A
R19	Sim	Sim	Sim	Sim	Sim	Sim
R20	Sim	Não	Não	Não	N/A	N/A
R21	Sim	Não	Sim	Sim	Sim	Sim
R22	Sim	Não	Sim	Sim	N/A	N/A
R23	Sim	Sim	Sim	Sim	Sim	Sim
R24	Sim	Não	Não	Não	N/A	N/A
R25	Sim	Sim	Não	Sim	Sim	Sim
R26	Sim	Não	Não	Sim	Sim	Sim
R27	Sim	Sim	Sim	Sim	Sim	Sim
R28	Não	Não	Sim	Sim	Sim	Sim
R29	Sim	Sim	Sim	Não	Sim	Sim
R30	Sim	N/A	N/A	N/A	N/A	Sim

Legenda: N/A – Não se aplica

**Tabela 7.** Jogo vs. requisito desejável

Jogo x Requisito	Ilha dos Requisitos	EAREq-Game	Software Quantum	UbiRE	i* Game	RE-O-Poly
D1	Sim	Não	Sim	Sim	Não	Não
D2	Não	Não	Não	Não	Sim	Sim
D3	Não	Não	Não	Não	N/A	N/A

D4	Sim	Não	Não	Não	Sim	Sim
D5	Sim	Sim	Sim	Sim	Sim	Sim
D6	Não	Não	Não	Não	Não	Não
D7	Não	Não	Não	Não	Não	Não
D8	Não	Não	Não	Não	Não	Não
D9	Sim	Não	Não	Não	Sim	Sim
D10	Não	Sim	Sim	Sim	N/A	N/A
D11	Não	Não	Não	Sim	Não	Não
D12	Não	Sim	Não	Sim	Sim	Sim
D13	Sim	Não	Sim	Sim	Sim	Sim
D14	Não	Não	Não	Não	Não	Não
D15	Sim	Não	Sim	Não	Sim	Sim
D16	Não	Não	Não	Não	Não	Não
D17	Não	Sim	Não	Sim	N/A	N/A
D18	Não	Não	Não	Não	Não	Sim

Legenda: N/A – Não se aplica

## 4.1 Análise dos Resultados

De acordo com os resultados apresentados, observa-se que os jogos de tabuleiro RE-O-Poly e i\* Game, e o Ilha dos Requisitos são aqueles que mais atendem aos requisitos avaliados, superando ou igualando os 80% de requisitos obrigatórios sendo avaliados de forma positiva (eliminando os requisitos que não se aplicam). Por outro lado, o EAReq-Game é um jogo que possuiu uma avaliação muito negativa, correspondendo a menos de 33% dos requisitos obrigatórios atendidos. Os dois jogos restantes atendem na média entre 45% e 65% dos requisitos.

Em relação aos requisitos desejáveis, o RE-O-Poly e o i\* Game mais uma vez se sobressaem, porém diminuindo a porcentagem para cerca de 50% de requisitos atendidos.

Observando a relação dos requisitos pedagógicos, dos dezesseis obrigatórios, o Ilha dos Requisitos atende a 75%, o EAReq-Game a apenas 25%, o *Quantum* a aproximadamente 44%, o UbiRE a aproximadamente 56%, o i\* Game a 93% (sendo um dos dezesseis requisitos não aplicável), e o RE-O-Poly a 100% (sendo um dos dezesseis não aplicável).

Nota-se com o resultado alguns problemas, principalmente em relação à área de acessibilidade, onde nenhum jogo foi capaz de apresentar propostas para torná-los mais acessíveis aos usuários com deficiências visuais. Outros problemas frequentes foram que os jogos não percebem as dificuldades dos jogadores e não oferecem ajuda decorrente destas; apenas um, a Ilha dos Requisitos, exibe instruções de uso sempre que solicitado.

Nenhum jogo atende ao requisito desejável de ser adaptável ao jogador ou aos sentimentos dele; todos são feitos de maneira a seguir apenas uma forma única de jogo. Nenhum jogo provê *feedback* sonoro para auxílio aos usuários.

De acordo com os resultados, conclui-se que, apesar de não respeitarem todos os requisitos, os jogos Ilha dos Requisitos, i\* Game e RE-O-Poly podem se tornar prováveis bons complementos de ensino. Cada um traz uma diferente área abordada, o que possivelmente contribui para um maior conhecimento daqueles que utilizam esses jogos. A Tabela 8 mostra a porcentagem que cada jogo atendeu em relação aos requisitos, enquanto a Tabela 9 divide esta porcentagem em relação aos tipos dos requisitos (pedagógicos, de usabilidade e gerais).

**Tabela 8.** Jogo vs. Quantidade de requisitos atendidos

Jogo x Requisito	Requisitos Obrigatórios				Requisitos Desejáveis			
	Atendidos	Não Atendidos	Não se Aplicam	% atendida (aprox.)	Atendidos	Não Atendidos	Não se Aplicam	% atendida (aprox.)
Ilha dos Requisitos	24	6	0	80%	6	12	0	33%
EAREq- Game	9	20	1	31%	4	14	0	22%
Software Quantum	14	15	1	48%	5	13	0	28%
UbiRE	18	11	1	62%	7	11	0	39%
i* Game	22	2	6	92%	7	8	3	47%
RE-O-Poly	24	1	5	96%	8	7	3	53%

**Tabela 9.** Jogo vs. Quantidade de requisitos

Jogo x Requisito	Requisitos Obrigatórios			Requisitos Desejáveis		
	% Pedagógicos (aprox.)	% de Usabilidade (aprox.)	% Gerais (aprox.)	% Pedagógicos (aprox.)	% Usabilidade	% Gerais (aprox.)
Ilha dos Requisitos	75%	88%	83%	44%	0%	25%
EAREq- Game	25%	25%	60%*	11%	100%	25%
Software Quantum	44%	50%	60%*	22%	100%	25%
UbiRE	56%	63%	80%*	22%	100%	50%
i* Game	93%*	75%*	100%*	50%*	N/A	43%
RE-O-Poly	100%*	75%*	100%*	50%*	N/A	57%

Legenda: N/A - Não se aplica  
\* - Excluindo os requisitos que não se aplicam

# Capítulo 5

## Considerações Finais

Este capítulo tem como objetivo apresentar algumas considerações finais sobre os principais tópicos abordados nesta monografia e indicações para trabalhos futuros.

### 5.1 Conclusão

A Engenharia de Requisitos é uma fase muito importante na Engenharia de *Software* para o desenvolvimento de sistemas. É a fase em que a equipe de desenvolvimento entende as necessidades dos *stakeholders*. Toda a estrutura e as funcionalidades do sistema são determinadas nessa fase do projeto.

Elicitar requisitos é uma tarefa que exige um cuidado especial para que os requisitos sejam compreendidos corretamente e atendam às expectativas neles lançado, uma vez que é o momento no qual os *stakeholders* expõem suas necessidades e a visão do sistema a ser desenvolvido. Por abordar requisitos, necessidades e características do sistema que estão sujeitos a constante mudança, a Engenharia de Requisitos é uma fase do desenvolvimento de *software* onde erros e inadequações a esses requisitos são comuns, podendo não atender às reais necessidades da organização, resultando em retrabalho, custo e principalmente, na insatisfação do cliente.

Durante a elaboração desta monografia, verificou-se que a Engenharia de Requisitos, quando lecionada nas instituições de ensino, não varia da forma tradicional de aulas expositivas e exercícios [1]. Os jogos são capazes de prover conhecimento e diversão simultaneamente, e utilizá-los pode ser uma boa opção além das práticas adotadas atualmente [10].

Esta monografia propôs uma análise mais detalhada dos jogos educativos relacionados à Engenharia de Requisitos disponíveis na literatura, considerando

primeiramente a área abordada por eles. Utilizando o EduCatalog4RE [11], foi possível conferir se os jogos conseguem atender a requisitos pedagógicos.

Foram realizados um resumo e uma avaliação das áreas focadas pelos jogos, e após esta tarefa, se eles atenderam ao tema desta monografia, os mesmos foram analisados de acordo com o catálogo EduCatalog4RE [11], que lista vários requisitos essenciais em *softwares* educativos. Dos cento e oito requisitos que se encontram no catálogo (totalizando os requisitos que não devem ser contemplados em jogos), chegou-se a trinta requisitos considerados obrigatórios, e outros dezoito foram considerados desejáveis.

Após a análise geral dos resultados, observa-se que os jogos Ilha dos Requisitos, i\* Game e RE-O-Poly obtiveram resultados considerados bons, onde atenderam a pelo menos 80% dos requisitos obrigatórios. Outros dois jogos, sendo eles o UbiRE e o *Software Quantum* atenderam a entre 45% e 65% dos requisitos, tornando-os jogos razoáveis para uso. E o jogo EAREq-Game obteve um péssimo resultado, com cerca de 30% ou menos de requisitos avaliados positivamente. Porém, todos os jogos deixam a desejar na questão dos requisitos desejáveis, observando que nenhum deles chegou a uma taxa de 60% de requisitos atendidos.

Nenhum dos jogos avaliados possui acessibilidade ou é adaptável em relação ao conteúdo. Também, nenhum deles consegue perceber as dificuldades de jogador, e trabalha em cima delas.

Ao final desta monografia, percebeu-se que é viável a utilização de alguns destes jogos, como a Ilha dos Requisitos, i\* Game e RE-O-Poly, no ensino de ER. Eles apresentam um forma complementar ao ensino tradicional, podendo motivar o interesse dos alunos, e fortalecimento do seu conhecimento nas áreas de conceitos de ER (Ilha de Requisitos), melhoria de modelos de requisitos organizacionais (i\* Game) e conceitos relacionados a boas práticas de ER (RE-O-Poly).

Na área de definir a importância dos requisitos (se são obrigatórios ou desejáveis) há o jogo EAREq Game, porém o mesmo se mostrou um jogo insatisfatório em relação aos critérios adotados. O jogo UbiRE mostra uma simulação de situação onde o jogador deve atender aos requisitos do cliente, sendo

porém, em relação aos requisitos, um jogo intermediário. E o *Software Quantum*, também um jogo intermediário em relação aos requisitos, tenta fortalecer a importância da área de ER, demonstrando simulações onde o cliente conversa com o analista de projeto, mas sem explicar ou detalhar conceitos sobre ER.

## 5.2 Trabalhos Futuros

Como recomendações e sugestões de trabalhos futuros, para evoluir a presente pesquisa, pode-se destacar:

- Análise do uso de jogos em sala de aula, atualmente, na disciplina de ER;
- Explorar novos tipos de jogos, como os videogames.
- Desenvolvimento de jogos para cada uma das etapas da Engenharia de Requisitos de forma mais completa, abrangendo todo o conteúdo. Estes jogos podem combinar os pontos positivos das avaliações, e também atender tanto os requisitos obrigatórios quanto os desejáveis apresentados nesta monografia.

# Bibliografia

- [1] BENITTI, F.B.V., MOLLERI, J.S., Utilização de um RPG no Ensino de Gerenciamento e Processo de Desenvolvimento de Software, 2014
- [2] BERENBACH B. "A hole in the curriculum. In: Proceedings of 1st international workshop on requirements engineering education and training.", 2005.
- [3] BERNARDI, G., FONTOURA, L. M., CORDENONSI, A. Z., Elicit@ção: Ferramenta de Apoio ao Ensino de Elicitação de Requisitos de Software baseada em Instituições Eletrônicas. II Workshop-Escola de Sistemas de Agentes para Ambientes Colaborativos. Santa Cruz do Sul – RS, 2008
- [4] CALVO-MANZANO, J. A., ÁLVAREZ, M. C. G., Diseño e Implementación de un Videojuego para la Enseñanza de Requisitos No Funcionales, 8th Iberian Conference on Information Systems and Technologies (CISTI), 2013.
- [5] CHIAVEGATTI, N., PETRI, G., EAReq-Game: Um Jogo Educacional para o Ensino de Elicitação e Análise de Requisitos , Anais Do EATI - Encontro Anual de Tecnologia da Informação e Semana Acadêmica De Tecnologia Da Informação, Frederico Westphalen, Brasil, 2014.
- [6] CONNOLLY, T. M., STANSFIELD, M. HAINEY, T., Web Information Systems and Technologies, Third International Conference, WEBIST 2007, Barcelona, Spain, 2007.
- [7] DANTAS, A. R., BARROS, M. O., WERNER, C. M. L., Treinamento Experimental com Jogos de Simulação para Gerentes de Projeto de Software. Simpósio Brasileiro de Engenharia de Software (SBES), 2004.
- [8] FERNANDES, M., et al. "A Requirements Engineering and Management Training Course for Software Development Professionals." 22th Conference on Software Engineering Education and Training, 2009.
- [9] FIGUEIREDO, E., et al. "SimulES: Um Jogo para o Ensino de Engenharia de Software." Relat. Técnico 34.06, 2006.

- [10] GIL R., et al. "Experiential learning approach for requirements engineering education. " Springer Publishing Company, Incorporated, 2009.
- [11] HENRIQUE, M. S. "Educatalog4re: Um Catálogo de Requisitos para Auxiliar o Desenvolvimento de Softwares Educacionais." Dissertação de Mestrado em Ciência da Computação, área de concentração em Engenharia de Software, UFPE, Recife. 2016.
- [12] KNAUSS, E., SCHNEIDER, K., STAPEL, K., A Game for Taking Requirements Engineering More Seriously. Multimedia and Enjoyable Requirements Engineering (MERE), 2008.
- [13] LIMA, T., et al. "UbiRE: A game for teaching requirements in the context of ubiquitous systems." Informatica (CLEI), 2012 XXXVIII Conferencia Latinoamericana En. IEEE, 2012.
- [14] NAVARRO, E. O. et al. Teaching software engineering using simulation games, 2003.
- [15] PEIXOTO, D. C. C., POSSA, R. M., RESENDE, R. F., PÁDU, C. I. P.S., Challenges and Issues in the Development of a Software Engineering Simulation Game. Frontiers in Education Conference Proceedings, 2012.
- [16] PIAGET, Jean. A formação do símbolo na criança: imitação, jogo e sonho imagem e representação. 3ªed. Rio de Janeiro: Zahar, 1978.
- [17] PIMENTEL, J., SANTOS, E. , PEREIRA, T., I\*Game., 2012
- [18] POHL, K., RUPP, C. Fundamentos da Engenharia de Requisitos. 1ªed. 2012. 190p.
- [19] PRENSKY, M. Digital game-based learning. New York, McGraw-Hill, 2001.
- [20] PRESSMAN, R. S. Engenharia de software. McGraw Hill Brasil, 2011.
- [21] RUSU, A., RUSSEL, R. COCCO, R., Simulating the Software Engineering Interview Process using a Decision-Based Serious Computer Game, The 16th International Conference on Computer Games, 2011.

- [22] SILVA, M. S., BARBOSA, M. W., Desenvolvimento e avaliação de jogo de treinamento para ensino de técnicas de elicitação de requisitos de software, Belo Horizonte, MG, 2008.
- [23] SMITH, R., ORLENA G. "RE-O-Poly: A Customizable Game to Introduce and Reinforce Requirements Engineering Good Practices." Department of Computer Science, Pace University, New York, 2008.
- [24] SOMMERVILLE. I. Engenharia de Software. 9 ed. São Paulo: Pearson Addison-Wesley, 2007.
- [25] SOMMERVILLE, I. , SAWYER, P. Requirements Engineering: A Good Practice Guide. Chichester, England, John Wiley and Sons, 1997.
- [26] Standish Group. The CHAOS Report, 1994 and 2003 versions, <[www.standishgroup.com](http://www.standishgroup.com)>, acesso em: 21 de abril de 2016
- [27] THIRY, M., et al. "Promovendo a Aprendizagem de Engenharia de Requisitos de Software através de um Jogo Educativo." XXI Simpósio Brasileiro de Informática na Educação, 2010.
- [28] TORREAO, P.G.B.C., Project Management Knowledge Learning Environment: Ambiente Inteligente de Aprendizado para Educação em Gerenciamento de Projetos. Dissertação de Pós-Graduação em Ciência da Computação, área de concentração em Engenharia de Software, UFPE, Recife. 2005
- [29] ZAPATA, C. M. J. , AWAD-AUBAD, G., Requirements Game: Teaching Software Project Management, Clei Electronic Journal, volume 10, number 1, paper 3, 2007.
- [30] ZARVI, N., DUIN, H., SEIFERT, M., THOBENL, K., BIERWOL, R., Collecting End User Requirements Playfully. IEEE International Technology Management Conference (ICE), 2009.

# Anexo A

## Requisitos não Utilizados

Os seguintes requisitos também foram coletados do EduCatalog4RE, porém não foram avaliados nesta monografia por não se adequarem aos critérios selecionados e por não trazerem análises aprofundadas sobre os jogos.

1- Exibir feedback individualizado.
2- A funcionalidade "Abrir" deve permitir interação do usuário com o sistema.
3- A funcionalidade "Acessar" deve permitir interação do usuário com o sistema.
4- A funcionalidade "Alterar" deve permitir interação do usuário com o sistema.
5- A funcionalidade "Analisar/Avaliar" deve permitir interação do usuário com o sistema.
6- A funcionalidade "Adicionar" deve permitir interação do usuário com o sistema.
7- A funcionalidade "Atualizar" deve permitir interação do usuário com o sistema.
8- A funcionalidade "Cadastrar" deve permitir interação do usuário com o sistema.
9- A funcionalidade "Configurar" deve permitir interação do usuário com o sistema.
10- A funcionalidade "Consultar/Pesquisar" deve permitir interação do usuário com o sistema.
11- A funcionalidade "Enviar" deve permitir interação do usuário com o sistema.
12- A funcionalidade "Excluir" deve permitir interação do usuário com o sistema.
13- A funcionalidade "Listar" deve permitir interação do usuário com o sistema.
14- O sistema deve propor problemas menores com um curto prazo para resolução.
15- O sistema deve ser capaz de detectar o perfil do usuário, isso deve acontecer antes dele começar a interagir com o sistema.
16- O sistema deve se adaptar as características do usuário.
17- O sistema deve levar em consideração a personalidade e/ou sentimentos do usuário, para propor estilos de aprendizagem.
18- O sistema deve disponibilizar áudio (variados de acordo com a necessidade do sistema).

19- O sistema deve permitir a configuração do áudio por parte do usuário.
20- O sistema deve dispor de um banco de dados para armazenar informações relevantes para a avaliação dos alunos.
21- O usuário poderá personalizar o avatar através da customização de novos elementos.
22- O sistema deve disponibilizar vários tipos de recursos para o usuário.
23- O sistema deve registrar as ações do usuário durante a interação dele com o ambiente.
24- O sistema deve ser combinado com elementos da robótica.
25- Os usuários devem ser autenticados antes de usar o sistema.
26- O sistema deve suportar vários perfis, que variam de acordo com o acesso as informações.
27- As funcionalidades dos sistemas devem ser documentadas e outros artefatos devem ser produzidos.
28- O sistema deve disponibilizar todas as informações aos usuários, levando em consideração as permissões de acesso de cada usuário.
29- O sistema deve ser eficiente em seus recursos.
30- O sistema deve ser implementado para permitir a possibilidade de manutenção e extensão.
31- O sistema deve ter a capacidade de se comunicar de forma transparente com outras tecnologias e sistemas.
32- O sistema deve possuir componentes para reuso.
33- O sistema poderá acessar os dados do jogador.
34- Tolerância a falhas.