

UNIVERSIDADE FEDERAL DE SANTA CATARINA  
CURSO DE SISTEMAS DE INFORMAÇÃO  
DEPARTAMENTO DE INFORMÁTICA E ESTATÍSTICA

JOÃO PAULO GIRARDI PICCININI

DESENVOLVIMENTO DE UM REPOSITÓRIO DE JOGOS  
PARA O ENSINO DO SCRUM

FLORIANÓPOLIS-SC

2013/2

UNIVERSIDADE FEDERAL DE SANTA CATARINA  
CURSO DE SISTEMAS DE INFORMAÇÃO  
DEPARTAMENTO DE INFORMÁTICA E ESTATÍSTICA

JOÃO PAULO GIRARDI PICCININI

DESENVOLVIMENTO DE UM REPOSITÓRIO DE JOGOS  
PARA O ENSINO DO SCRUM

Trabalho de conclusão de curso desenvolvido  
como parte dos requisitos para obtenção do  
grau de Bacharel em Sistemas de Informação.

Orientadora:

Prof. Dr. rer. nat. Christiane A. Gresse von Wangenheim, PMP

Florianópolis / SC

2013/2

“Diga-me e eu esquecerei,  
Mostre-me e eu lembrarei,  
Envolve-me e eu entenderei.”

(CONFUCIUS, 450B.C. apud HSU; MALKIN, 2011, tradução nossa).

## **Dedicatória**

Dedico este trabalho à minha esposa, Rêneli,  
Por me apoiar nos momentos difíceis me motivando a seguir em frente.  
Aos meus pais que me deram grandiosos exemplos de vida e sabedoria.  
À Deus, por iluminar meu caminho.

## **Agradecimentos**

Agradeço à minha esposa por ter feito de tudo para que eu conseguisse chegar até aqui e por ter compreendido o motivo de minhas inúmeras ausências.

À minha família, principalmente, aos meus pais e minha irmã, que me apoiaram em todos os momentos de minha vida, para que eu continuasse em busca de meus sonhos.

Aos meus amigos, principalmente ao José Teixeira, Marcos Gomes, Ederson Melo, Fernanda Zerbin e Luccas Maso, por me darem grandes exemplos de foco para alcançar os objetivos, de comprometimento e de parceria, ao compartilharem desafios como este ou ainda maiores.

Aos professores, pelos ensinamentos e cobranças, que contribuíram para que eu me tornasse o aluno e profissional que sou hoje.

## Resumo

As empresas de desenvolvimento de software estão buscando recursos e ferramentas para desenvolver seus projetos da forma mais rápida e adaptativa possível para manter os negócios em patamares mínimos de rentabilidade. Uma das metodologias de gerenciamento de projetos que mais auxilia na gestão de mudanças e retorno rápido do investimento é o Scrum.

A metodologia Scrum possui conceitos básicos, fundamentados no gerenciamento ágil de projetos, que facilitam a sua disseminação na comunidade. O Scrum pode ser ensinado por meio de livros e artigos, ou por meio de jogos, que são métodos mais apropriados para o desenvolvimento de atitudes e habilidades, tão necessários para o profissional que deseja utilizar o Scrum.

Os jogos para o ensino do Scrum podem ser encontrados em *sites* e *blogs*, por meio de uma busca exploratória pela *Internet*. Os resultados de uma busca como esta podem ser diversos, desde a visualização de jogos que pouco tem em comum com os requisitos pesquisados inicialmente, até a obtenção de um jogo que, a priori, parece adequado, mas quando aplicado demonstra uma falha no ensino de acordo com os objetivos estabelecidos.

Considerando este cenário, propõe-se a construção de um repositório de jogos para o ensino do Scrum, que possibilite a busca por meio de filtros com as características do Scrum e o cadastro para compartilhamento destes jogos.

Para possibilitar a implementação deste repositório, primeiramente é realizada a fundamentação teórica nas áreas de gerenciamento ágil de projetos, aprendizagem e jogos educacionais, objetos de aprendizagem, metadados, repositórios de jogos e comunidades de prática. Após a fundamentação dos conceitos, é realizada a análise dos repositórios de jogos existentes e, a posteriori, é realizado o desenvolvimento do repositório.

Espera-se que o repositório desenvolvido neste projeto forneça meios para a busca e o compartilhamento de jogos para o ensino do Scrum, que estimulem a criação de jogos, a aplicação destas instruções em cursos e salas de aula e a criação de uma comunidade em volta do repositório.

**Palavras-chave:** Scrum. Jogos Educativos. Repositório de Objetos de Aprendizagem. Objetos de Aprendizagem.

# Lista de Figuras

Figura 1 - Papéis, marcos e ciclo de vida do Scrum. ....	25
Figura 2 - Etapas da fase de planejamento da <i>Sprint</i> . ....	33
Figura 3 - Quadro de tarefas associado as funcionalidade. ....	35
Figura 4 – Gráfico de burndown. ....	36
Figura 5 - Modelo de <i>design</i> instrucional.....	42
Figura 6 - Estratégias e métodos instrucionais . ....	49
Figura 7 – Níveis de participação e trajetórias em uma comunidade de prática.....	76
Figura 8 - Lista de jogos da categoria Agile, exibição de categorias e do campo de busca no repositório TastyCupCakes.....	81
Figura 9 – Exemplo de exibição de um jogo no repositório TastyCupCakes .....	82
Figura 10 - Parâmetros para a busca por jogos no Instructional Games Repository.....	83
Figura 11 - Exemplo de exibição de um jogo no IGR .....	84
Figura 12 - Lista de tópicos de discussão sobre jogos no grupo AgileGames.....	85
Figura 13 - Indicação de um jogo por um participante do grupo AgileGames .....	85
Figura 14 - Lista de jogos na página da <i>wiki</i> Games for Scrum.....	86
Figura 15 – Perfis e permissões de usuário no SGR. ....	108
Figura 16 - Atores e casos de uso do SGR .....	109
Figura 17 - Principais tabelas e relacionamentos na modelagem objeto-relacional do SGR. ....	119
Figura 18 - Tabelas e relacionamentos para descrever as características do jogo na modelagem objeto-relacional do SGR.....	120
Figura 19 - Tabelas e relacionamentos para classificação de jogos na modelagem objeto-relacional do SGR.....	121
Figura 20 – Tabelas e relacionamentos para descrever os recursos dos jogos na modelagem objeto-relacional do SGR.....	121
Figura 21 – Tabelas e relacionamentos para caracterização do usuário na modelagem objeto-relacional do SGR .....	122
Figura 22 - Grupos de elementos visuais no SGR .....	125
Figura 23 – Telas no caso de uso 1 - buscar jogo.....	126
Figura 24 – Telas no caso de uso 2 – visualizar informações do jogo .....	127
Figura 25 – Tela inicial no caso de uso 3 – cadastrar usuário.....	128
Figura 26 - Tela de cadastro no caso de uso 3 – cadastrar usuário .....	128
Figura 27 - Tela de autenticação no caso de uso 3 – cadastrar usuário .....	128
Figura 28 - Tela inicial no caso de uso 4 – visualizar termos de serviço .....	129
Figura 29 - Tela dos termos de serviço no caso de uso 4 – visualizar termos de serviço .....	129
Figura 30 - Tela inicial no caso de uso 5 – autenticar como usuário.....	130
Figura 31 - Tela de autenticação no caso de uso 5 – autenticar como usuário .....	130
Figura 32 - Tela inicial no caso de uso 5 – autenticar como usuário.....	131
Figura 33 - Tela inicial com usuário autenticado no caso de uso 6 – remover autenticação de usuário .....	131
Figura 34 - Tela inicial sem usuário autenticado no caso de uso 6 – remover autenticação de usuário .....	132
Figura 35 - Tela de avaliação e comentários no caso de uso 7 – avaliar e comentar jogo.....	133
Figura 36 – Tela com as avaliação e comentários do jogo no caso de uso 7 – avaliar e comentar jogo .....	133
Figura 37 - Tela inicial no caso de uso 8 – cadastrar jogo .....	134
Figura 38 - Tela de cadastro no caso de uso 8 – cadastrar jogo.....	134
Figura 39 – Tela com as informações do jogo no caso de uso 8 – cadastrar jogo .....	135
Figura 40 - Tela com a lista de jogos no caso de uso 9 – editar jogo cadastrado pelo próprio usuário. ....	136
Figura 41 - Tela para edição de informações no caso de uso 9 – editar jogo cadastrado pelo próprio usuário .....	136
Figura 42 - Tela com as informações editadas no caso de uso 9 – editar jogo cadastrado pelo próprio usuário .....	137
Figura 43 - Tela para acesso a lista de jogos no caso de uso 10 – visualizar jogos cadastrados pelo próprio usuário.....	137
Figura 44 - Tela para visualização da lista de jogos no caso de uso 10 – visualizar jogos cadastrados pelo próprio usuário .....	138

Figura 45 - Tela com a lista de jogos no caso de uso 11 – remover jogo cadastrado pelo próprio usuário.	138
Figura 46 - Tela com o jogo removido da lista no caso de uso 11 – remover jogo cadastrado pelo próprio usuário.	139
Figura 47 - Tela com para acesso o painel administrativo no caso de uso 12 – acessar o painel de administração.	139
Figura 48 - Tela com o painel administrativo no caso de uso 12 – acessar o painel de administração.	139
Figura 49 - Tela do caso de uso 13 – visualizar todos os jogos cadastrados.	140
Figura 50 - Tela para remoção de jogo no caso de uso 14 – remover qualquer jogo.	141
Figura 51 – Tela pós-remoção de jogo no caso de uso 14 – remover qualquer jogo.	141
Figura 52 - Tela seleção do jogo para edição no caso de uso 15 – editar qualquer jogo.	142
Figura 53 - Telas de edição e visualização no caso de uso 15 – editar qualquer jogo.	142
Figura 54 - Telas de edição e visualização no caso de uso 16 – visualizar usuários cadastrados.	143
Figura 55 - Telas de seleção de usuário para remoção no caso de uso 17 – remover usuário.	143
Figura 56 - Tela de exibição dos usuários restantes no caso de uso 17 – remover usuário.	144
Figura 57 – Tela para escolha de usuário para edição no caso de uso 18 – editar usuário.	144
Figura 58 - Tela para edição do cadastro do usuário no caso de uso 18 – editar usuário.	145
Figura 59 - Tela com todos os usuários cadastrados no caso de uso 18 – editar usuário.	145

## Lista de Tabelas

Tabela 1 - Papéis e competências necessárias para desempenhá-los .....	29
Tabela 2 - Classificação de jogos por público alvo .....	55
Tabela 3 – Características de quantidade e relação entre jogadores, da duração e do espaço necessário para o jogo. ....	56
Tabela 4 – Exemplo da classificação de jogos .....	57
Tabela 5 – Categoria geral do modelo LOM .....	61
Tabela 6 - Categoria de ciclo de vida do modelo LOM.....	62
Tabela 7 - Categoria de meta-metadados do modelo LOM.....	62
Tabela 8 - Categoria de metadados técnicos do modelo LOM.....	63
Tabela 9 - Categoria de metadados educacionais do modelo LOM.....	65
Tabela 10 - Categoria de metadados sobre direitos autorais do modelo LOM .....	65
Tabela 11 - Categoria de metadados sobre relacionamentos entre objetos do modelo LOM .....	66
Tabela 12 - Categoria de metadados para anotação do modelo LOM.....	66
Tabela 13 - Categoria de metadados para classificação do modelo LOM .....	67
Tabela 14 – Comparativo entre os tipos de arquitetura para os servidores de objetos de aprendizagem e metadados .....	69
Tabela 15 - Conhecimento nas entidades de uma comunidade e suas características. ....	72
Tabela 16 – Requisitos considerados e características gerais dos “repositórios” avaliados .....	87
Tabela 17 – Distribuição das informações dos jogos encontrados dos metadados propostos.....	92
Tabela 18 - Grupo de metadados sobre informações gerais.....	94
Tabela 19 – Grupo de metadados sobre informações da criação do objeto. ....	95
Tabela 20 – Grupo de metadados sobre informações técnicas .....	96
Tabela 21 - Grupo de metadados sobre classificações gerais do jogo. ....	99
Tabela 22 – Grupo de metadados sobre direitos autorais.....	99
Tabela 23 - Grupo de metadados sobre comentários. ....	100
Tabela 24 - Grupo de metadados sobre classificações de aprendizagem.....	102
Tabela 25 – Grupo de metadados sobre classificações em relação à metodologia ágil Scrum. ....	103
Tabela 26 - Grupo de metadados sobre avaliação. ....	104
Tabela 27 – Requisitos funcionais .....	106
Tabela 28 – Requisitos não funcionais do SGR. ....	107
Tabela 29 - Tabelas e relacionamentos adicionados no modelo de banco de dados. ....	118
Tabela 30 – Resultados dos testes de aceitação dos casos de uso. ....	147
Tabela 31 – Resultados dos testes unitários nas classes utilitárias do SGR. ....	148
Tabela 32 – Aspectos de usabilidade, objetivos, questões, hipóteses e métricas utilizados na avaliação do SGR. ....	153

## Lista de Abreviaturas e Siglas

ABES. Associação Brasileira de Empresas de Software  
ASD. *Adaptive Software Development*  
CC. *Creative Commons*  
Crystal. *Crystal Family of Methodologies*  
CSS. *Cascading Style Sheets*  
DSDM. *Dynamic System Development Method*  
FDD. *Feature Driven Development*  
GPS. *Goal, Purpose and Scope*  
GQM. *Goal-question-metric*  
GQS. Grupo de Qualidade de Software  
HTML. *HyperText Markup Language*  
IEC. *International Electrotechnical Commission*  
IEEE. *Institute of Electrical and Electronics Engineers*  
IGR. *Instructional Games Repository*  
INCoD. Instituto Nacional para Convergência Digital  
INE. Departamento de Informática e Estatística  
ISO. *International Organization for Standardization*  
LO. *Learning Object*  
LOM. *Learning Object Metadata*  
MVC. *Model-view-controller*  
PMI. *Project Management Institute*  
PO. *Product Owner*  
SCORM. *Sharable Content Object Reference Model*  
SGBD. Sistema de Gerenciamento de Banco de Dados  
SGR. *Scrum Games Repository*  
SQL. *Structured Query Language*  
UFSC. Universidade Federal de Santa Catarina  
XP. *Extreme Programming*

# Conteúdo

Lista de Figuras.....	8
Lista de Tabelas .....	10
Lista de Abreviaturas e Siglas .....	11
1. Introdução.....	14
1.1. Motivação .....	14
1.2. Problema .....	16
1.3. Objetivos.....	17
1.4. Método de pesquisa .....	18
2. Fundamentação Teórica.....	21
2.1. Gerenciamento ágil de projetos e Scrum.....	21
2.2. Aprendizagem e jogos educacionais .....	39
□ Aprendizagem Experiencial: é centrada no aluno e orientada pelas atividades, sendo que a ênfase desta estratégia está no processo de aprendizado e não no produto. Ocorre quando.....	48
2.3. Objetos de aprendizagem e metadados .....	57
2.4. Repositório de Objetos de Aprendizagem.....	67
2.5. Comunidades de Prática.....	72
3. Estado da Arte .....	78
3.1. Requisitos .....	78
3.2. Critérios de inclusão e exclusão .....	78
3.3. Execução da busca .....	79
3.4. Resultados Observados.....	80
4. Solução desenvolvida .....	89
4.1. Especificação dos metadados dos jogos .....	89
4.2. Especificação dos requisitos e casos de uso.....	105
4.3. Arquitetura e modelagem do sistema .....	117
4.4. Interfaces de usuário .....	124
4.5. Implementação do sistema .....	125
4.6. Testes do sistema .....	145
5. Avaliação .....	149
5.1. Definição.....	149
5.2. Identificação dos objetivos, questões e métricas.....	150
5.3. Criação da plano de execução e seleção dos participantes .....	153
5.4. Execução .....	155
5.5. Análise dos dados .....	155
6. Conclusão.....	160
Referências .....	161

Apêndice A .....168  
Apêndice B.....172  
Apêndice C.....173

# 1. Introdução

Neste tópico é apresentado o cenário atual na área de desenvolvimento de *software*, o problema encontrado neste cenário e a proposta de projeto para solucioná-lo, com os objetivos e metodologia de pesquisa utilizados em seu desenvolvimento.

## 1.1. Motivação

As empresas brasileiras de desenvolvimento de *software* e prestação de serviços obtiveram um faturamento de cerca de 60,2 bilhões de reais no ano de 2013, um aumento de 26,7% em relação a 2012 (ABES, 2013, p. 5).

O crescimento acelerado e a complexidade crescente dos produtos e serviços estão levando as empresas do setor a buscar uma redução de custos e um aumento no controle, através de métricas e práticas para a identificação de falhas e gargalos na operação, uma melhor visibilidade dos processos e um planejamento mais estruturado (ABES, 2011, p. 18).

O gerenciamento de projetos vêm ao encontro destas necessidades com a identificação e acompanhamento dos requisitos de um projeto, e a aplicação de conhecimentos e ferramentas às atividades do projeto (PMI, 2008). Este gerenciamento pode ser feito da forma tradicional, que busca a previsão de cada fase com o objetivo de fornecer uma grande quantidade de informações acerca dos requisitos e do desenvolvimento do projeto, e com a entrega de todo o produto no final do projeto (CHARVAT, 2003). Outra alternativa é realizar este gerenciamento com o auxílio da metodologia ágil mais utilizada (de acordo a pesquisa realizada por PMSURVERY.ORG, 2012, p. 59): o Scrum. Esta metodologia proporciona um aumento no retorno do investimento, uma redução de custos, uma aumento nas chance de sucesso em um negócio com constantes mudanças e resultados mais rápidos para o cliente (RUBIN, 2012, p. 6).

O Scrum é uma metodologia para gerenciamento ágil de projetos baseada em ciclos curtos de inspeção e adaptação, com a priorização no desenvolvimento das funcionalidades que agregam mais valor ao negócio. Contando com um time auto organizado e multifuncional, um líder disposto a resolver qualquer tipo de problema e um cliente que centraliza as necessidades do produto, o Scrum consegue entregar um produto com mais funcionalidades a cada final de ciclo (SUTHERLAND; SCHWABER, 2007). Os benefícios

desta metodologia no fluxo de trabalho e no negócio ficam evidentes, segundo Cohn (2010, p. 10-17), com o aumento de mais de 80% na produtividade, a diminuição de um quarto do custo e a melhoria de cerca de 40% da qualidade dos projetos.

Para qualquer pessoa que tenha interesse em aprender o Scrum e desenvolvê-lo a ponto de obter os benefícios mencionados, é necessário não apenas o conhecimento sobre os conceitos básicos, mas, principalmente, sobre os princípios ágeis por trás das práticas utilizadas nesta metodologia (SHORE; WARDEN, 2008). O conhecimento sobre estes conceitos pode ser obtido através da leitura de livros, artigos e publicações não oficiais em *sites* e *blogs*. Para diversificar o aprendizado é possível estimular outras competências intelectuais, utilizando: mapas conceituais, linhas temporais, medidas estatísticas, jogos e simulações (GARDNER, 2011).

Uma alternativa à leitura, que estimula diversas inteligências, é a participação em cursos de treinamento, com o objetivo de ampliar as capacidades e melhorar o desempenho no trabalho (FORLI INOCENTE, 2006). Entretanto, os poucos cursos de treinamento existentes para o Scrum, como os fornecidos pela Caelum (2013), Globalcode (2013) e Adaptworks (2013), e a pequena duração deles impede a abordagem de uma das partes mais importantes da metodologia: os valores e princípios ágeis.

A falta de disciplinas específicas para o ensino do Scrum nas universidades é outro problema, como nos curso de graduação em Sistemas de Informação oferecidos pela UFSC (INE, 2012), PUCRS (2012), UNISINOS (2012) e UFPE (2012), que oferecem, no máximo, disciplinas de gerenciamento de projetos.

Com o intuito de complementar o aprendizado dos conceitos e transmitir a importância dos princípios, os jogos educativos podem ser utilizados para o ensino do Scrum. “O jogo é uma atividade física ou mental organizada por um sistema de regras” (HAIDT, 2001, p. 175).

Segundo Dewey e Piaget, a utilização de jogos oferece ao aluno uma aprendizagem pela conquista de experiências pessoais, de verdades criadas por ele próprio; em oposição ao aprendizado pela repetição e conservação de conceitos, de verdades acabadas (GADOTTI, 1997). O jogo possui um forte teor motivacional, é capaz de absorver o aluno de forma intensa e total, e é considerado uma atividade lúdica, natural do ser humano: joga-se pelo simples prazer de jogar (HAIDT, 2001).

Para beneficiar-se deste recurso didático, o professor pode criar seus próprios jogos, de acordo com o conteúdo a ser estudado e os objetivos de ensino-aprendizagem (HAIDT,

2001) ou re-utilizar os jogos existentes, utilizando o formato de objetos de aprendizagem. Neste formato, o material (jogo) é modular, pode ser aproveitado em diferentes contextos e possui informações para classificação (LONGMIRE, 2000).

## 1.2. Problema

Jogos para o ensino do Scrum podem ser encontrados nos *sites* de seus criadores, em *blogs* e *wikis* sobre gerenciamento ágil de projetos ou sobre Scrum, como o TastyCupCakes (MCCULLOUGH, 2010) e em grupos de discussão sobre ensino com o auxílio de jogos, como o *Agile Games* (HANOULLE, 2013). Apesar da quantidade de jogos disponibilizada ser considerável e da comunidade formada em torno destes sistemas ser consistente e participativa, nenhum deles apresenta uma ferramenta para a busca de jogos por papéis, artefatos e cerimônias do Scrum e, princípios e valores do gerenciamento ágil.

Considerando este cenário e buscando melhorias no processo de pesquisa por jogos educacionais sobre o Scrum, apresenta-se como solução, o desenvolvimento ou modificação de um repositório de jogos (objetos de aprendizagem) para o ensino do Scrum.

Este repositório precisará transpor dificuldades de ordem jurídica - direitos autorais, por exemplo - e de ordem computacional, como inconsistências na classificação do material e falhas técnicas. Para minimizar estes problemas, pode ser estimulada a criação de um grupo de pessoas – uma comunidade de prática – com os mesmos interesses, que compartilhe as mesmas necessidades sobre o uso e a busca de jogos educativos, e a mesma paixão pelo ensino de gerenciamento ágil de projetos e Scrum (WENGER, MCDERMOTT e SNYDER, 2002).

## **1.3. Objetivos**

### **Objetivo Geral**

O objetivo deste trabalho é desenvolver um repositório de jogos para o ensino dos conceitos, práticas e princípios do Scrum, e que possa ser utilizado pela comunidade internacional.

Este repositório deve possibilitar a busca, cadastro, edição e remoção de jogos, com informações necessárias para descrição, execução, classificação e avaliação de um jogo. Devem ser disponibilizados filtros que permitam a busca de jogos por estas informações de classificação, como, por exemplo, os papéis da equipe do Scrum, o número de participantes necessários para o jogo, a duração média e a licença em que o jogo deve ser distribuído.

Para que estas operações possam ser utilizadas de forma controlada, deve ser oferecido um cadastro de usuários com informações básicas de identificação. O controle destes usuários e dos jogos cadastrados por eles deve ser disponibilizado para usuários que estejam em um grupo com perfil de administração.

### **Objetivos específicos**

- Análise da fundamentação teórica nas áreas do Scrum, aprendizagem, jogos educacionais, objetos de aprendizagem, metadados, repositórios de objetos de aprendizagem e comunidades de prática;
- Análise do estado da arte sobre jogos para o ensino de Scrum, e sobre repositórios para estes jogos;
- Desenvolvimento de um repositório para buscar, cadastrar e visualizar informações sobre jogos para o ensino de Scrum;
- Análise e avaliação do repositório a partir de requisitos funcionais e de usabilidade.

## **Limites da pesquisa**

- Foco exclusivo no Scrum, excluindo outras metodologias para gerenciamento de projetos;
- Foco exclusivo na utilização de jogos para o ensino de Scrum ou de conceitos relacionados ao Scrum, excluindo simulações, dramatizações (*role-playing*) e demais trabalhos em grupo;
- Foco exclusivo em jogos manuais, excluindo os que necessitem de interação com um dispositivo eletrônico.

### **1.4. Método de pesquisa**

O presente trabalho tem como fim gerar conhecimentos para aplicação prática através de um repositório de jogos para o ensino do Scrum. O problema da falta de repositórios e a proposta de solução são explicitados através do levantamento bibliográfico e da revisão do estado da arte (SILVA, 2005, p. 19-22).

Na primeira etapa do trabalho é efetuada a revisão teórica dos principais conceitos envolvendo Scrum, jogos e repositórios; na segunda etapa é efetuada a busca e análise dos repositórios disponíveis à comunidade; na terceira etapa são definidos os requisitos necessários para o repositório e é realizado o desenvolvimento da solução propriamente dita; e, na quarta e última etapa, é executada a análise e avaliação do repositório desenvolvido.

#### **Etapa 1 - Revisão teórica**

Principais etapas da revisão teórica:

- 1.1 – Revisão teórica sobre gerenciamento ágil de projetos e Scrum;
- 1.2 – Revisão teórica sobre aprendizagem e jogos educacionais;
- 1.3 – Revisão teórica sobre objetos de aprendizagem e metadados;
- 1.4 – Revisão teórica sobre repositórios de objetos de aprendizagem
- 1.5 – Revisão teórica sobre comunidades de prática.

## **Etapa 2 – Análise do Estado da Arte**

Para analisar o estado da arte, é realizada uma revisão sistemática da literatura (KITCHENHAM et al, 2009), identificando os repositórios de objetos de aprendizagem existentes para o ensino do Scrum e especificando os pontos fortes e fracos de cada um.

Principais etapas da análise do estado da arte sobre repositórios:

- 2.1 – definição dos requisitos de um repositório;
- 2.2 – definição dos critérios de inclusão e exclusão da busca;
- 2.3 – definição dos termos e execução da busca; e
- 2.4 – extração das informações e análise;

## **Etapa 3 – Desenvolvimento**

Nesta etapa, um repositório de jogos é desenvolvido ou modificado para compartilhar objetos de aprendizagem que auxiliam no aprendizado do Scrum.

Primeiramente, são efetuadas a definição e a validação do modelo de metadados, para possibilitar a especificação dos requisitos e a documentação dos casos de uso. Com estas informações, a modelagem do repositório e dos metadados dos objetos de aprendizagem é criada. Em paralelo a isto, é realizada a análise dos repositórios existentes, com o objetivo de verificar a possibilidade de adaptação de algum destes.

Após as avaliações e modificações, o sistema é desenvolvido com base nos requisitos descritos. Ao concluir o desenvolvimento, o sistema é instalado e avaliado através de testes de aceitação.

Etapas secundárias do desenvolvimento do repositório:

- 3.1 – especificação dos metadados
- 3.1 – especificação dos requisitos;
- 3.2 – interfaces de usuário;
- 3.3 – implementação do sistema; e
- 3.4 – testes do sistema.

## **Etapa 4 – Aplicação e avaliação**

Para verificar se o sistema desenvolvido possui as funcionalidades necessárias é realizada uma avaliação utilizando o modelo *GQM* (Basili, 1994), que por meio da definição de objetivos, elaboração e aplicação de questões, extração e análise corretiva, e análise preventiva, tem como objetivo validar o desenvolvimento e direcionar as correções e melhorias.

Este modelo de avaliação é utilizado através de um questionário aplicado a um grupo de especialistas na área de gerenciamento ágil de projetos com Scrum.

Etapas secundárias da avaliação:

- 4.1 – Definição da metodologia de avaliação;
- 4.2 – Identificação dos objetivos, questões e métricas;
- 4.3 – Criação do plano de execução e seleção dos participantes;
- 4.4 – Execução;
- 4.5 – Análise dos dados;

## 2. Fundamentação Teórica

Neste capítulo são estabelecidos os principais conceitos utilizados ao longo do projeto, nos domínios do gerenciamento ágil de projetos e do Scrum; das estratégias de ensino, dos jogos; dos objetos de aprendizagem e repositórios, dos metadados e das comunidades de práticas.

### 2.1. Gerenciamento ágil de projetos e Scrum

Nesta seção, são abordados os princípios e práticas do gerenciamento ágil e os conceitos sobre os artefatos, papéis e cerimoniais da metodologia Scrum.

#### 2.1.1. Gerenciamento ágil de projetos

A base do gerenciamento ágil de projetos foi estabelecida com o surgimento do movimento ágil para gerenciamento e desenvolvimento de projetos de *software*.

O movimento ágil iniciou-se na década de 90, com a divulgação dos princípios e práticas do Desenvolvimento de *Software Lean* (Poppendieck, 2002), da metodologia Scrum para gerenciamento de projetos (SCHWABER, 1996) e do *Extreme Programming*, mais conhecido como *XP* (BECK, 2004).

As metodologias para gerenciamento ágil de projetos são compostas por elementos estruturais (ferramentas, cerimoniais e papéis) e elementos semânticos, que fornecem o embasamento para a utilização dos elementos estruturais chamados de práticas, princípios e valores (SHORE e WARDEN, 2008).

As **práticas** são o conhecimento aplicado ao dia-a-dia, são técnicas claras e objetivas, que fornecem um ponto de início, um referencial (BECK, 2004). Elas são combinadas em uma forma única, acentuando os conceitos que suportam a filosofia ágil e descartando os demais (SHORE e WARDEN, 2008).

Diferentemente das práticas, os **valores** fornecem raízes e pontos de julgamento: o que é certo ou errado e o que aprova-se ou desaprova-se. “Os valores trazem propósitos às práticas. Explicitar os valores é importante porque sem eles, as práticas tornam-se um hábito, tornam-se atividades executadas sem propósito e direção” (BECK, 2004, p. 23,

tradução nossa).

As pontes conceituais entre os valores e as práticas são os **princípios**. Como os valores são conceitos muito abstratos para guiar diretamente um comportamento, os princípios servem de guia para um domínio específico (BECK, 2004).

As práticas, princípios e valores do *XP*, do Scrum e de outras metodologias utilizados na década de 90 foram resumidos, reunidos e explicitados através do Manifesto Ágil (BECK et al, 2001), que clareou os objetivos centrais das metodologias ágeis.

O Manifesto Ágil tem como base quatro valores centrais:

- Indivíduos e interações são mais importantes do que ferramentas e processos;
- *Software* em funcionamento é mais importante do que documentação abrangente;
- Colaboração com o cliente é mais importante do que negociação de contratos;
- Resposta a mudanças é mais importante do que seguir um plano.

Com os valores centrais estabelecidos, foi criada uma lista de princípios que devem ser seguidos para se obter um maior sucesso no gerenciamento e desenvolvimento ágil de projetos de *software*. Os doze princípios são listados a seguir (BECK et al, 2001):

- Atendimento a solicitações do cliente – “A maior prioridade deve ser em satisfazer o cliente entregando *software* com valor agregado de forma contínua, o mais cedo possível”;
- Favorecimento a mudanças – “Aceitar a mudança de requisitos, mesmo sendo em etapas finais do projeto, garantindo vantagem competitiva ao cliente”;
- Entregas de *software* frequentes – “Entregar *software* em funcionamento frequentemente, com um intervalo preferencial de poucas semanas”;
- Desenvolvimento em equipe - “Desenvolvedores e responsáveis pelo negócio devem trabalhar juntos durante todo o projeto, diariamente”;
- Ambiente motivador - “Desenvolver projetos com pessoas motivadas e garantir um ambiente e suporte necessários, para que, com a confiança necessária, estas possam fazer o trabalho direito”;
- Conversação direta - “A forma mais efetiva de compartilhar informações entre o time é a conversação direta, face a face”;
- Funcionamento do *Software* como métrica - “A principal medida de progresso é

o *software* em funcionamento”;

- Desenvolvimento constante e sustentável - “Patrocinadores, desenvolvedores e usuários devem ser capazes de manter um ritmo constante de trabalho, garantindo a sustentabilidade no desenvolvimento do projeto”;
- Excelência técnica - “Atenção contínua à excelência técnica e design de boa qualidade aumentam a agilidade”;
- “Simplicidade – a arte de maximizar a quantidade de trabalho não feito - é essencial”;
- Times auto-organizados - “As melhores arquiteturas, requisitos e design surgem de times auto-organizados”;
- Melhoria contínua - “O time, em intervalos regulares, deve refletir sobre ações para tornar-se mais efetivo, aplicando e ajustando o comportamento”.

Com a declaração dos princípios e valores utilizados na comunidade ágil, houve um melhor entendimento e visibilidade destas “novas” metodologias, aumentando o interesse e a utilização por parte da comunidade de *software*.

## Os conceitos do Gerenciamento Ágil de projetos

O gerenciamento de projetos é a aplicação de conhecimentos, habilidades, ferramentas e técnicas às atividades do projeto a fim de atender aos seus requisitos (PMI, 2008, p. 12). O projeto, segundo o PMI (2008, p. 11), “é um esforço temporário empreendido para criar um produto, serviço ou resultado exclusivo”.

“Quando uma metodologia para gerenciamento de projetos é **incremental**, com pequenos protótipos e ciclos rápidos; **cooperativa**, com comunicação intensa e trabalho em equipe; **direta**, de fácil aprendizado e modificação; e **adaptativa**, que permite modificações até as últimas fases do projeto, pode-se afirmar que é uma metodologia ágil para gerenciamento de projetos” (ABRAHAMSSON et al, 2002, p. 19, tradução nossa, grifo do autor).

O Scrum é uma das únicas metodologias ágeis para gerenciamento de projetos e, segundo PMSURVEY.ORG (2011, p. 61), é a mais utilizada, superando por ampla margem a

utilização de outras, como XP, ASD, Crystal, DSDM e FDD.

A metodologia Scrum foi escolhida por vários fatores, entre eles, maior produtividade, custos menores e maior satisfação dos interessados no produto, como apontado na introdução do presente trabalho.

### **2.1.2. Scrum**

“Scrum é uma metodologia simples utilizada para organizar times e fazer o trabalho de uma forma mais produtiva e com uma maior qualidade” (SUTHERLAND; SCHWABER, 2007, p. 14, tradução nossa).

No Scrum, o trabalho começa com a seleção de funcionalidades, por parte do *Product Owner*, de acordo com os desejos dos clientes e interessados no produto. Com a seleção de funcionalidades é criada uma lista ordenada em que as funcionalidades com maior valor de negócio estão “em cima”, prontas para serem selecionadas e desenvolvidas pelo Time. A seleção das funcionalidades acontece nas reuniões de planejamento e resultam em uma lista pequena (se comparada com lista de funcionalidades do produto), que contém apenas as funcionalidades mais importantes para os clientes, chamada de lista de funcionalidades da *Sprint*. Com a posse das funcionalidades que serão desenvolvidas durante a *Sprint*, o time pode começar o processo de desenvolvimento, reportando diariamente o desempenho e os problemas encontrados para o *Scrum Master* e para os demais membros do time. Ao fim da *Sprint* o time deve se reunir com o *Product Owner*, para apresentar as funcionalidades que foram completamente desenvolvidas. Após os comentários e ajustes recomendados pelo *Product Owner*, o time realiza uma reunião de retrospectiva buscando a melhoria do processo internamente. Esta reunião, de retrospectiva, não marca apenas o fim de uma *Sprint*, mas o começo de outra, fornecendo insumos que serão abordados nas reuniões de planejamento (DEEMER; BENEFIELD, 2006).

Uma ilustração do processo principal do Scrum, desde a seleção de funcionalidades até a reunião de retrospectiva, é apresentada na figura 1.

# SCRUM

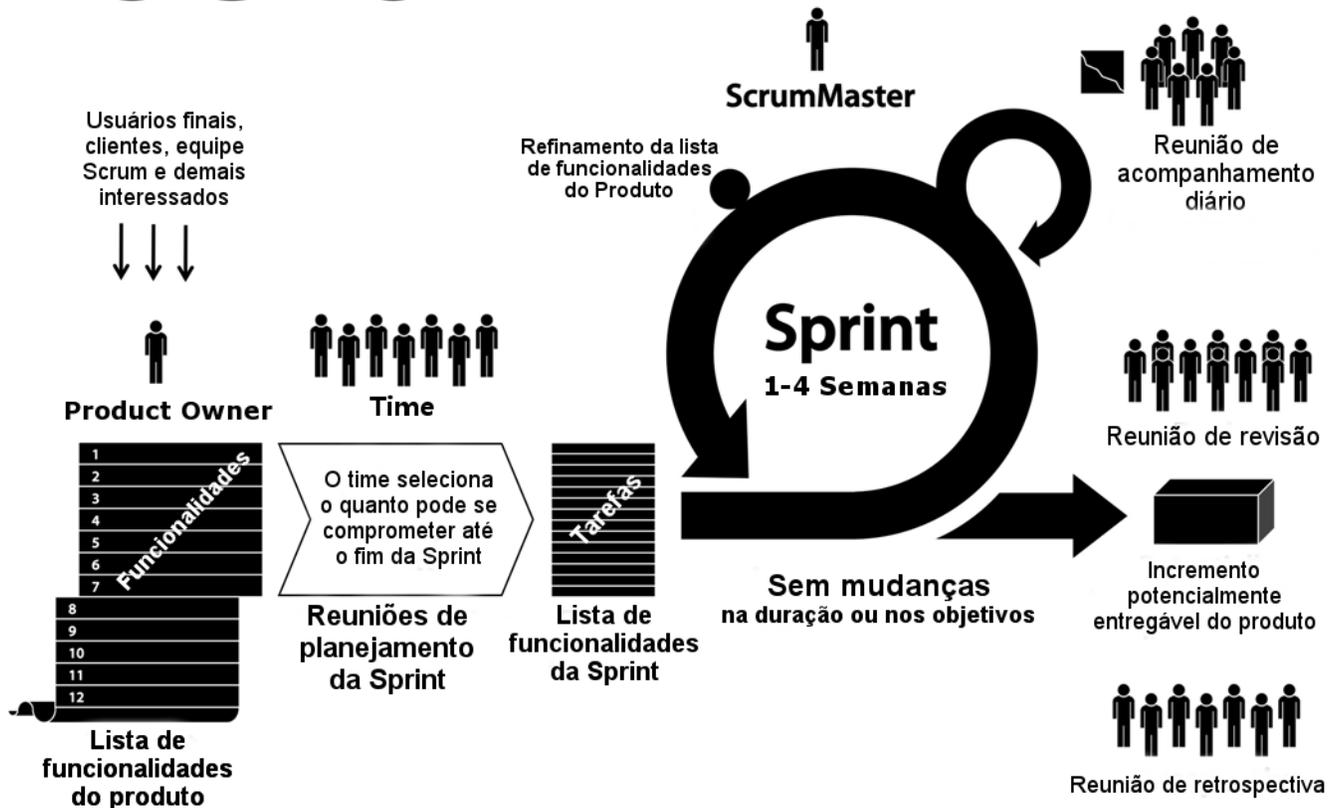


Figura 1 - Papéis, marcos e ciclo de vida do Scrum (DEEMER; BENEFIELD, 2006, tradução nossa).

O gerenciamento de projetos com esta metodologia deve ser realizado por meio de profissionais dedicados e comprometidos, que exercem papéis específicos, trabalhando em ciclos de desenvolvimento com adaptação e entrega de valor constante. A descrição dos papéis que precisam ser desempenhados e do ciclo de desenvolvimento é realizada a seguir.

## 2.1.2.1. Papéis da equipe

O fluxo de trabalho dinâmico e as constantes demandas dentro de um ciclo de desenvolvimento requerem um trabalho sincronizado com responsabilidades bem definidas. O Scrum possui define três papéis distintos para executar as tarefas necessárias para o

desenvolvimento do projeto: *Product Owner*, *Scrum Master* e time de desenvolvimento.

As atividades e responsabilidades herdadas de metodologias de gerenciamento de projetos e desenvolvimento de *software*, como as do gerente de projetos, do analista de negócio e do especialista em bancos de dados, devem ser distribuídas entre os componentes da equipe<sup>1</sup>.

### ***Product Owner***

O *Product Owner*, conhecido simplesmente por *PO*, atua como centralizador das expectativas das pessoas interessadas no produto (consumidores, financiadores, gerentes, diretores), decidindo o que é necessário desenvolver e em que ordem isso deve ser feito (SCHWABER, 1996, p. 16). Com esta responsabilidade, o *PO* torna-se responsável direto pelo sucesso da solução que está sendo desenvolvida (RUBIN, 2012).

Segundo Cohn (2010, p. 125-130), para que o *PO* consiga especificar para o time quais funcionalidades devem ser desenvolvidas, é necessário muito diálogo, interação e, principalmente, a definição de uma lista com as funcionalidades para o produto (*product backlog*). O *PO* deve manter esta lista sempre priorizada em função da importância – valor de negócio – das funcionalidades, para os interessados no produto, buscando, desta forma, um bom retorno do investimento.

Para guiar o desenvolvimento e avaliar o produto resultante, o *PO*, em conjunto com o time, deve estabelecer critérios que permitam qualificar uma funcionalidade como “pronta” (RUBIN, 2012, p. 169).

### ***Scrum Master***

O *Scrum Master* tem duas grandes responsabilidades: de auxiliar a todos no entendimento e na utilização do Scrum, e de atuar como um líder para o time de desenvolvimento.

---

<sup>1</sup> A equipe é composta pelas pessoas que desempenham os papéis do Scrum: time de desenvolvimento, *Product Owner* e *Scrum Master*.

Para auxiliar na aplicação desta metodologia, o *Scrum Master* deve agir como um treinador, liderando e organizando o processo de adoção ou adaptação do Scrum. Para, posteriormente, auxiliar na adaptação às necessidades e especificidades do projeto e da empresa (RUBIN, 2012, p. 16). Sempre que o *Scrum Master* perceber que houve uma estagnação ou que existe uma oportunidade de melhoria, é responsabilidade dele atuar como um agente de mudanças, promovendo a melhoria na atividade ou processo (RUBIN, 2012, p. 187).

Como facilitador do time, o *Scrum Master* deve auxiliar na resolução de problemas, na remoção de impedimentos e na proteção contra pressões (demasiadas) e interferências externas, proporcionando um fluxo de trabalho contínuo (COHN, 2010, p. 117-119).

### **Time de desenvolvimento**

O time de desenvolvimento tem como principal responsabilidade o desenvolvimento do produto. Para que isso seja atingido, ele deve executar as tarefas de análise de requisitos, programação, integração de sistemas, execução de testes, criação da interface do usuário e avaliação de tempo necessário para entregar uma funcionalidade. O desenvolvimento destas tarefas, em conformidade com a proposta de produto incremental do Scrum, só é possível se o time inspecionar o que faz diariamente e agir em cima desta análise (RUBIN, 2012, p. 196,197).

A lista de funcionalidades do produto sofre um processo de adaptação semelhante no qual, durante a *Sprint*, o time deve auxiliar o PO nas tarefas de criação, refinamento, estimação e priorização (RUBIN, 2012, p. 197).

### **Competências necessárias para a equipe do Scrum**

O desenvolvimento do projeto e o sucesso do produto dependem diretamente da forma como o *Product Owner*, o *Scrum Master* e o time de desenvolvimento desempenham os seus papéis.

O conhecimento sobre tecnologia, processo e negócio, as habilidades pessoais e

interpessoais e as atitudes que facilitem a convivência em grupo, são apenas algumas das competências necessárias para que os integrantes da equipe possam desempenhar suas funções com excelência, apresentadas na tabela 1.

		<b>Scrum Master</b>	<b>Product Owner (PO)</b>	<b>Time</b>
<b>Conhecimento</b>	<b>Técnico</b>	Conhecimento geral sobre as tecnologias utilizadas no desenvolvimento, para auxiliar na resolução de problemas técnicos.	Conhecimento sobre o domínio do produto, variáveis de mercado, cálculos financeiros, e tudo mais que puder auxiliar na construção de um produto melhor.	Conhecimento diverso e balanceado sobre as tecnologias utilizadas.
	<b>Negócio</b>	Conhecimento geral sobre o negócio, para responder ou repassar dúvidas sobre o entendimento de funcionalidades.	Conhecimento profundo sobre mercado, concorrentes, clientes, usuários finais, para poder decidir sobre as funcionalidades que serão desenvolvidas.	Conhecimento geral sobre o negócio, aprofundado conforme o desenvolvimento das funcionalidades.
	<b>Processo (Scrum)</b>	Entendimento e experiência sólida para ensinar e aplicar o Scrum.	Conhecimento geral sobre o processo.	Conhecimento geral sobre o processo, aprofundado conforme o andamento do projeto.
<b>Habilidade</b>	<b>Técnica</b>	Nas ferramentas de análise e controle do desempenho do time, de exibição e manutenção da lista de funcionalidades; nos métodos para obtenção de estimativas e para garantia de qualidade do que foi desenvolvido.	Nas ferramentas de controle de qualidade do produto (incluindo defeitos).	Nas linguagens de programação, nas ferramentas e métodos de desenvolvimento, para poder desenvolver o produto com as tecnologias escolhidas.
	<b>Comunicação e Influência</b>	Para negociar com o PO sobre melhores condições de prazo e escopo, para sugerir novas técnicas e métodos, e para convencer a equipe dos benefícios do Scrum.	Para convencer os interessados no produto, sobre a importância de determinadas funcionalidades .	Para escolher a ordem de desenvolvimento das funcionalidades e, formas e métodos para este trabalho.
<b>Atitude</b>	<b>Poder e Autoridade</b>	Para alterar o processo de desenvolvimento.	Para tomar decisões sobre os rumos do produto e responder por elas.	Para desenvolver qualquer parte do produto e para responder por isso.
	<b>Disponibilidade</b>	Se colocar à disposição do time para responder as dúvidas sobre o processo.	Se colocar à disposição do time, do Scrum Master e dos interessados no produto para responder às dúvidas sobre o negócio e o produto.	Para aprender mais sobre desenvolvimento, processo e negócio.
	<b>Responsabilidade</b>	Pelo time e pelo processo, para cumprir com os acordos firmados.	Pelo sucesso do produto, por motivar os interessados e comprometidos no projeto a continuar trabalhando.	Coletiva, pela qualidade e conformidade do produto frente ao que foi acordado.
	<b>Humildade</b>	Para colocar as necessidades do time em primeiro lugar.	Para perceber que certas decisões tomadas precisam ser revistas e que todos podem contribuir para a melhoria do produto.	Para melhorar o código escrito, para receber novas ideias e para criar um ambiente colaborativo.

	<b>Scrum Master</b>	<b>Product Owner (PO)</b>	<b>Time</b>
<b>Colaboração</b>	Para criar e manter um ambiente de colaboração, encorajando o time a encontrar soluções aos problemas, que beneficiem a todos.	Para mediar as escolhas de negócio com os interessados no produto, buscando atender a necessidade de todos.	Para buscar soluções em que todos tenham a ganhar.
<b>Comprometimento</b>	Para liderar o time no desenvolvimento do produto, para bloquear interferências e para remover impedimentos.	Fazer o que for preciso (planejamento de testes, verificação manual do que foi desenvolvido, etc.) para que seja desenvolvido o melhor produto possível.	Facilitar no envolvimento do time em tarefas além do desenvolvimento, e na busca por um propósito, para que o time desenvolva o produto com compromisso.
<b>Confiança</b>	No time e no processo.	No produto, no time e no processo.	No produto, no time e no processo.
<b>Melhoria Contínua</b>	Para evoluir o processo e influenciar a equipe a evoluir em suas atividades.	Para evoluir o produto e a comunicação com as pessoas.	Para evoluir o produto.
<b>Simplicidade (eliminação do desperdício)</b>	No processo, para torná-lo mais enxuto e eficaz.	No produto, para torná-lo mais simples de usar e lucrativo, e na comunicação, para torna-la mais fácil e objetiva.	No desenvolvimento, utilizando tecnologias mais simples de aprender e aplicar, para desenvolver mais rápido, com mais qualidade e menor esforço.
<b>Envolvimento</b>	No ensino do Scrum à equipe, no processo de desenvolvimento e na análise geral do produto.	Desde a concepção da ideia do produto até a entrega ao cliente final.	Em todo o processo de desenvolvimento, desde a especificação de requisitos até a entrega do produto.

Tabela 1 - Papéis e competências necessárias para desempenhá-los (COHN, 2002, p. 117-217; RUBIN, 2012, p. 166-223)

O desenvolvimento das competências de cada membro da equipe, no nível e extensão comentados, é fundamental para exercer o trabalho na forma adaptativa, incremental e colaborativa, assim como a realização das cerimônias e utilização dos artefatos presentes nas fases do Scrum.

Segundo Schwaber (1996, p. 12), as atividades do Scrum podem ser distribuídas em três fases: o *Pregame*, que é caracterizado pelo conjunto de ações de conceituação e análise do produto; o *Game (Sprints)*, que são os ciclos de trabalho do Scrum; e o *Postgame*, que é a fase de fechamento do produto ou de uma versão deste.

### 2.1.2.2. **Sprint: O processo de desenvolvimento do produto**

*Sprint* é a fase do Scrum na qual ocorre a análise dos requisitos, o desenvolvimento, a

especificação de interface do usuário e todas as outras tarefas necessárias para a criação das funcionalidades.

“Com um tempo fixo de duração, normalmente curto e consistente, cada ciclo de trabalho possui um objetivo que não deve ser alterado depois do início da Sprint, e precisa alcançar um estado final conforme especificado pelo time” (RUBIN, 2012, p. 61).

O início da *Sprint* ocorre com o planejamento das funcionalidades que são desenvolvidas durante o período, sendo que a cada dia é efetuado um controle para verificar o desempenho e os problemas encontrados. Ao final deste período são feitas duas reuniões de fechamento, uma voltada ao produto e a outra à equipe. Estas fases são descritas de forma aprofundada a seguir.

### **Planejamento da *Sprint***

Para planejar o que é executado na *Sprint* são realizadas reuniões para selecionar e estimar o esforço das funcionalidades que serão desenvolvidas. As técnicas, como *Planning Poker*, histórias de usuário, e definição de pronto, e os detalhes do planejamento são abordados em seguida.

A reunião de planejamento é realizada para que o time, com o intermédio do *Scrum Master* e do *Product Owner*, entenda ou revise as funcionalidades do *product backlog*, se comprometa com um objetivo para a *Sprint* e de acordo com este, selecione as funcionalidades de maior prioridade, que podem ser realmente completadas durante o próximo ciclo (DEEMER; BENEFIELD, 2006).

Uma das formas de especificar as funcionalidades é por meio do formato de *User Stories*, utilizado para lembrar dos objetivos, do que se considera como pronto e dos detalhes importantes de uma funcionalidade. Ex.: “Um usuário pode visualizar informações de cada oferta de trabalho resultantes de uma busca” (COHN, 2004, p. 4; 6). No exemplo citado, percebe-se a utilização do papel “usuário” para guiar a especificação e posteriormente, o desenvolvimento e o teste da funcionalidade. Este modelo busca simular a utilização de funcionalidades do produto pelos tipos de usuários previstos no sistema, proporcionando diferentes perspectivas a equipe do Scrum (COHN, 2004, p. 31-33).

Ao escrever uma história é importante que esta seja independente de outras, que seja

negociável (permitindo futuras negociações sobre detalhes do escopo), que contenha valor para os usuários e que possa ser estimada e testada (COHN, 2004, p. 17-28).

Cada estória deve ser quebrada em partes menos complexas e dispendiosas, chamadas de tarefas. O conjunto destas, associado à funcionalidade da qual foram originadas, compõe uma segunda lista de funcionalidades, específica da *Sprint*, chamada de *Sprint Backlog* (RUBIN, 2012, p. 22).

Para confirmar que o trabalho selecionado pode ser concluído na *Sprint*, o time realiza uma estimativa do esforço (ou duração) necessária para desenvolver cada estória. Esta estimativa está atrelada ao conhecimento prévio do time sobre aspectos em volta do produto, como: experiência de trabalho, adaptação do sistema atual, complexidade do negócio, tecnologia empregada e risco associado ao desenvolvimento.

Uma das técnicas mais conhecidas para a obtenção de estimativas é o *Planning Poker*. Nela, os participantes estimam uma funcionalidade e escolhem a carta do baralho que exprime com maior precisão o tempo e o esforço necessários para o desenvolvimento. O conjunto das cartas escolhidas pelos participantes revela a estimativa para o item do *backlog* em questão (PICHLER, 2010, p. 66). Segundo Greening (2002), os principais passos para a utilização desta técnica, são:

- Preparação: os participantes sentam em uma mesa, virados para o centro, e recebem um conjunto igual de cartas de baralho, com os possíveis valores para uma estimativa;
- Escolha: Cada um escolhe, em segredo, a carta do baralho que lhe parecer mais coerente com a estimativa e a leva para o centro da mesa, com a face virada para baixo;
- Revelação: Quando todos estiverem com as cartas no centro da mesa, elas são viradas simultaneamente, revelando as estimativas do time;
- Justificativa: Se alguma carta revelar uma estimativa discrepante, o participante é convidado a justificar, revelando os fatores que o levaram a escolha de tal valor.
- Finalização: Quando houver uma homogeneidade no valor das cartas, a estimativa está concretizada. Caso contrário, são feitas novas rodadas até que uma estimativa uniforme seja atingida.

O valor nas cartas utilizado nas estimativas pode ser expresso em escala numérica,

com a série de Fibonacci: “1,2,3,5,8,13...”, ou com o Planning Poker original: “1,2,3,5,7,10,∞”; ou qualitativa, com Pontos de Estória, Pontos de Casos de Uso ou Pontos de Função. Esses valores podem representar diretamente a quantidade de dias necessários para finalizar uma funcionalidade ou podem ser relacionados ao esforço requerido para tal empreendimento.

Ao fim das rodadas de estimativas, uma revisão deve ser efetuada a fim de se identificar falhas no processo. Tudo que possa impedir um entendimento completo das funcionalidades pelos participantes, uma estimativa assertiva e sincera (sem pressão), e um ambiente isolado de interrupções e integrador para o time, deve ser identificado e avaliado, com a escolha de uma proposta de solução para o problema.

Associado à estimativa, deve haver uma descrição dos requisitos necessários para que uma funcionalidade seja aceita como pronta (definição de pronto). Nesta especificação, deve constar todos os critérios de implementação, teste e documentação que o *Product Owner* e o time julgarem necessários (PICHLER, 2010, p. 99).

A essência da fase planejamento, com a reunião dos comprometidos com o produto, a seleção de funcionalidades do *product backlog* (compondo o *sprint backlog*), a realização das estimativas utilizando *planning poker* e a revisão do processo, podem ser visualizadas na figura 2.

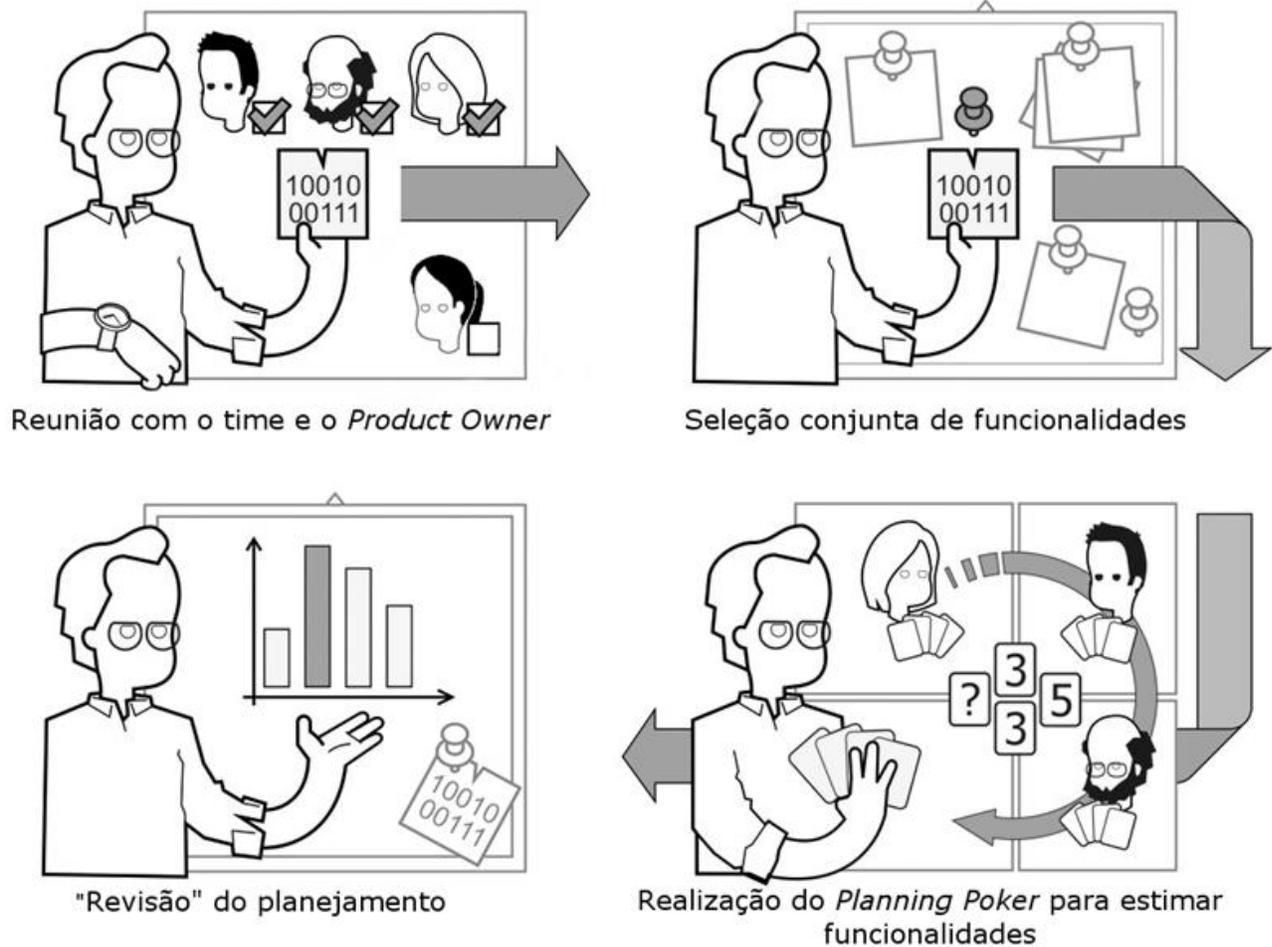


Figura 2 - Etapas da fase de planejamento da *Sprint* (adaptado de SPARTEZ, 2013).

## Execução da *Sprint*

No Scrum, a fase de desenvolvimento<sup>2</sup> das funcionalidades é caracterizada como imprevisível. Para controlar o risco e gerenciar esta imprevisibilidade são utilizados mecanismos de controle (SCHWABER, 1996, p. 10), como o quadro de tarefas, o gráfico de desempenho e as reuniões diárias.

O Scrum não especifica diretamente nenhum método ou técnica para o desenvolvimento, deixando uma abertura para que as equipes procurem o que mais se

<sup>2</sup> Nesta seção, o conceito de desenvolvimento, quando isolado, é associado às tarefas de análise, *design*, desenvolvimento, propriamente dito, e teste das funcionalidades.

adapta a suas necessidades.

O desenvolvimento de tarefas similares de modo sequencial (*waterfall*), por exemplo, realizando a análise e *design* de todas as funcionalidades antes de iniciar o desenvolvimento, pode parecer uma técnica segura e rápida, mas há um sério risco do time não conseguir testar todas elas até o fim do ciclo. Em um cenário utilizando esta metodologia de desenvolvimento, pode ocorrer a entrega de todas as funcionalidades com alguma parte faltando, quando o necessário seria entregar as funcionalidades de maior prioridade (RUBIN, 2012, p. 352).

Uma falha desse tipo, no fim da *Sprint*, pode ser evitada se o time organizar um cronograma de desenvolvimento das tarefas considerando a priorização por valor de negócio e a qualificação do membro do time que fará este trabalho. A viabilização deste fluxo de trabalho é realizada por meio do uso de práticas de gerenciamento e desenvolvimento ágil, que ofereçam rápido *feedback*, que possibilitem, aos membros do time, desenvolver qualquer tipo de tarefa e que suportem a constante evolução do produto (RUBIN, 2012, p. 353), como, por exemplo, o desenvolvimento orientado a testes, o *design* emergente e a integração contínua.

Para gerenciar o desenvolvimento e a utilização de práticas e ferramentas neste processo, o Scrum dispõe de meios de controle que evidenciam a existência de problemas ou atrasos no processo, para que possam ser resolvidos por parte do *Scrum Master* ou do *Product Owner*.

Por estes motivos, o Scrum determina a utilização de um quadro com as tarefas que serão desenvolvidas durante a *Sprint* (*task board*). Neste quadro, as tarefas de todas as funcionalidades selecionadas para a *Sprint* são colocadas em uma coluna de “Afazeres”. Quando um dos membros do time for iniciar a tarefa, ele desloca-a para outra coluna, identificada como “Em Progresso”. Quando a tarefa tiver sido finalizada o desenvolvedor a coloca na última coluna do quadro: “Completada” (RUBIN, 2012, p. 357).

O quadro citado anteriormente é apenas um exemplo do quadro de tarefas. O time tem a liberdade de definir colunas adicionais, conforme a necessidade. A figura 3 mostra um quadro de tarefas com as colunas refletindo os estados básicos do ciclo de desenvolvimento.

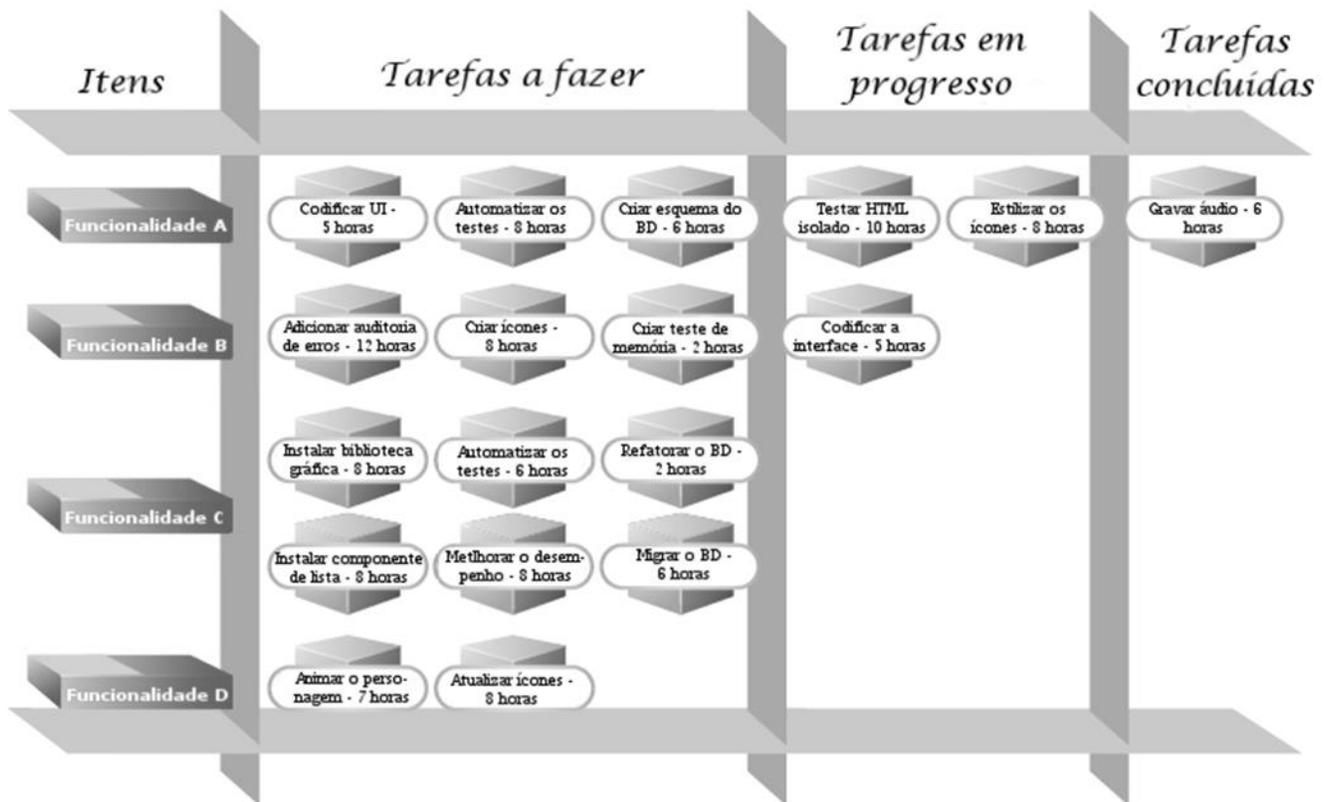


Figura 3 - Quadro de tarefas associado as funcionalidade (RUBIN, 2012, p. 356, tradução nossa).

Enquanto o quadro de tarefas serve como uma representação atual, diária, para o acompanhamento do desenvolvimento progredido, o gráfico de *burndown* exhibe o desempenho atingido desde o início da *Sprint* até o momento atual e o desempenho esperado até o fim do ciclo.

No gráfico de *burndown*, o eixo vertical representa a quantidade máxima de “pontos de estória” ou de horas de trabalho disponível para a equipe, e o eixo horizontal representa os dias de duração da *Sprint*. Este gráfico deve conter uma linha central, que indica o desempenho diário ideal, uma linha superior indicando que o desenvolvimento está atrasado (desempenho mínimo) e uma linha inferior indicando que o desenvolvimento está em um ritmo acelerado (alto desempenho).

A figura 4 representa um exemplo do gráfico de *burndown* para uma *Sprint* de duas semanas com o objetivo de entregar funcionalidades com um esforço máximo de quarenta “pontos de estória”.

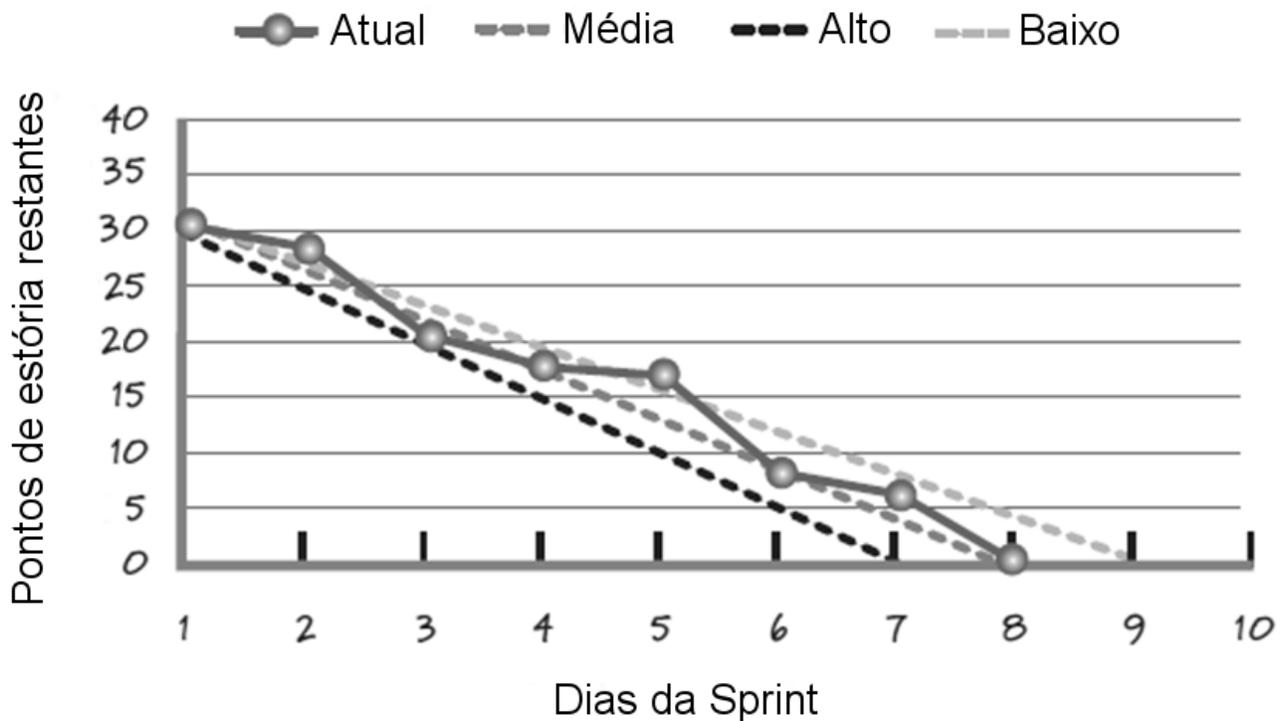


Figura 4 – Gráfico de burndown (adaptado de RUBIN, 2012, p. 327).

O acompanhamento do projeto e a atualização dos artefatos descritos - quadro de tarefas e gráfico de *burndown* - acontecem durante a reunião de controle diária do Scrum.

Mais conhecida como *Daily Meeting*, esta reunião de inspeção e adaptação deve ser rápida, com duração máxima de quinze minutos. Nela, busca-se a identificação e adaptação dos problemas de desenvolvimento, a distribuição de tarefas entre os membros do time e o acompanhamento do desempenho por parte do *Scrum Master*.

Para alcançar estes objetivos, cada membro do time tem três responsabilidades principais (SUTHERLAND e SCHWABER, 2010):

- explicar de forma clara e breve, as tarefas que foram desenvolvidas desde a reunião anterior;
- apontar o que pretende desenvolver até a próxima reunião; e
- apresentar os impedimentos ou obstáculos encontrados que impactaram no desenvolvimento.

A apresentação dos participantes é suportada pelo quadro de tarefas, de forma que

quando um membro do time apontar que determinada tarefa foi finalizada, ele deve movimentá-la para a coluna “Completada” do quadro. O mesmo deve acontecer para as demais mudanças de estado das tarefas.

Ao término da apresentação de cada participante, o Scrum Master deve somar a pontuação do que foi concluído, com base no quadro de tarefas, e atualizar o gráfico de *burndown* (SUTHERLAND; SCHWABER, 2007, p. 17).

## **Revisão do Produto**

Ao terminar a *Sprint*, o time, com o auxílio do *Scrum Master*, deve apresentar aos interessados no projeto o produto total do desenvolvimento, incrementado pelo trabalho realizado no ciclo anterior (RUBIN, 2012, p. 26).

Para viabilizar esta apresentação, é necessário determinar quem deve ser convidado, marcar uma data para reunião, confirmar que o trabalho da *Sprint* foi concluído, preparar a demonstração e determinar o papel de cada membro da equipe na apresentação.

A reunião de revisão oferece uma oportunidade importante para a equipe do Scrum receber questionamentos, críticas, idéias e sugestões (*feedback*), de pessoas que não estão disponíveis no dia-a-dia do desenvolvimento. Todos os interessados no produto, como executivos, gerentes, vendedores e clientes finais, podem comparecer a reunião (RUBIN, 2012, p. 365).

Conciliar a agenda de todos os interessados pode ser uma tarefa dispendiosa. Para minimizar esse problema é necessário identificar as pessoas que são essenciais para a reunião e consultá-las sobre uma data e horário adequado, facilitando o comparecimento (RUBIN, 2012, p. 366).

Quando o produto é apresentado para interessados de fora do círculo normal de pessoas que normalmente atendem a revisão, é necessário um esforço adicional na preparação da apresentação. Causar uma impressão negativa porque o *Product Owner* percebeu que determinada funcionalidade não está em total conformidade com o acordado no meio da apresentação, não é um *feedback* agradável (RUBIN, 2012, p. 367).

A preparação para a demonstração deve receber igual atenção por parte da equipe. Não se trata de elaborar uma apresentação enfeitada, com efeitos visuais e sonoros, mas

sim oferecer transparência e organização para facilitar a inspeção e adaptação do produto.

Como extensão da preparação deve haver uma separação do papel de cada membro do time na apresentação. Normalmente, cabe ao *Scrum Master* a responsabilidade de apresentar o produto, entretanto, pode ser conveniente que o *Product Owner* faça uma breve introdução à novos participantes, ou que um desenvolvedor faça uma explicação aprofundada quando questionado sobre o funcionamento de determinada funcionalidade (RUBIN, 2012, p. 368).

Através da apresentação do produto e avaliação por parte dos interessados, a equipe do Scrum recebe insumo para a evolução do produto. A inspeção e adaptação do processo ocorre na fase a seguir de revisão do produto e da equipe.

## **Retrospectiva**

Buscando a melhoria contínua no processo, é realizada uma reunião de retrospectiva ao fim da *Sprint*. Nesta reunião, os dados e acontecimentos que marcaram o ciclo são revisados e avaliados: desde as atividades diárias e *feedback* das reuniões, passando pelas ferramentas e linguagens de programação, chegando até aos problemas de relacionamento.

Derby e Larsen (2008, p. 4-14) propõe um método para facilitar o desenvolvimento da retrospectiva, que utiliza a seguinte estrutura:

1. Estabelecimento do estado atual, conjunto, sobre o resultado da *Sprint*, que pode ser obtido ao perguntar a cada membro por uma palavra que defina a sensação sobre a iteração finalizada;
2. Colheita dos dados, que podem ser eventos, métricas, histórias, tecnologias utilizadas, enfim, tudo que teve um impacto significativo a ponto de ser lembrado; juntamente com a importância que tal item teve na *Sprint*, para cada membro.
3. Análise do problema: neste momento é necessário questionar sobre os por quês do acontecimento de determinado evento / dado;
4. Decisão do que fazer: é a hora de listar os experimentos e melhorias (cerca de dois ou três itens) que podem ser efetuados na próxima *Sprint* e decidir quem se responsabilizará pelo cumprimento destes;
5. Fechamento da retrospectiva, com um agradecimento ao esforço empregado pelo

time durante a *Sprint*, salvando um registro do que foi discutido na *Sprint* e endereçando os itens acordados aos responsáveis.

### **Fechamento do Ciclo: a entrega do produto**

As fases e eventos citados anteriormente, de planejamento, execução, revisão e retrospectiva, formam o ciclo do Scrum e devem ser repetidas indefinidamente até se atingir as funcionalidades necessárias para compor o produto ou uma versão consistente (Deemer; Benefield, 2006).

Quando todas as funcionalidades requisitadas para uma versão do produto são concluídas, ocorre a adição da fase de *PostGame* sucedendo o ciclo normal do Scrum.

O *PostGame* é caracterizado como o esforço final para o lançamento do produto desenvolvido e inclui atividades de integração, testes de sistema, documentação para o usuário, criação de material para treinamento e marketing do produto (SCHWABER, 1996).

O conhecimento sobre os princípios do gerenciamento ágil e os papéis, fases, cerimoniais, artefatos e competências utilizados no Scrum, são essenciais para quem deseja aplicar esta metodologia em seu processo de desenvolvimento de projetos. A utilização de instruções como livros e artigos, ou atividades com maior interação, como debates e jogos, é uma das formas de se adquirir o conhecimento necessário para a prática do Scrum. Os conceitos sobre aprendizagem e jogos para o ensino do Scrum são abordados na seção a seguir.

## **2.2. Aprendizagem e jogos educacionais**

Nesta seção são abordados os conceitos sobre ensino e *design* instrucional, taxionomia de objetivos educacionais, estratégias instrucionais, aprendizagem experiencial e jogos educacionais.

### **2.2.1. *Design* instrucional**

“Aprendizagem é toda mudança relativamente permanente no potencial de comportamento, que resulta da experiência, mas não é causada por cansaço, maturação, drogas, lesões ou doença” (LEFRANÇOIS, 2008, p. 6). Experiência é a exposição (ou participação) a eventos que resultem em uma resposta do aluno.

Após o processo de ensino, o aluno pode não demonstrar que obteve tal experiência, até ser submetido a uma avaliação de desempenho, como uma prova escrita ou prática. “O fato de a maioria dessas mudanças (experiência) permanecer latente, evidenciando-se quando há a oportunidade de ação – num exame, por exemplo – não as faz menos reais” (LEFRANÇOIS, 2008, p. 6-7).

No modelo de *design* instrucional proposto por Dick, Carey L. e Carey J. (2009, p. 1), o ensino é considerado um sistema; o aluno, o professor, o material de instrução e o ambiente de aprendizado são os componentes deste sistema (que interagem entre si para atingir o objetivo); e os testes servem para avaliar se o aprendizado está acontecendo. Este modelo inclui conjuntos de procedimentos e técnicas que devem ser adotados pelo professor para analisar, projetar, desenvolver, executar e avaliar uma instrução (DICK; CAREY L.; CAREY J., 2009, p. 6-8), organizados nos passos a seguir:

1. **Identificar metas instrucionais:** o primeiro passo do modelo é a identificação de quais conhecimentos, habilidades e técnicas o aluno deve obter ao completar a instrução.
2. **Conduzir análise instrucional:** após a identificação das metas, é necessário determinar cada passo que será percorrido para que o aluno complete cada meta instrucional. A última atividade da análise instrucional é determinar quais habilidades, conhecimento e atitudes - comportamentos de entrada - são necessários para que os alunos possam iniciar a instrução.
3. **Analisar aprendizes e contextos:** é necessário efetuar a análise dos alunos, do contexto em que aprenderão as habilidades e do contexto em que as usarão.
4. **Formular objetivos de desempenho:** considerando a análise instrucional e os comportamentos de entrada, deve ser escrita uma especificação com as habilidades a serem aprendidas, com as condições em que estas habilidades devem ser utilizadas e os critérios para uma utilização bem sucedida.
5. **Desenvolver instrumentos de avaliação:** com base nas metas, é necessário

desenvolver avaliações que possam ser executadas paralelamente, com o intuito de medir a capacidade dos alunos em atingir os tipos de habilidades descritas para as metas instrucionais.

6. **Desenvolver estratégias instrucionais:** baseando-se nas informações dos passos anteriores, deve ser determinada a estratégia que será utilizada para atingir o objetivo final. A estratégia deve ser baseada nas teorias de aprendizagem, nas características do material a ser utilizado, no conteúdo a ser ensinado e nas características dos alunos que receberão a instrução.
7. **Desenvolver e selecionar materiais instrucionais:** utilizando a estratégia desenvolvida no passo anterior, devem-se criar guias de ensino, transparências, vídeos e outras formas de mídia, e páginas para o ensino a distância.
8. **Desenvolver avaliações formativas:** avaliações para coletar dados devem ser desenvolvidas com o objetivo de fornecer insumos para a melhoria da instrução. Estas avaliações podem ser individuais, em um pequeno grupo ou avaliações de campo, sendo que cada uma destas fornece diferentes tipos de informação que podem ser utilizados no processo de melhoria.
9. **Revisar instrução:** considerado como o primeiro passo do ciclo de melhoria da instrução, esta atividade compreende a sumarização dos dados obtidos nas avaliações formativas, a interpretação destes dados, a fim de identificar as dificuldades encontradas pelos alunos em atingir os objetivos, e a associação destas dificuldades em deficiências específicas da instrução.
10. **Projetar e conduzir avaliações sumativas:** neste passo é feita a avaliação da efetividade da instrução (do valor absoluto e/ou relativo desta) por um avaliador independente. Este procedimento ocorre apenas quando a instrução atinge os padrões definidos pelo designer instrucional, ou seja, após as sucessivas avaliações formativas, revisões e melhorias.

Caso o professor resolva reaproveitar uma instrução desenvolvida previamente, nem todos os passos do *desing* instrucional (figura 5) são necessários, como no desenvolvimento e seleção de materiais instrucionais ou até na identificação de metas instrucionais (caso o objetivo que se quer alcançar seja o mesmo da instrução original).

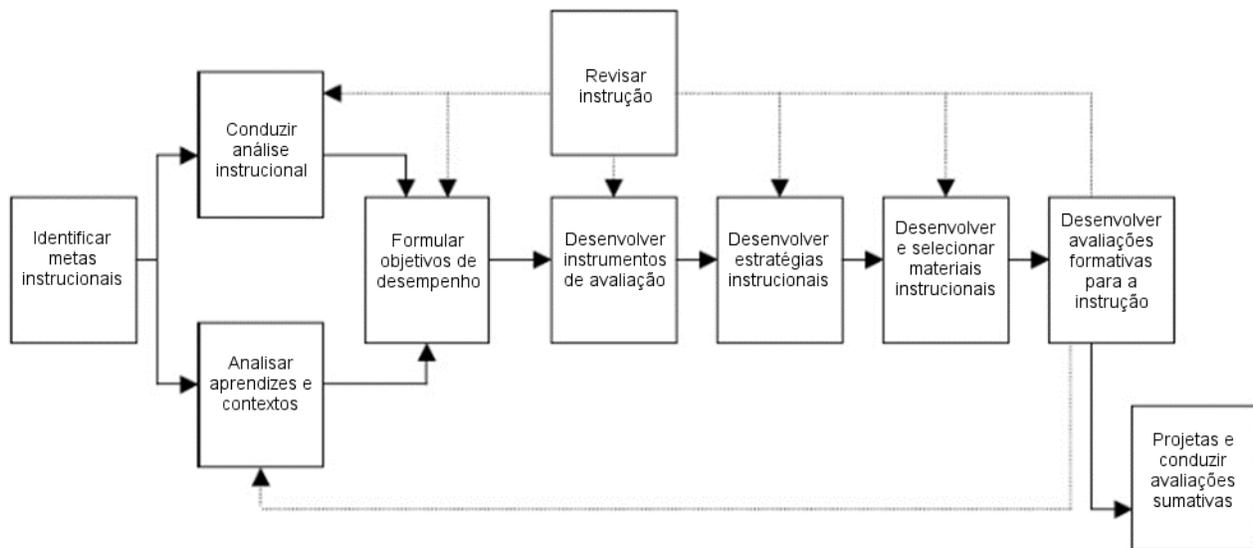


Figura 5 - Modelo de *design* instrucional (DICK; CAREY L.; CAREY J., 2009, p. 1, tradução nossa)

A identificação ou formulação dos objetivos educacionais assume uma importância elevada no processo do *design* instrucional, por tratar dos resultados desejados e previstos para a ação educativa, como pode ser visto no tópico a seguir.

O desenvolvimento de estratégias instrucionais e a utilização de métodos condizentes com estas estratégias são tarefas não menos importantes, que devem ser desempenhadas pelo instrutor, por este motivo, estes conceitos são abordados no tópico subsequente ao tópico dos objetivos educacionais.

### 2.2.2. Objetivos educacionais

“A formulação de objetivos de ensino consiste na definição de todos os comportamentos que podem modificar-se como resultado da aprendizagem” (HAIDT, 2001, p. 113).

A falta de objetivos claros e precisos, ou a falta de especificação e explicitação destes, faz com que o ensino perca a orientação e a direção, aumentando a possibilidade de falha do professor e seu projeto (HAIDT, 2001, p. 112,113).

Com a finalidade de apoiar professores e especialistas envolvidos em problemas de

currículo e avaliação, Benjamin Bloom e uma equipe de pesquisadores propôs uma taxionomia – classificação – dos objetivos educacionais. Esta taxionomia facilita a formulação de objetivos, o preparo de programas de avaliação e o intercâmbio de informações sobre os desenvolvimentos curriculares e os planos de avaliação (BLOOM et al, 1979, p. 1,2).

A taxionomia de Bloom et al (1979, p. 6,7) é dividida nos domínios listados a seguir:

- Domínio cognitivo: que inclui os objetivos vinculados à memória, ao reconhecimento e ao desenvolvimento de capacidades e habilidades intelectuais.
- Domínio afetivo: que inclui objetivos que descrevem mudanças de interesse, atitudes e valores, e o desenvolvimento de apreciações e ajustamento adequado.
- Domínio psicomotor: que inclui objetivos nas áreas de habilidades manipulativas ou motoras.

Segundo Bloom et al (1979, p. 6), o domínio cognitivo “é onde se encontra as mais claras definições de objetivos expressas em termos de comportamento do aluno”, enquanto que nos outros domínios, “os objetivos não apresentam uma formulação precisa e (...) os professores parecem não estar muito esclarecidos sobre as aprendizagens apropriadas à consecução dos mesmos”. As classes do domínio cognitivo são descritas e exemplificadas a seguir (BLOOM et al, 1979, p. 171-179):

1. **Conhecimento:** evocação de elementos de informação específicos e universais, de métodos e processos, de padrões, estruturas, composições e abstrações, por exemplo: letras que compõe uma linguagem, datas, acontecimentos, lugares, metodologias, normas, convenções, princípios e teorias;
2. **Compreensão:** tipo de entendimento que possibilita ao indivíduo fazer uso da ideia que está sendo comunicada; através da translação de uma forma de comunicação em outra, como em metáforas, simbolismos e ironias; através da interpretação, para captar a ideia central de uma obra ou filme, por exemplo; e através da extrapolação de direções ou tendências, determinando implicações, consequências, efeitos, etc.;
3. **Aplicação:** utilização de ideias, regras, métodos, princípios e teorias em

situações particulares e concretas; como no caso da aplicação de conceitos científicos em um trabalho, que foram aprendidos em outro anterior, e da pilotagem inaugural de um avião, em que o piloto utiliza os conhecimentos adquiridos em um simulador e em voos anteriores;

4. **Análise:** desdobramento dos elementos de uma comunicação de modo que a organização ou a relação entre ideias é tornada clara e explícita; como no caso do reconhecimento de suposições, na distinção entre fatos e hipóteses e na capacidade de reconhecer formas e padrões em obras literárias e artísticas;
5. **Síntese:** combinação e alteração da disposição de elementos e partes da comunicação para que constituam um padrão ou estrutura que antes não estava evidente. As descobertas e generalizações matemáticas e a criação eficaz do relato sobre uma experiência pessoal são exemplos de utilização da síntese em informações; e
6. **Avaliação:** utilização de julgamentos quantitativos e qualitativos acerca da medida em que material e métodos satisfazem os critérios. A avaliação é utilizada, por exemplo, nos casos de verificação de desempenho escolar e nos testes para candidatos em um processo de recrutamento.

Os objetivos de aprendizagem do domínio afetivo, “ênfatisam uma tonalidade de sentimentos, uma emoção ou um grau de aceitação ou de rejeição” (BLOOM; KRATHWOHL; MASIA, 1977, p. 5). Os interesses, atitudes e valores que um indivíduo adquire ou modifica durante uma atividade são exemplos de objetivos deste domínio. A lista de objetivos da taxionomia proposta por Bloom, Krathwohl e Masia (1977, p. 179-188) é apresentada em seguida:

- **recepção:** através da percepção de situações, fenômenos, objetos ou estádios de um acontecimento; da disposição para receber um estímulo e a tolerá-lo, sem julgamentos; e da atenção seletiva, com diferenciação de aspectos e controle da atenção sobre um estímulo (a atenção à um locutor, de forma ativa, a tolerância à padrões culturais diferentes, e a perspicácia em relação a valores humanos, são exemplos deste objetivo);
- **resposta:** através da submissão na resposta, com consentimento mas sem plena aceitação; da disposição para responder, voluntariamente; e da

satisfação na resposta, na forma emocional, com gosto ou prazer (e.g., desenvolvimento de interesses, familiarização em assuntos culturais através da leitura e discussão voluntárias);

- **valorização:** através da aceitação de que um fenômeno, comportamento ou objeto tem valor; da preferência por um valor, compromisso para buscá-lo e querê-lo; e do cometimento, lealdade e convicção em uma crença, posição, grupo ou causa (a fé no poder da razão e nos métodos de experimentação e de discussão são exemplos deste objetivo);
- **organização:** através da conceituação de um valor, na tentativa de identificar as características integrais do valor ou da crença; e da organização de um sistema de valores, para que valores distintos possam ser relacionados e comparados, criando um novo valor ou complexo de valores; e
- **internalização de valores:** através da direção generalizada, expressa na orientação em relação à fenômenos ou na predisposição para agir de certa maneira; e da caracterização, do indivíduo, através do desenvolvimento de comportamentos baseados em princípios éticos ou ideais democráticos ou de uma filosofia de vida, por exemplo.

A classificação de objetivos educacionais no domínio psicomotor é utilizada para indicar um conjunto de habilidades e atitudes trabalhados em uma instrução. Os objetivos do domínio psicomotor são apresentados a seguir (BRUM, 1977, p. 25-40):

- **percepção:** ato de tomar conhecimento de objetos, qualidades ou relações através das atividades motoras;
- **preparação:** ajustamento preparatório, ou prontidão, para um tipo de ação ou experiência;
- **resposta orientada:** ação comportamental evidente em função da necessidade de uma resposta, ou tomada de decisão;
- **resposta mecânica:** comportamento natural, produto da aquisição de níveis consideráveis de confiança e habilidade, resgatado de um repertório de possíveis respostas à estímulos;
- **resposta completa evidente:** exibição de um ato motor, com padrões de movimentos complexos, executada com gasto mínimo de tempo e de energia; e

- **reorganização:** modificação consciente ou inconsciente dos automatismos ou hábitos em relação às mudanças que ocorrem no ambiente.

Os objetivos do domínio podem parecer específicos para o trabalho manual, distante do ensino de gerenciamento ágil de projetos com Scrum através de jogos. Entretanto, é neste último elemento, que proporciona uma aprendizagem através da experiência, que as habilidades e os valores são trabalhados através de estímulos motores e afetivos, para que, a posteriori, sejam internalizados, fazendo parte do conhecimento. A reunião de acompanhamento diário do Scrum, por exemplo, acontece normalmente com os participantes de pé (um de frente para o outro), para que passem uma indicação corporal de que estão dispostos a falar de forma clara e a ouvir ativamente.

A extensão natural do processo de formulação ou identificação de objetivos para a construção de uma atividade para o ensino, é a avaliação da aprendizagem. Para viabilizar este processo, é utilizado um modelo de avaliação composto pelos cinco níveis listados a seguir (KIRKPATRICK, 1998, p. 20-26):

- reação: uma medida de satisfação do aluno, de como este reagiu à atividade aplicada; apesar de uma reação positiva não ter relação direta com o aprendizado, é importante buscá-la, porque uma reação negativa certamente reduz a possibilidade deste processo acontecer;
- aprendizado: pode ser definido como a aquisição ou mudança de atitudes, conhecimento e habilidades como resposta à participação na atividade;
- comportamento: uma mudança no comportamento após a participação na atividade, percebida quando o aluno apresenta um desejo de mudança, sabe o que fazer e como fazer, se encontra no ambiente favorável e recebe uma recompensa pela mudança; e
- resultado: os resultados obtidos após a participação na atividade, por exemplo, uma diminuição no tempo de desenvolvimento, a melhoria na postura corporal diária e o aumento na atenção das instruções do líder do time.

Os objetivos e a avaliação da aprendizagem, como foram descritos, compõe a base das atividades instrucionais, a finalidade pelo qual devem ser executadas. As estratégias para aplicação destas instruções e as formas de absorção do conhecimento por parte dos

alunos são abordadas nos tópicos a seguir.

### 2.2.3. Estratégias instrucionais

A forma de se utilizar métodos e habilidades para atingir os objetivos de aprendizagem é conhecida como estratégia instrucional (SASKATCHEWAN EDUCATION, 1991, p. 23,24).

“Os métodos são utilizados pelos professores para criar ambientes de aprendizado e para especificar a natureza da atividade na qual o aprendiz e o professor estarão envolvidos”. Um grupo de aprendizado cooperativo, uma investigação e formalização de hipóteses, e o interrogatório didático (o que, quando, onde e como) são exemplos de métodos de instrução.

As habilidades do instrutor são necessárias para a estruturação apropriada das experiências de aprendizado dos alunos; e incluem técnicas, como questionamento, discussão e demonstração; e ações, como planejamento, estruturação e gerenciamento (SASKATCHEWAN EDUCATION, 1991, p. 23).

Cinco categorias de estratégias instrucionais podem ser destacadas, segundo Saskatchewan Education (1991, p. 16-19):

- **Instrução Direta:** é dirigida pelo professor e tem como ponto forte o fornecimento de informações e o desenvolvimento de habilidades, passo-a-passo. Alguns métodos utilizados com esta estratégia são: palestra, interrogatório didático, prática e exercício, e demonstração.
- **Instrução Indireta:** é centrada no aluno e busca o envolvimento deste através da observação, investigação e formulação de hipóteses. Exemplos de métodos incluem discussão reflexiva, formação e obtenção de conceitos, procedimento de preencher lacunas textuais, e resolução de problemas.
- **Instrução Interativa:** é centrada na discussão e compartilhamento de informações entre os participantes e proporciona a oportunidade de reagir ao conhecimento e experiência de seus pares; desenvolvendo habilidades sociais e argumentos racionais. Exemplos de métodos incluem discussão de classe, discussão ou projetos em grupos, e desenvolvimento de tarefas e pares ou trios.

- **Aprendizagem Experiencial:** é centrada no aluno e orientada pelas atividades, sendo que a ênfase desta estratégia está no processo de aprendizado e não no produto. Ocorre quando o aluno gera conhecimento a partir da análise crítica efetuada em cima de uma experiência prática (atividade). Alguns métodos utilizados na aprendizagem experiencial são: simulação, dramatização, jogo, condução de experimentos e construção de modelos.
- **Estudo independente:** é centrado no aluno e ocorre com a aplicação de estudos independentes com os alunos sob a supervisão do professor. Esta estratégia busca a transformação dos alunos em aprendizes independentes, tornando-os cidadãos mais autossuficientes e responsáveis. Exemplos de métodos incluem relatório, questionário, tarefa para casa, projeto de pesquisa e contrato de aprendizado.

As estratégias citadas e os métodos utilizados para cada estratégia podem ser observados na figura 6.

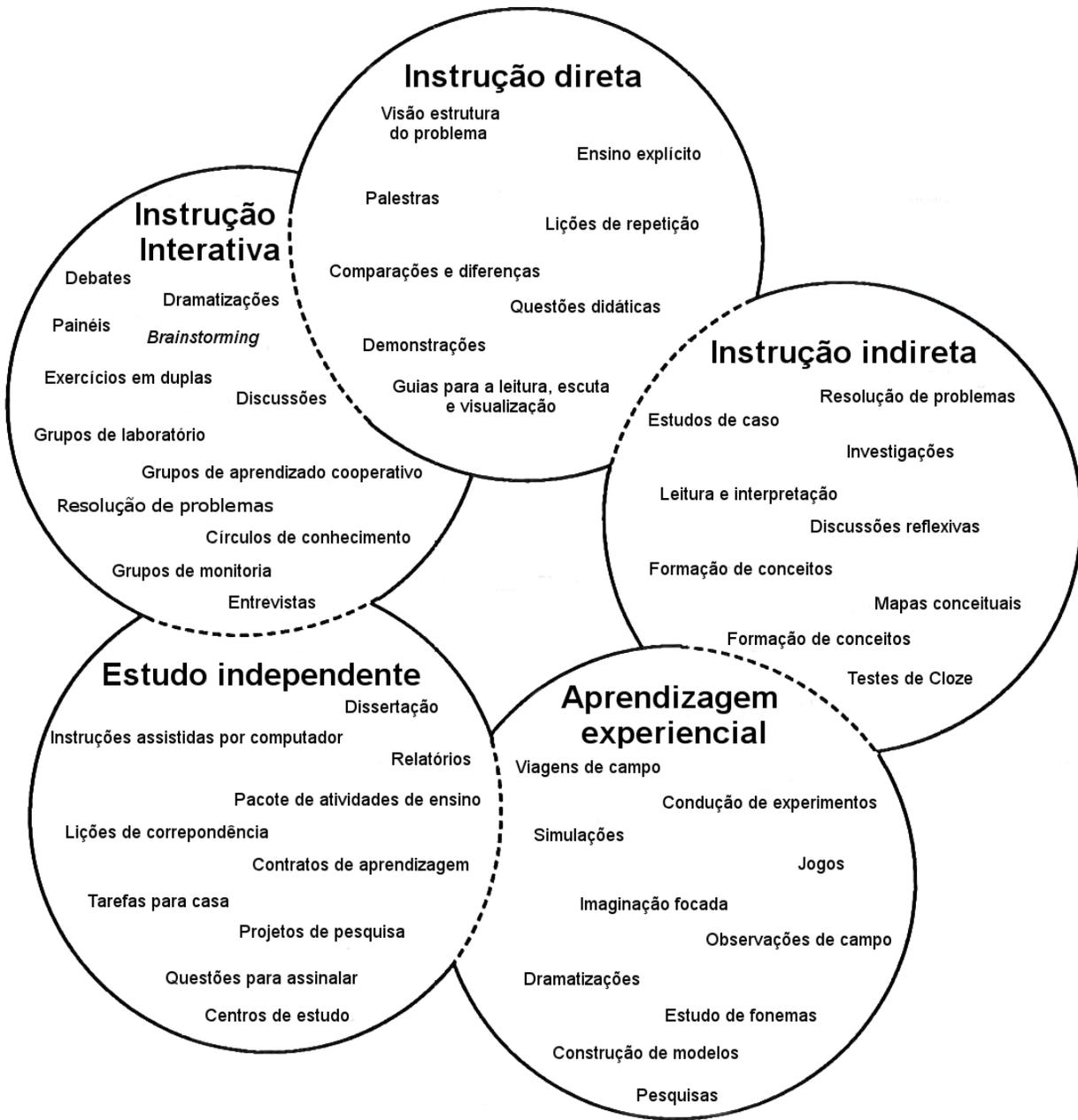


Figura 6 - Estratégias e métodos instrucionais (adaptado de SASKATCHEWAN EDUCATION, 2001, p. 20).

Estas estratégias utilizam métodos para ensinar alunos com idades, objetivos e capacidades de aprendizado distintos. Uma das estratégias que mais se destaca para o ensino dos conhecimentos, habilidades e atitudes necessárias para a utilização do Scrum é a aprendizagem experiencial, através da aplicação de jogos.

### 2.2.3.1. Aprendizagem Experiencial

A aprendizagem experiencial proporciona lições tiradas de forma espontânea através da reflexão, utilizando atividades, jogos e simulações (PIAGET, 1985). Kolb (2008, p. 5) define aprendizagem experiencial como sendo a aquisição de conhecimento através das experiências de compreensão e de transformação.

A Experiência Concreta e a Conceptualização Abstrata são modos de compreensão e, a Observação Reflexiva e a Experimentação Ativa são modos de transformação. Estes quatro modos de compreensão e transformação são as fases do aprendizado do aluno, descritos a seguir (KOLB, 1984, p. 30):

1. Inicialmente, o aluno deve ser exposto a uma experiência nova que o envolva totalmente, de forma aberta, sem preconceitos (**Experiência Concreta**). A combinação do problema com os conhecimentos e processos já existentes, aprendidos anteriormente, é o que torna a experiência inédita.
2. Ele deve ser capaz de observar esta experiência e refletir, por várias perspectivas (**Observação Reflexiva**). A identificação de elementos, construção de associações e determinação de características, são exemplos de aplicação da observação reflexiva em uma experiência.
3. Ele deve ser capaz de criar conceitos que transformem suas observações em teorias lógicas (**Conceituação Abstrata**). A generalização de regras e princípios e a comparação da experiência, dos elementos e características, com realidades semelhantes, são exemplos da utilização da conceituação abstrata.
4. No último passo do ciclo, o aluno deve ser capaz de usar estas teorias para tomar decisões e resolver novos problemas (**Experiência Ativa**). Ações para a aplicação prática, seja ela no âmbito pessoal ou interpessoal, são esperadas de quem utiliza a experiência ativa.

A participação de um aluno no jogo *Scrum Game*<sup>3</sup> (WAKE; COHN, 2007), por

---

<sup>3</sup> O *Scrum Game* é um jogo colaborativo, onde os participantes têm o objetivo de conduzir uma Reunião de Revisão após a finalização de uma *Sprint*. Para finalizar a *Sprint* é necessário que o time realize uma carga de trabalho determinada conforme a escolha das cartas e, a sorte nos dados e nas casas do tabuleiro. O

exemplo, faz com que ele se depare com uma situação inédita, mesmo que os participantes, o tabuleiro e as peças não o sejam. O processo de observação e reflexão possibilita a percepção sobre as jogadas, as cartas sorteadas, a pontuação obtida nos dados, e a vitória ou derrota. Com base nestas informações, o aluno pode generalizar, ou buscar padrões, como, por exemplo, avaliar se o outro participante está com sorte nos dados, com motivação para jogar, ou como foi a reação dos outros participantes após a vitória, ou a derrota. Concluindo o aprendizado nesta experiência, o aluno pode desenvolver uma nova estratégia ou tática, para ser aplicada diretamente em uma nova partida ou, extrair uma lição, aprendizado, que possa ser aplicado em um universo de atividades mais amplo.

Outros jogos para o aprendizado do Scrum, podem ter uma dinâmica diferente e serem baseados mais no conhecimento do que na sorte, mas compartilham a mesma plataforma de aprendizado, possibilitando que o aluno experimente, reflita, pense e aja, passando por um processo recursivo a cada nova experiência vivida (KOLB, 2008, p. 5).

Em detrimento de outros métodos de ensino como observação da prática e construção de modelos, a aplicação de jogos é escolhida por ser uma atividade lúdica, natural do ser humano, onde joga-se simplesmente pelo prazer de jogar (HAIDT, 2001, p. 175). Como o ensino do Scrum não visa, apenas, a aplicação de testes de conhecimento, mas a utilização de conceitos aprendidos na prática, é vantajoso que se utilize um método que seja apreciado pelo aluno.

#### **2.2.4. Jogos**

“O jogo é uma atividade executada com certos limites de tempo e espaço, com regras aceitas em comum acordo, mas obrigatórias, com o propósito contido; acompanhadas por sensações de tensão, prazer e consciência diferentes das encontradas no dia-a-dia” (PAULVALÉRY, 1943 apud CAILLOIS, 2006, p. 149, tradução nossa).

Segundo Caillois (2006, p.128), os jogos, de forma geral, possuem as seguintes características:

- Livre: quando não há uma obrigação no ato de jogar, o que mantém a atratividade e o prazer do jogo;

---

período máximo para duração da *Sprint* é de 10 dias, se o time não realizar o trabalho necessário ou não alcançar a “casa final” do tabuleiro, o jogo é dado como perdido.

- Limitado: por um conjunto de limites de espaço e tempo, definidos previamente;
- Incerto: o curso ou o resultado não podem ser determinados com precisão;
- Improdutivo: não gera produtos, riqueza, nem elementos de qualquer tipo, com exceção do conhecimento trocado entre os jogadores e extraído da execução da atividade;
- Governado por regras: com convenções que suspendem as leis do mundo real, estabelecendo uma nova legislação para o espaço de tempo do jogo.
- Simulado: pode estabelecer uma realidade secundária ou uma livre “irrealidade” (como oposição ao mundo real).

As características citadas anteriormente são puramente formais e não abordam, nem classificam os jogos de acordo com seu conteúdo. Uma classificação neste domínio deve abranger uma infinita variedade de jogos, como jogos de cartas, jogos de habilidade, jogos para um e para vários participantes, jogos para o ar livre, etc. Em consideração a estes desafios, propõe-se a divisão de jogos nas categorias a seguir.

Os jogos em que uma igualdade de chances é criada artificialmente para que os jogadores se confrontem em igual condição, são chamados de **jogos de competição** (*agôn*). Quando esta igualdade não pode ser obtida naturalmente, uma determinada vantagem (*handicap*) é fornecida para um dos jogadores, na tentativa de tornar a disputa o mais igual possível. Independentemente da dinâmica do jogo, se está associado à características físicas (esportes) ou mentais (xadrez, damas), dar-se-á a sensação de que o vencedor parece melhor do que o perdedor (CAILLOIS; 2006, p. 131-133).

Em oposição à utilização do trabalho, da experiência, da paciência e de outras qualificações necessárias a um jogo de competição, existem os jogos em que o jogador não possui controle sobre a jogada, sobre o fluxo do jogo ou sobre o término, chamados de **jogos de sorte** (*alea*). Ao participar de uma partida de roleta ou de caça-níquel, o jogador não pode utilizar nenhuma habilidade ou conhecimento para determinar com precisão o destino do jogo: fracasso completo ou favorecimento absoluto (CAILLOIS; 2006, p. 131-135).

Quando o principal componente do jogo não é mais a competição, nem a sorte, mas a ilusão, onde as “jogadas” pressupõem a aceitação, mesmo que temporária, de um universo imaginário, está sendo feita uma referência a **jogos de simulação** (*mimicry*). Neste tipo de jogo o papel do jogador é acreditar ou fazer com que outros acreditem que ele não é ele

mesmo, abandonando, disfarçando ou esquecendo sua própria personalidade. Fazer caricaturas, mímicas e imitações, são exemplos nos quais o jogador altera seu comportamento, fala ou aparência para que se parece com outra pessoa ou coisa, mesmo que por um curto espaço de tempo (CAILLOIS; 2006, p. 131-137).

A negação ou divergência da realidade não é característica exclusiva dos jogos de simulação, ela pode ser encontrada na tentativa momentânea de destruir a estabilidade de percepção e infringir um tipo de pânico em uma mente outrora lúcida. Esta rendição à espasmos, apreensão ou choque, que destrói a realidade de forma brusca, é encontrada nos **jogos de vertigem** (ilinx), como *Bungee Jumping*, montanha russa e roda gigante (CAILLOIS; 2006, p. 138-140).

Existem jogos que contém características de mais de uma categoria, como o *poker* e a corrida de cavalos, que, apesar da disposição igualitária dos elementos, pouco podem fazer para mudar o resultado do jogo. Fazendo uma referência mais próxima ao presente trabalho, o jogador do *Scrum Game* pode conhecer as regras do jogo, o tabuleiro e pode até calcular a chance que o time tem de vencer em um determinado momento, mas nada pode garantir qual será o valor tirado no dado ou a próxima carta retirada da mesa.

Esta diferença, entre jogar por diversão ou jogar guiado por análise e estratégia é chamada de grau de disciplina. Para que passe a jogar de **forma regrada**, burocrática, utilizando os valores intelectuais e morais, as habilidades físicas e psicológicas, e os conhecimentos técnicos e analíticos, o jogador deve, primeiramente, adquirir o gosto pelo jogo, aproveitando o de **forma espontânea**, livre, com o intuito de estimular a distração, a fantasia e o prazer (CAILLOIS, 2006, p. 141).

No caso dos jogos educativos (para o ensino do Scrum), não se pode esperar que o aluno jogue apenas por prazer, por diversão, mas, também, que o faça pelo aprendizado. Para estes jogos espera-se um alto grau de disciplina, fazendo com que o aluno comece a jogar de forma espontânea e, rapidamente, passe a utilizar o seu conhecimento e habilidades para vencer o jogo.

Como o *Scrum* é um metodologia para gerenciamento de projetos, os alunos podem aprender, por exemplo, com uma competição direta entre participantes; com uma avaliação de quando desprezar garantias em função da sorte; e com simulações de desenvolvimento de um produto ou gestão de processo; mas, dificilmente, será encontrado um jogo em que o participante aprenda um conceito, princípio ou prática útil ao *Scrum*, através do medo ou

vertigem.

Considerando que jogo educativo é uma extensão da teoria geral dos jogos, o conceito de que “um jogo pode ser chamado de atividade ‘não séria’” (HUIZINGA, 1950 apud CAILLOIS, 2006, p. 123), foi modificado segundo Ritterfeld et al (2009, p.3), passando a considerar um jogo que envolve tanto a diversão como a educação, como sendo um jogo sério.

## Jogos sérios

São considerados jogos educacionais, ou sérios, os que não possuem como seu propósito primário o entretenimento, prazer ou diversão, mas sim a educação (FELICIA, 2011, p. 121).

Os jogos sérios podem ser classificados de acordo com a forma com que são jogados, com o propósito da atividade e com o público alvo, segundo o modelo de Objetivo, Propósito e Escopo, ou GPS (FELICIA, 2011, p. 126).

Segundo Portugal (2006 apud FELICIA, 2011, p. 126), um jogo depende de regras, estipuladas antes do início do jogo, de ações, executadas pelos jogadores, e de configurações de espaço físico, tempo e dramatização. Na definição destas regras, devem constar os objetivos que devem ser buscados na execução do jogo (como, por exemplo, “executar todas as tarefas da *Sprint* antes do tempo”) e, tratando-se de jogos educacionais, espera-se que junto com aqueles, estejam especificados os objetivos de aprendizagem.

Dependendo das características de tempo, custo, quantidade de participantes e área disponível para a execução de um jogo, este pode ser considerado como inadequado para um determinado público alvo. Para o presente trabalho são considerados os públicos descritos na tabela 2.

Classificação	Descrição
Ensino superior (faculdades e cursos técnicos)	Ênfase no envolvimento do aluno, na diversão ( <i>ludus</i> ), para que desperte o gosto pelo aprendizado e que as lições do jogo sejam tiradas de forma natural. Um jogo que contenha regras muito rígidas

	pode se tornar monótono, tirando o interesse do aluno
<b>Ensino corporativo</b> (cursos e treinamentos)	Ênfase na obtenção de resultados, junção de teoria e prática. Para este público evitam-se jogos abertos, de extensa duração e com ênfase no divertimento.

Tabela 2 - Classificação de jogos por público alvo (FELICIA, 2011)

Como complemento da classificação GPS, é importante destacar que a escolha do jogo pode ser auxiliada pela classificação de acordo com a quantidade de jogadores necessários para iniciar a partida e a duração do jogo (ABRIL, 1978). Após uma análise sobre alguns jogos para o ensino do Scrum, observou-se a necessidade de modificar os valores referência da quantidade de jogadores mínima para o início de uma partida e especificar a duração em uma medida real.

Segundo Peng e Hsieh (2012, p. 2102), um jogo que possibilite o desenvolvimento de uma experiência de conflito pode auxiliar no desenvolvimento de tarefas cognitivas. Por outro lado, ao jogar com o auxílio (cooperação) de um amigo ou colega, haverá um ambiente de maior afinidade entre os jogadores, diminuindo a possibilidade de descontentamento e de conflito emocional, e, por conseguinte, proporcionando um melhor desempenho. Este ambiente de segurança se mantém em jogos que misturam a experiência de conflito e cooperação, nos quais o participante encontra desafios dos dois tipos de relação. Os jogos individuais, que não possibilitam a relação entre dois jogadores durante à execução do jogo (a introdução e finalização pode ser em conjunto), são considerados jogos sem interação entre participantes.

Como estes jogos serão executados em um ambiente de estudo ou de trabalho, eles precisam ser classificados quanto a necessidade de espaço. Os valores de referência para esta categoria, assim como os valores para as categorias mencionadas anteriormente, podem ser encontrados na tabela 3.

Quantidade de jogadores	Um jogador
	Dois jogadores
	Até quatro jogadores
	Até seis jogadores

	Até oito jogadores
	Mais de oito jogadores
Tipo de relação	Competitiva
	Colaborativa
	Cooperativa e Competitiva
	Sem interação entre participantes
Duração	Até trinta minutos
	Até uma hora
	Até duas horas
	Até quatro horas
	Até oito horas
	Mais de oito horas
Espaço por grupo	Pequeno (até 4m <sup>2</sup> )
	Médio (até 30m <sup>2</sup> )
	Grande (acima de 30m <sup>2</sup> )

Tabela 3 – Características de quantidade e relação entre jogadores, da duração e do espaço necessário para o jogo.

As informações de categoria geral (CAILLOIS, 2006), de público alvo (FELICIA, 2011), de propósito (BLOOM et al, 1979), de restrições de tempo e participantes (ABRIL, 1978) e de espaço físico (proposta pelo presente trabalho), são os grupos utilizados para classificar as informações de identificação e execução dos jogos (tabela 4).

	<b><i>Scrum Game</i></b>	<b><i>Scrum from Hell</i></b>
Categoria Geral	Sorte	Simulação
Público Alvo	Ensino superior e corporativo	Ensino superior e corporativo
Propósito	Aprender os conceitos sobre o fluxo da <i>Sprint</i> , e a importância de terminar o trabalho antes do tempo acordado.	Aprender sobre a importância da reunião de acompanhamento diário, e dos papéis do Scrum Master e do Time.
Nº de Participantes	Até oito jogadores	Até oito jogadores
Relação entre os	Colaborativa	Competitiva

participantes		
Duração	Até trinta minutos	Até trinta minutos
Espaço Físico	Pequeno	Pequeno

Tabela 4 – Exemplo da classificação de jogos

Uma classificação que evidencie os pontos relevantes na identificação de um jogo educativo é essencial para a busca efetiva por jogos que se adequam ao objetivo educacional e a fase de desenvolvimento de um curso ou disciplina. Supervisionar a aplicação de um jogo que aborde princípios da reunião de retrospectiva, sem antes ter comentado sobre os conceitos e funcionamento da *Sprint*, pode resultar em um aprendizado fraco ou nulo por parte dos jogadores.

Para possibilitar a busca e armazenamento destes jogos, é necessário distribuí-los em um formato digital que favoreça o compartilhamento em aplicações disponibilizadas na *Internet* ou em uma *Intranet*. Este formato e as aplicações destinadas ao compartilhamento dos jogos são abordados a seguir.

## 2.3. Objetos de aprendizagem e metadados

Nesta seção são apresentados os conceitos sobre recursos que podem ser utilizados para o ensino, os objetos de aprendizagem, e sobre os dados utilizados para identificar as informações contidas nestes objetos, os metadados.

### 2.3.1. Objetos de Aprendizagem

“Objeto de aprendizagem é definido como qualquer entidade digital ou não-digital, que possa ser utilizada para aprendizado, educação ou treinamento” (IEEE, 2002, p. 5, tradução nossa). Exercícios, livros, cursos, programas de estudo e pessoas, podem ser considerados objetos de aprendizagem, de acordo com esta definição (KOPER, 2003).

Com o intuito de criar uma semântica mais precisa, aumentando a aplicabilidade prática e científica, Koper (2003, p. 47) considera um objeto de aprendizagem como sendo

“qualquer recurso digital, reproduzível e endereçável, utilizado para a aplicação ou para o suporte de atividades de ensino, disponibilizado para o uso de terceiros”. Esta definição exclui materiais não digitais, exemplares únicos, não-reproduzíveis, e recursos não endereçados (que não estão conectados a uma URL ou a metadados). Pessoas, serviços, atividades e cursos - agregado de objetos de aprendizagem e atividades - são excluídos pela mesma definição (KOPER, 2003, p. 47).

Wiley (2000, p.2) aponta que, diferentemente de um livro e de uma palestra em DVD, que podem estar em apenas um lugar, os objetos de aprendizagem podem ser utilizados por inúmeras pessoas ao mesmo tempo. Este conceito de utilização é definido como a disponibilidade dos objetos de aprendizagem para a utilização por outras pessoas simultaneamente. As condições para que isso seja possível são listadas a seguir (KOPER, 2003, p. 48,50):

- O método ou estratégia de ensino, o contexto e a mídia de distribuição, devem ser abstraídos do objeto;
- Os objetos devem ser construídos em um formato pequeno, atômico, para que possam ser agregados em unidades de ensino maiores;
- Os objetos devem ser independentes.

A utilização de um objeto em um contexto diferente do que foi inicialmente aplicado é chamada de reutilização. Esta forma de utilização é facilitada quando um objeto pode ser pesquisado e acessado em um repositório, através da inclusão de informações – metadados - sobre este objeto (KOPER, 2003, p. 48).

O reuso de um objeto pode ser feito pelo próprio criador, aplicando-o a um contexto diferente para o qual foi originalmente criado; por uma pessoa da mesma comunidade ou organização; ou por uma pessoa externa à comunidade (KOPER, 2003, p. 48).

A identificação do contexto, do criador, do formato e de todas as outras características do objeto de aprendizagem é realizada através da especificação dos metadados e posterior preenchimento das informações relativas à estes dados. O conceito de metadados e o modelo de referência, utilizado no presente trabalho, são apresentados a seguir.

### 2.3.2. Metadados

Com o objetivo de facilitar o compartilhamento e o reuso de objetos de aprendizagem entre diferentes repositórios, é recomendado que este material esteja associado a um padrão comum de metadados (ROY; SARKAR; GHOSE, 2010, p. 103).

O termo metadados se refere à informação sobre informação ou dados sobre dados, segundo BRAND et al (2003). Essa definição não é precisa, de forma que é impossível identificar metadados apenas olhando para eles. Metadados são dados utilizados para descrever outro conjunto de dados, dessa forma o uso destes o tornam metadados. O contexto de uso é necessário para a identificação correta de dado e metadado (BARGMEYER; GILLMAN, 2002). As principais vantagens de referenciar documentos com metadados são listadas a seguir:

- busca, aquisição e utilização de objetos de aprendizagem pelo estudante;
- reusabilidade de objetos de aprendizagem, permitindo a utilização em diferentes contextos instrucionais; e
- interoperabilidade dos objetos de aprendizagem permitindo o compartilhamento entre sistemas de aprendizado desenvolvidos em qualquer tecnologia.

Um dos modelos de metadados com as vantagens citadas, que se tornou referência para a comunidade, é o *Draft Standard for Learning Object Metadata* (IEEE, 1484.12.1, 2002), ou simplesmente *LOM*, que descreve objetos de aprendizagem em elementos de dados e categorias. O *LOM* pode ser visualizado como um modelo em árvore, onde as categorias e subcategorias formam os galhos, e os elementos são como folhas, que carregam os valores, tipos e cardinalidades dos metadados. Esta representação ilustra uma restrição do modelo, de que apenas os elementos (metadados “folha”) são utilizados para referenciar um objeto (ou um metadado). As nove categorias que compõe o modelo são descritas a seguir, associadas à tabelas que contém os demais elementos da categoria:

1. Geral: agrupa as informações gerais que descrevem o objeto de aprendizagem como um todo (tabela 5);
2. Ciclo de vida: agrupa os elementos relacionados à história e ao estado atual do objeto de aprendizagem, associados aqueles que tenham o afetado (tabela 6);

3. Meta-metadado: agrupa informações sobre a instância do próprio metadado (tabela 7);
4. Técnico: agrupa os requisitos técnicos e as características técnicas do objeto de aprendizagem (tabela 8);
5. Educacional: agrupa as características educacionais e pedagógicas do objeto de aprendizagem (tabela 9);
6. Direitos Autorais: agrupa os direitos de propriedade intelectual e as condições de uso para o objeto de aprendizagem (tabela 10);
7. Relacionamento: agrupa os relacionamentos entre objetos de aprendizagem (tabela 11);
8. Anotação: fornece comentários sobre o uso educacional dos objetos de aprendizagem e inclui informações sobre quem criou o comentário (tabela 12); e
9. Classificação: descreve o relacionamento do objeto de aprendizagem com um sistema de classificação (tabela 13).

1. Metadados gerais				
Número	Nome	Explicação	Tipo e valores possíveis	Exemplo
1.1	Identificador	Valor global único que identifica o objeto.	--	--
1.1.1	Catálogo	Catálogo global que armazena a identificação do objeto.	Texto simples	"ISBN", "ARIADNE" ou "GQS-UFSC"
1.1.2	Registro	Valor que identifica o objeto dentro de um esquema de catalogação.	Texto simples	"2-7342-0318" ou "http://www.gqs.ufsc.br/documentos/123"
1.2	Título	Nome do objeto	Texto simples	"Scrum From Hell" ou "The Scrum Game"
1.3	Língua	Idiomas utilizados pelo objeto.	Texto simples	"pt-BR", "em-US-philadelphia" e "fr-CA".
1.4	Descrição	Descrição textual do conteúdo do objeto	Texto simples	"Neste jogo os participantes desempenham papéis de apoio e de sabotagem em uma Reunião diária de acompanhamento do Scrum".
1.5	Palavras-chave	Palavras-chave ou frase que descreve o tópico abordado no objeto.	Texto simples	"Scrum", "Reunião de acompanhamento diário", "Daily Meeting".
1.6	Cobertura	Época, cultura, geografia ou região na qual o objeto de aprendizagem se aplica.	Texto simples	"2013, sul do Brasil".
1.7	Estrutura	Estrutura	Seleção de um dos valores a	• "atômica" livro ou jogo;

		organizacional do objeto	seguir: <ul style="list-style-type: none"> <li>• atômica: objeto indivisível;</li> <li>• coleção: conjunto de objetos sem inter-relação;</li> <li>• interligação: conjunto de objetos interligados;</li> <li>• hierarquia: conjunto de objetos com o um relacionamento hierárquico; ou</li> <li>• linear: conjunto de objetos ordenado.</li> </ul>	<ul style="list-style-type: none"> <li>• “coleção” conjunto de jogos de tabuleiro;</li> <li>• “interligação”: atividades de uma aula;</li> <li>• “hierarquia”: cursos com uma árvore de pré-requisitos; ou</li> <li>• “linear”: conjunto de livros formando módulos de ensino.</li> </ul>
1.8	Nível de agregação	Granularidade do objeto	Seleção de um dos valores a seguir: <ul style="list-style-type: none"> <li>• 1: mais baixo nível de agregação, para objetos simples;</li> <li>• 2: coleção de objetos do nível 1;</li> <li>• 3: coleção de objetos do nível 2; ou</li> <li>• 4: o maior nível de granularidade.</li> </ul>	<ul style="list-style-type: none"> <li>• “nível 1”, um jogo;</li> <li>• “nível 2”, uma aula;</li> <li>• “nível 3”, um curso; ou</li> <li>• “nível 4”, o conjunto de cursos para tirar uma certificação.</li> </ul>

Tabela 5 – Categoria geral do modelo LOM (traduzida de IEEE, 2002, p. 10-15).

2. Metadados do ciclo de vida				
Número	Nome	Explicação	Tipo e valores possíveis	Exemplo
2.1	Versão	Versão do objeto	Texto simples	<ul style="list-style-type: none"> <li>• “1.1.2”;</li> <li>• “2.0rc2”; ou</li> <li>• “Ganymede Version”.</li> </ul>
2.2	Estado	O estado ou condição do objeto	Seleção de um dos valores a seguir: <ul style="list-style-type: none"> <li>• rascunho;</li> <li>• final;</li> <li>• revisado; ou</li> <li>• indisponível</li> </ul>	“Rascunho”
2.3	Contribuinte	Entidade (pessoa ou organização) que contribuiu no ciclo de vida do objeto (criação, edição, publicação, disponibilização, etc.).	--	--
2.3.1	Papel	Tipo de contribuição	Seleção dentro deste conjunto de valores: <ul style="list-style-type: none"> <li>• autor;</li> <li>• publicador;</li> <li>• iniciante;</li> <li>• finalizador;</li> <li>• validador;</li> <li>• editor;</li> <li>• designer gráfico;</li> <li>• implementador técnico;</li> <li>• validador técnico;</li> <li>• validador educacional;</li> <li>• roteirista;</li> <li>• designer educativo; ou</li> <li>• especialista no assunto.</li> </ul>	“Finalizador” (para uma entidade que deixou o objeto no estado “indisponível”).

2.3.2	Entidade	Identificação e informação sobre a entidade que contribuiu no ciclo de vida do objeto.	Texto no formato IMC vCard 3.0 <sup>4</sup>	"BEGIN:VCARD FN:João Piccinini TEL:+55-48-3200-0000 TITLE:estudante de Sistemas de Informação EMAIL\;TYPE=INTERNET:joao.piccinini@ufsc.com.br END:VCARD"
2.3.3	Data	Data da contribuição	Data no formato "yyyy-MM-dd"	"2013-01-01"

Tabela 6 - Categoria de ciclo de vida do modelo LOM (traduzida de IEEE, 2002, p. 16-17)

3. Meta-metadados				
Número	Nome	Explicação	Tipo e valores possíveis	Exemplo
3.1	Identificador	Valor global único que identifica o registro do metadado.	--	--
3.1.1	Catálogo	Catálogo global que armazena a identificação do metadado.	Texto simples	"ISBN", "ARIADNE" ou "GQS-UFSC"
3.1.2	Registro	Valor que identifica o metadado dentro de um esquema de catalogação.	Texto simples	"KUL543" ou "http://www.gqs.ufsc.br/metadados/123"
3.2	Contribuinte	Entidade (pessoa ou organização) que contribuiu no ciclo de vida do metadado (criação, validação, etc.).	--	--
3.2.1	Papel	Tipo de contribuição	Seleção de um dos valores a seguir: <ul style="list-style-type: none"> <li>• criador; ou</li> <li>• validador.</li> </ul>	"criador"
3.2.2	Entidade	Identificação e informação sobre a entidade que contribuiu com o metadado	Texto no formato vCard	"BEGIN:VCARD FN:João Piccinini TEL:+55-48-3200-0000 TITLE:estudante de Sistemas de Informação EMAIL\;TYPE=INTERNET:joao.piccinini@ufsc.com.br END:VCARD"
3.2.3	Data	Data da contribuição	Data no formato "yyyy-MM-dd"	"2013-01-01"
3.3	Esquema do metadado	O nome e versão da especificação oficial utilizada para criar a instância dos metadados.	Texto simples	"LOMv1.0", "IGRver.2013" ou "SGRv1.0"
3.4	Língua	Idioma utilizado pelo metadado	Texto simples	"pt-BR", "em-US-philadelphia" ou "fr-CA"

Tabela 7 - Categoria de meta-metadados do modelo LOM (traduzida de IEEE, 2002, p. 10-15)

## 4. Metadados técnicos

<sup>4</sup> Mais informações sobre o formato IMC vCard 3.0 podem ser encontradas nos documentos RFC 2425 e RFC 2426.

Número	Nome	Explicação	Tipo e valores possíveis	Exemplo
4	Técnico	Categoria com as características e requisitos técnicos do objeto.	--	--
4.1	Formato	Formato do objeto	Tipos MIME baseados no registro IANA <sup>5</sup>	"vídeo/mpeg" ou "text/html"
4.2	Tamanho	Tamanho digital do objeto	Números expressos na unidade de "bytes"	"4200" ou "5767168"
4.3	Localização	Local (URL ou URI) onde o objeto está armazenado	Texto simples	"http://xp123.com/articles/scrum-from-hell/"
4.4	Requisitos	Requisitos técnicos necessários para a utilização do objeto	--	--
4.4.1	Composição	Grupo de múltiplos requisitos	--	--
4.4.1.1	Tipo	Tipo de tecnologia necessária para a utilização do objeto.	Texto simples	"Sistema Operacional", "Aplicativo" ou "Navegador Web"
4.4.1.2	Nome	Nome da tecnologia necessária para a utilização do objeto.	Texto simples	"Windows", "Windows Media Player", "Mozilla Firefox" ou "Microsoft Internet Explorer"
4.4.1.3	Versão mínima	Versão mínima permitida da tecnologia necessária para a utilização do objeto	Texto simples	"XP" ou "1.0"
4.4.1.4	Versão mínima	Versão mínima permitida da tecnologia necessária para a utilização do objeto	Texto simples	"Vista" ou "1.5"
4.5	Descrição de instalação	Descrição de como instalar o objeto	Texto simples	"Descompacte o arquivo jogo.rar e, dentro da pasta do jogo, execute o arquivo jogar.jar".
4.6	Outros requisitos	Informações sobre outros <i>softwares</i> ou <i>hardwares</i> necessários (que não podem ser descritos através do metadado 4.4 – Requisitos)	Texto simples	"Placa de som", "Direct X"
4.7	Duração	Duração do objeto de aprendizagem em uso contínuo.	Texto no formato "P[yY][mM][dD][T[hH][mM][s.s]S]" <sup>6</sup>	"PT51M37S" para uma vídeo-aula, ou P1DT10M" para um coleção de livros em áudio.

Tabela 8 - Categoria de metadados técnicos do modelo LOM (traduzida de IEEE, 2002, p. 10-15)

## 5. Metadados educacionais

Número	Nome	Explicação	Tipo e valores possíveis	Exemplo
5	Educacional	Categoria que descreve as características educacionais e pedagógicas do objeto	--	--
5.1	Tipo de	Principal modo de	Seleção de um dos valores a	"ativa" para simulações,

<sup>5</sup> Mais informações sobre o registro IANA podem ser encontradas no documento RFC2048:1996.

<sup>6</sup> Formato de tempo ou duração no padrão ISO8601.

	interação	aprendizagem para o objeto	seguir: <ul style="list-style-type: none"> <li>• ativa;</li> <li>• expositiva; ou</li> <li>• mista.</li> </ul>	questionários e exercícios; ou “expositiva” para material de áudio ou de vídeo.
5.2	Tipo de recurso de aprendizagem	Tipo específico de objeto de aprendizagem	Seleção de um dos valores a seguir: <ul style="list-style-type: none"> <li>• exercício;</li> <li>• simulação;</li> <li>• questionário;</li> <li>• diagrama;</li> <li>• figura;</li> <li>• gráfico</li> <li>• índice;</li> <li>• “lâmina” de apresentação;</li> <li>• tabela;</li> <li>• texto narrativo;</li> <li>• exame;</li> <li>• experimento;</li> <li>• enunciado de problema;</li> <li>• auto avaliação; ou</li> <li>• palestra;</li> </ul>	“exercício” ou “exame”.
5.3	Grau de interação	Grau que determina o quanto o usuário influencia no comportamento do objeto	Seleção de um dos valores a seguir: <ul style="list-style-type: none"> <li>• muito baixo;</li> <li>• baixo;</li> <li>• médio;</li> <li>• alto; ou</li> <li>• muito alto.</li> </ul>	“alto” para uma simulação com o controle do usuário, ou “baixo” para um experimento com o conjunto de instruções pré-definido.
5.4	Densidade semântica	Grau de densidade semântica do objeto	Seleção de um dos valores a seguir: <ul style="list-style-type: none"> <li>• muito baixa;</li> <li>• baixa;</li> <li>• média;</li> <li>• alta; ou</li> <li>• muito alta.</li> </ul>	“baixa” densidade semântica, para uma lâmina de apresentação com um texto, uma figura e um botão para avançar à próxima lâmina; ou “muito alta” densidade semântica, para um jogo com dois grupos de oito pessoas, blocos de LEGO, cartazes e <i>PostIts</i> para planejamento, etc..
5.5	Público alvo	Principal usuário ou grupo de usuários para qual o objeto foi desenvolvido	Seleção de um dos valores a seguir: <ul style="list-style-type: none"> <li>• professor;</li> <li>• autor;</li> <li>• aprendiz; ou</li> <li>• gestor.</li> </ul>	<ul style="list-style-type: none"> <li>• “professor”, para objetos que devem ser utilizados em conjunto com outros, ou aplicados à um contexto educacional específico;</li> <li>• “autor”, para objetos que devem ser utilizados na construção de outros objetos;</li> <li>• “aprendiz”, para objetos que permitem a extração do conhecimento diretamente;</li> <li>• “gestor” educacional, para objetos que definem ou auxiliam na definição da estrutura de um curso, como currículo e plano educacional.</li> </ul>
5.6	Contexto	Principal ambiente onde o objeto é utilizado.	Seleção de um dos valores a seguir: <ul style="list-style-type: none"> <li>• escola;</li> <li>• ensino superior;</li> <li>• treinamento;</li> <li>• outros;</li> </ul>	“ensino superior”
5.7	Faixa etária recomendada	Faixa etária do usuário que utilizará o objeto	Texto simples	“7-9”, “15+” ou “recomendado para adultos, apenas”
5.8	Dificuldade	Grau de dificuldade para utilizar o objeto.	Seleção de um dos valores a seguir:	“muito fácil”

			<ul style="list-style-type: none"> <li>• muito fácil;</li> <li>• fácil;</li> <li>• médio;</li> <li>• difícil; ou</li> <li>• muito difícil.</li> </ul>	
5.9	Duração	Tempo aproximado que o usuário leva para utilizar o objeto <sup>7</sup>	Texto no formato "P[yY][mM][dD][T[hH][mM][s].s]S"	"PT51M37S" ou PT10M"
5.10	Descrição	Comentários sobre como este objeto deve ser utilizado.	Texto simples	"O tabuleiro, os dados, as peças e as cartas devem ser impressas e recortadas..." ou "O guia para o jogo encontra-se no documento README.txt"
5.11	Língua	Idioma típico do usuário que utiliza o objeto <sup>8</sup> .	Texto simples	"pt-PT" ou "pt-BR".

Tabela 9 - Categoria de metadados educacionais do modelo LOM (traduzida de IEEE, 2002, p. 10-15)

6. Metadados sobre direitos autorais				
Número	Nome	Explicação	Tipo e valores possíveis	Exemplo
6.1	Custo	Indicativo de que a utilização deste objeto é permitida mediante à pagamento.	Seleção de um dos valores a seguir: <ul style="list-style-type: none"> <li>• sim; ou</li> <li>• não.</li> </ul>	"sim"
6.2	Direitos e outras restrições	Indicativo de que direitos autorais e restrições são aplicadas à este objeto.	Seleção de um dos valores a seguir: <ul style="list-style-type: none"> <li>• sim; ou</li> <li>• não.</li> </ul>	"não"
6.3	Descrição	Comentários sobre às restrições de utilização do objeto	Texto simples	"A utilização deste objeto é permitida desde que não seja utilizado para fins comerciais e que o nome do autor seja divulgado junto com o material.

Tabela 10 - Categoria de metadados sobre direitos autorais do modelo LOM (traduzida de IEEE, 2002, p. 10-15)

7. Metadados sobre relacionamentos entre objetos				
Número	Nome	Explicação	Tipo e valores possíveis	Exemplo
7.1	Tipo	Natureza do relacionamento entre os objetos	Seleção de um dos valores a seguir: <ul style="list-style-type: none"> <li>• faz parte do outro objeto;</li> <li>• contém o outro objeto;</li> <li>• é uma versão do outro objeto;</li> <li>• é o formato do outro objeto;</li> <li>• contém o formato do outro objeto;</li> <li>• referencia o outro objeto;</li> <li>• é referenciado pelo outro</li> </ul>	O objeto jogo de adivinhações "contém" o objeto baralho de cartas.

<sup>7</sup> O metadado de Duração [5.9], da categoria Educacional [5], é destinado a representar o tempo necessário para a utilização do objeto como um todo. Considerando, por exemplo, uma palestra em vídeo, o professor levaria quarenta minutos para utilizar o objeto como um todo onde, vinte minutos seriam destinados à apresentação e comentários posteriores sobre o vídeo, e os outros vinte minutos seriam utilizados para a execução do vídeo propriamente dito (Duração [4.7], da categoria Técnica [4]).

<sup>8</sup> O metadado de Língua [5.11], da categoria Educacional [5], é utilizado para descrever o idioma do usuário, enquanto que o metadado Língua [1.3], da categoria Geral [1], representa no qual o objeto foi construído. Um jogo desenvolvido em português, do Brasil, por exemplo, pode ser utilizado por portugueses e brasileiros.

			objeto; <ul style="list-style-type: none"> <li>• é baseado no outro objeto;</li> <li>• é a base para o outro objeto;</li> <li>• requer o outro objeto;</li> <li>• é um requisito do outro objeto;</li> </ul>	
7.2	Recurso	O objeto alvo ao qual o relacionamento se referencia.	--	--
7.2.1	Identificador	Valor global único que identifica o registro do objeto alvo.	--	--
7.2.1.1	Catálogo	Catálogo global que armazena a identificação do objeto alvo.	Texto simples	"ISBN", "ARIADNE" ou "GQS-UFSC"
7.2.1.2	Registro	Valor que identifica o objeto alvo dentro de um esquema de catalogação.	Texto simples	"2-7342-0318" ou "http://www.gqs.ufsc.br/documentos/123"
7.2.2	Descrição	Descrição do objeto alvo	Texto simples	"Jogo composto por tabuleiro, dois baralhos de cartas, cinco peças identificando os jogadores e dois dados".

Tabela 11 - Categoria de metadados sobre relacionamentos entre objetos do modelo LOM (traduzida de IEEE, 2002, p. 10-15)

8. Metadados para anotação				
Número	Nome	Explicação	Tipo e valores possíveis	Exemplo
8.1	Entidade	Identificação e informação sobre a entidade que criou a anotação	Texto no formato vCard	"BEGIN:VCARD FN:João Piccinini TEL:+55-48-3200-0000 TITLE:estudante de Sistemas de Informação EMAIL\;TYPE=INTERNET:joao.piccinini@ufsc.com.br END:VCARD"
8.2	Data	Data da criação da anotação	Data no formato "yyyy-MM-dd"	"2013-01-01"
8.3	Descrição	Conteúdo da anotação	Texto simples	"Eu utilizei este jogo em uma das aulas que ministrei sobre gerenciamento ágil de projetos para cerca de 30 alunos. Eles acharam gostaram muito da experiência."

Tabela 12 - Categoria de metadados para anotação do modelo LOM (traduzida de IEEE, 2002, p. 10-15)

9. Metadados para classificação				
Número	Nome	Explicação	Tipo e valores possíveis	Exemplo
9.1	Propósito	O propósito de classificação do objeto	Seleção de um dos valores a seguir: <ul style="list-style-type: none"> <li>• disciplina;</li> <li>• conceito;</li> <li>• pré-requisito;</li> <li>• objetivo educacional;</li> <li>• acessibilidade;</li> <li>• restrições;</li> <li>• nível educacional;</li> <li>• nível de habilidades;</li> <li>• nível de segurança; ou</li> </ul>	"disciplina"

			• competência;	
9.2	Taxionomia	Taxionomia utilizada na classificação	--	--
9.2.1	Nome	Nome da taxionomia	Texto simples	"Bloom"
9.2.2	Termo	Valor particular da taxionomia	--	--
9.2.2.1	Identificador		Texto simples	"1", "1203" ou "BF180"
9.2.2.2	Registro	Rótulo da taxionomia	Texto simples	"conhecimento"
9.3	Descrição	Descrição do propósito da classificação	Texto simples	"informações sobre fatos específicos, padrões e conceitos".
9.4	Palavras-chave	Palavras-chave ou frases que descrevem o sistema de classificação.	Texto simples	"objetivo de aprendizagem", "classificação de objetivos de aprendizagem" e "taxionomia de Bloom"

Tabela 13 - Categoria de metadados para classificação do modelo LOM (traduzida de IEEE, 2002, p. 10-15)

Segundo Roy, Sarkar e Ghose (2010, p. 106) existem outros conjuntos de metadados que são amplamente utilizados e possuem semelhanças nos elementos e na semântica em relação ao IEEE LOM, dentre eles, é possível destacar: *Dublin Core Metadata Initiative*, *IMS Content Packaging Information Model*, *Sharable Content Object Reference Model (SCORM)* e *CanCore Learning Resource Metadata Initiative*.

Objetos de aprendizagem com estes metadados podem ser disponibilizados na rede interna de uma escola, em uma página da Web disponível para professores, ou em um repositório aberto à comunidade; e é justamente neste último caso que as vantagens da utilização, não apenas de metadados, mas dos objetos de aprendizagem como um todo, se tornam mais perceptíveis.

## 2.4. Repositório de Objetos de Aprendizagem

Um repositório, segundo Barton e Waters (2005), é um sistema que fornece um conjunto de serviços para guardar e recuperar dados, proporcionando meios de preservar e redistribuir o material armazenado em formato digital.

As características básicas de um repositório digital, segundo Heery (2005), são:

- O conteúdo é sempre armazenado em um repositório, inserido pelo criador, pelo dono ou por um terceiro;
- A arquitetura de um repositório gerencia o conteúdo e os metadados;
- O repositório deve ser sustentável, confiável, bem gerenciado e deve possuir

suporte técnico constante;

- O repositório deve oferecer um conjunto básico de serviços, e.g. inserção, recuperação, busca, e controle de acesso.
- Um conjunto adicional de serviços pode ser estipulado, contendo:
  - Acesso facilitado à recursos;
  - Modelos diferenciados de publicação e revisão de objetos;
  - Gerencia de informações corporativas;
  - Compartilhamento e reuso de objetos;
  - Preservação de recursos digitais.

Repositórios de objetos de aprendizagem podem ser diferenciados por duas macroestruturas: centralizada e distribuída, decompostas nas quatro estruturas da tabela 14 (HARMAN e KOOHANG, 2007).

Arquitetura dos servidores		Vantagens	Dificuldades	Ilustração
Objetos	Metadados			
Centralizada	Centralizada	Melhor desempenho na busca e recuperação dos metadados e LOs, por não depender da uma rede para comunicação com outros servidores.	Necessidade de infraestrutura robusta para suportar uma forte demanda de busca e exibição dos LOs e metadados.	
Centralizada	Distribuída	Diminuição nos custos de processamento para a busca dos objetos. Possibilidade de ter diferentes padrões de metadados para referenciar o mesmo objeto.	Necessidade um serviço de registros para manter a localização dos servidores dos metadados.	

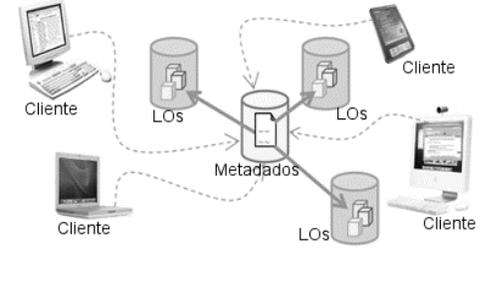
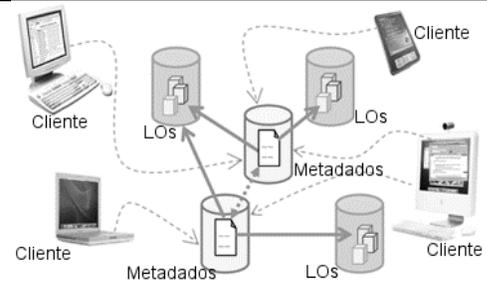
Distribuída	Centralizada	Diminuição nos custos para armazenamento dos LOs, sendo necessário apenas a infraestrutura para os metadados.	Se houver um erro no servidor central (dos metadados), o sistema ficará indisponível	
Distribuída	Distribuída	Distribuição do processamento de busca e exibição entre os servidores e tolerância a falhas.	Custo da infraestrutura e dificuldade de manutenção	

Tabela 14 – Comparativo entre os tipos de arquitetura para os servidores de objetos de aprendizagem e metadados (HARMAN; KOOHANG, 2007, p. 134-137)

Harman e Koohang (2007, p. 138) destacam a tendência de construção de repositórios com arquitetura centralizadas, principalmente com arquitetura totalmente centralizada e com arquitetura de índices centralizados e objetos distribuídos (considerando repositórios como MERLOT, CAREO, ARIADNE, Splash e MIT Opencourseware).

Outro requisito que deve ser considerado na fase de construção de um repositório é a estratégia utilizada para armazenar e recuperar os objetos de aprendizagem e os metadados. As principais estratégias são listadas a seguir:

- Armazenamento direto em arquivo: utiliza índices para referenciar formatos pré-determinados de arquivos que são armazenados em um sistema operacional;
- Armazenamento em banco de dados: utiliza uma estrutura de armazenamento com um sistema de gerenciamento (SGBD) que realiza a abstração da lógica operacional e fornece operações para consulta, recuperação, alteração e remoção. Flexibilidade limitada de modelos de armazenamento, podendo ser relacional, orientado a objetos, híbrido ou XML.
- Armazenamento em containers de objetos: utiliza a estrutura de objetos serializados armazenados em containers específicos de plataformas de desenvolvimento corporativas (Java e .NET).

Os requisitos computacionais levantados, referente à distribuição dos objetos de aprendizagem e dos metadados e à estrutura de armazenamento, e os modelos propostos não são suficientes para garantir a difusão, com valores educacionais, de materiais para um ensino de qualidade.

Levando em consideração o ponto de vista pedagógico, Buseti et al (2004, p. 1) aponta uma série de dificuldades em aproveitar um repositório de objetos de aprendizagem que utilize o modelo de metadados IEEE LOM. Os principais fatores são listados a seguir (BUSETTI et al, 2004, p.2):

- Distância entre a comunidade que desenvolveu o repositório e a comunidade que o utiliza, sobretudo com o contraste entre a linguagem tecnológica e a linguagem educacional; que resulta na diferença entre os aspectos técnicos, como segurança, desempenho e disponibilidade; e os aspectos educacionais, como objetivos de aprendizagem, ensino e avaliação de qualidade;
- Dificuldade em individualizar um conjunto de metadados para balancear a definição das características essenciais de um objeto (fins de produção e reutilização), com a definição precisa de requisitos (fim de localização). Um problema similar ocorre da dificuldade de conciliação da definição de metadados a um nível internacional, com as especificidades de sistemas nacionais de educação.
- Dificuldade em representar objetos de natureza construtivista no formato de objetos de aprendizagem;
- A produção de objetos de aprendizagem é vista mais como um problema de desenvolvimento de *software* do que um problema educacional. Portanto, o ponto de vista do educador e as necessidades de ferramentas e treinamento são consideradas de forma superficial na grande maioria dos casos.

Ao projetar e criar objetos de aprendizagem, o instrutor deve planejar de que forma este objeto será distribuído: com a atribuição de direitos autorais, que permitam o uso mediante o pagamento de uma determinada quantia monetária, ou com a atribuição da identificação do criador, permitindo o uso sobre determinadas circunstâncias. Todos os objetos de um repositório devem exibir uma informação precisa sobre os termos nos quais

estão licenciados.

Utilizar uma obra para comentários, críticas ou para propósitos educacionais é ação permitida na maioria das licenças, mas isso significa que partes da obra podem ser citadas ou copiadas, não a obra integral.

A organização *Creative Commons* (2013), por exemplo, oferecem uma forma padronizada para publicação de condições de uso e compartilhamento, através das licenças listadas a seguir:

- **CC BY:** permite a distribuição, alteração e comercialização, inclusive, desde que seja mencionado o autor original;
- **CC BY-SA:** permite a distribuição, alteração e comercialização, desde que as obras resultantes sejam distribuídas sobre a mesma licença;
- **CC BY-ND:** permite a redistribuição, comercial ou não, desde que a obra permaneça inalterada;
- **CC BY-NC:** permite a distribuição, não comercial, sobre uma licença que pode diferir da original;
- **CC BY-NC-SA:** permite a distribuição e alteração, desde que seja feita de forma não comercial e sob os mesmos termos da obra original; e
- **CC BY-NC-ND:** permite a redistribuição, não comercial, da obra, desde que permaneça inalterada.

A manutenção do repositório no âmbito tecnológico, na qualidade do conteúdo, na conformidade com direitos autorais, na garantia da privacidade e no respeito entre membros deve ser realizada por um grupo de usuários com perfil restrito, para que apenas usuários experientes e confiáveis possam realizar modificações que afetam a exibição de informações e a experiência de outros usuários no repositório. Um usuário deste perfil pode realizar a manutenção por motivação profissional, remunerada, ou por interesse particular no repositório ou em alguma atividade associada a ele.

Na seção a seguir são apresentados o conceito, as características e os princípios das comunidades que compartilham interesses comuns, alheios ao ambiente corporativo, que podem ser utilizados para a manutenção de um repositório de jogos educativos.

## 2.5. Comunidades de Prática

“Comunidades de prática são grupos de pessoas que compartilham um interesse em comum, um conjunto de problemas ou a paixão por um assunto, aprofundando seus conhecimentos e práticas nestas áreas em um processo contínuo” (WENGER; MCDERMOTT; SNYDER, 2002, p. 4).

A finalidade desta comunidade não está no acúmulo de valores materiais, embora, no decorrer do processo, possam ser criadas ferramentas, padrões e manuais. O maior ativo de uma comunidade de prática é o conhecimento adquirido através do compartilhamento de informações, conselhos e avaliações entre os participantes. A discussão sobre a situação de um projeto, a aspiração e as necessidades de um nicho de mercado, e o auxílio na resolução de problemas são apenas alguns exemplos de atividades de uma comunidade de prática (WENGER; MCDERMOTT; SNYDER, 2002, p. 4;5).

Para que este conhecimento seja disseminado, é possível tratá-lo como um objeto, separado do ser humano, que contém informação codificada; como uma parte inerente e inseparável da mente; ou como uma forma de atividade ligada diretamente à prática em uma comunidade (WASKO; FARAJ, 2000, p 157-161). Estas manifestações do conhecimento e suas características podem ser visualizadas na tabela 15.

	<b>Conhecimento como objeto</b>	<b>Conhecimento embutido nas pessoas</b>	<b>Conhecimento embutido na comunidade</b>
<b>Definição</b>	Independente da percepção humana	Internalizado	Prática do saber (da experiência)
<b>Organização</b>	Conteúdo organizacional, como documentos, bases de dados e objetos de aprendizagem	Soma do conhecimento individual	Linguagem compartilhada, narrativas e códigos
<b>Tecnologias de compartilhamento</b>	Repositórios e agentes de busca	E-mail, telefone, conversação direta e mapas mentais	Grupos de discussão, salas de <i>chat</i> e quadros brancos
<b>Desenvolvimento</b>	Precisa ser codificado, descontextualizado Torna-se um ativo, uma posse de seu criador	Precisa de identificação e de interações para a transferência	É desenvolvido no contexto de uma comunidade Permite a criação de fluxos de conhecimento Torna-se um bem da comunidade
<b>Direitos autorais</b>	Organizacional	Individual	Comunitário
<b>Motivação para o compartilhamento</b>	Interesse pessoal	Interesse pessoal	Obrigação moral
<b>Interesses no compartilhamento</b>	Extrínseco ou por compensação financeira	Por reputação, <i>status</i> ou obrigação	Por reciprocidade, necessidade de atualização ou acesso à comunidade

Tabela 15 - Conhecimento nas entidades de uma comunidade e suas características (adaptado de WASKO; FARAJ, 2000, p. 158).

Para que estas manifestações do conhecimento possam ser exploradas, sem fugir da área de conhecimento e dos objetivos definidos para o grupo, Wenger, McDermott e Snyder (2002, p. 45-46) sugerem a utilização de um modelo como guia para o desenvolvimento de uma comunidade de prática, com as três dimensões a seguir:

- Domínio: esta dimensão da comunidade indica a área de conhecimento, assunto ou tópico tratados por ela. Para a estruturação do domínio é necessário considerar, principalmente, as lacunas teóricas, os problemas práticos e a conexão com o universo externo (empresas, cursos, universidades e público geral).
- Comunidade: ao perseguir seus interesses em um determinado domínio, os membros inscrevem-se em atividades conjuntas e discussões, ajudando uns aos outros e compartilhando o conhecimento. Estes relacionamentos que interligam os membros da comunidade podem ser promovidos ou preservados por grupos e elementos de liderança (coordenadores da comunidade).
- A prática: uma comunidade de prática não é apenas uma comunidade de interesses, é uma comunidade de praticantes, onde histórias, experiências e ferramentas são desenvolvidas e compartilhadas. Para que este trabalho seja realizado em favor da comunidade, a prática precisa ser bem definida, levando em consideração o que deve ser compartilhado, desenvolvido e documentado; quais objetos de aprendizagem devem ser considerados; como o repositório de conhecimento deve ser organizado (para refletir a prática dos membros e ser acessado facilmente) e quais fontes de conhecimento e métricas de avaliação existem fora da comunidade.

Segundo, Wenger, McDermott e Snyder (2002, p. 46), as atividades dos três domínios devem ser desenvolvidas em paralelo. Ao tentar desenvolver um repositório de jogos sem um domínio claro ou sem uma comunidade coesa, por exemplo, o resultado pode ser a geração de uma ferramenta inútil. Analogicamente, a perda de foco na construção de uma prática em comum, pode resultar em um simples grupo de amizade, ineficaz como comunidade, por mais que haja satisfação no âmbito social.

Para que a construção e manutenção de uma comunidade de prática consiga

balancear o desenvolvimento de atividades nos três domínios citados, Wenger, McDermott e Snyder (2002, p. 51-63) propõe a utilização dos sete princípios a seguir:

1. Coordenação para evolução – como as comunidades são orgânicas e dinâmicas, novos participantes podem sugerir modificações no domínio. Cabe à comunidade a tarefa de gerenciar tais mudanças; como no caso da Wikipedia, onde a própria comunidade altera as páginas e faz o papel de moderação para conteúdo inadequado.
2. Abordagem dos problemas por perspectivas opostas – um integrante com uma visão interna da comunidade, que seja membro por um período considerável, possui um nível de compreensão e empatia com os demais integrantes e com os desafios impostos, enquanto que um indivíduo externo, por causa da falta de conhecimento deste ambiente, pode propor melhorias que dificilmente seriam sugeridas por um membro antigo da comunidade.
3. Estímulo de diferentes níveis de participação – a participação de integrantes com interesses distintos é normal e deve ser aceita na comunidade, a participação conjunta de coordenadores, membros e meros interessados deve ser estimulada.
4. Desenvolvimento de espaços públicos e privados - espaços públicos são essências para a divulgação da comunidade, mas as discussões privadas, sobre problemas e tópicos sensíveis devem ser respeitadas.
5. Foco no valor – a comunidade deve sempre gerar valor para a organização que a suporta, para os times em que serve ou para seus próprios membros;
6. Equilíbrio do trabalho árduo com trabalho confortável – o suporte a discussões sem compromisso, que não necessitem de investimento maior do interessado é essencial para a comunidade; devem haver eventos nos quais as pessoas possam se sentir conectadas e outros nos quais possa haver um comprometimento total com a causa;
7. Criação de um ritmo para a comunidade – um ritmo lento, pode não significar diretamente que a comunidade está prestes a se desfazer, apenas que os seus integrantes não têm tempo ou motivação necessária para um ritmo diferente; portanto, manter um ritmo para a comunidade, mesmo que lento, é o melhor indicador para garantir se está ainda está vida.

Os princípios citados devem ser seguidos não apenas na criação, mas em todo o desenvolvimento de uma comunidade; como esta possui um forte dinamismo, não há garantias de que um membro que tenha auxiliado na criação da especificação do domínio, da prática ou da comunidade, se mantenha nela por muito tempo.

A especificação dos níveis de participação dos membros é um princípio diretamente vinculado aos de coordenação para evolução e de estímulo à opiniões opostas. Wenger, McDermott e Snyder (2002, p. 56), destacam que os níveis de participação das pessoas em uma comunidade de prática podem ser estruturados nos grupos à seguir:

- **Principal:** pessoas com participação ativa, em discussões, debates e fóruns. Considerado como o “coração” de uma comunidade, este pequeno grupo (dez por cento da comunidade) tem a responsabilidade de identificar novos tópicos para discutir, de criar e gerenciar projetos e de coordenar o trabalho de todos os grupos.
- **Ativo:** os membros deste grupo (quinze por cento da comunidade) participam da maior parte dos encontros e fóruns, mas sem a regularidade e a intensidade do grupo principal.
- **Periférico:** composto por pessoas que raramente participam das atividades da comunidade. Esta contribuição limitada do participante pode ser por falta de tempo, ou decorrente do sentimento de que suas observações não são úteis para a comunidade, ou que não possuem autoridade suficiente para opinar. As pessoas deste grupo, por não estarem engajadas ativamente, possuem uma opinião própria, menos fundamentada na comunidade, que, quando exposta, contribui com novas ideias como “agente de mudança”.
- **Externo:** pessoas sem participação direta, mas que contribuem ao expor problemas ou dúvidas que são capturados pelos participantes da comunidade, e ao utilizar artefatos produzidos pela comunidade.

As barreiras dos quatro níveis de participação são fracas, fluídas, e proporcionam, ao participante, a oportunidade de se movimentar entre os diferentes níveis sempre que precisar e merecer. Segundo Wenger (2010, p. 133-134), um participante pode se sentir confortável

no grupo periférico, com pequenas contribuições e com um aprendizado suficiente para seus objetivos (**trajetória periférica**). Outro participante pode entrar na comunidade com o objetivo de fazer parte do grupo principal (**trajetória de entrada**), de contribuir ativamente, desenvolvendo sua participação para este fim, por mais que ocupe, atualmente, uma posição periférica. Ao atingir o grupo principal, a adesão plena à comunidade, um membro pode optar por evoluir a prática (**em uma trajetória interna**), com novos eventos, demandas e invenções; pode seguir na direção da saída da comunidade (**trajetória de saída**), considerando que atingiu seus objetivos pessoais; ou pode auxiliar na interação entre os níveis da comunidade (**trajetória de limite**), movendo as pessoas para os grupos em que mais se identificam e proporcionando a livre troca de informações e ideias entre eles. A figura 7 apresenta os níveis de participação, as trajetórias e exemplos de possíveis membros.

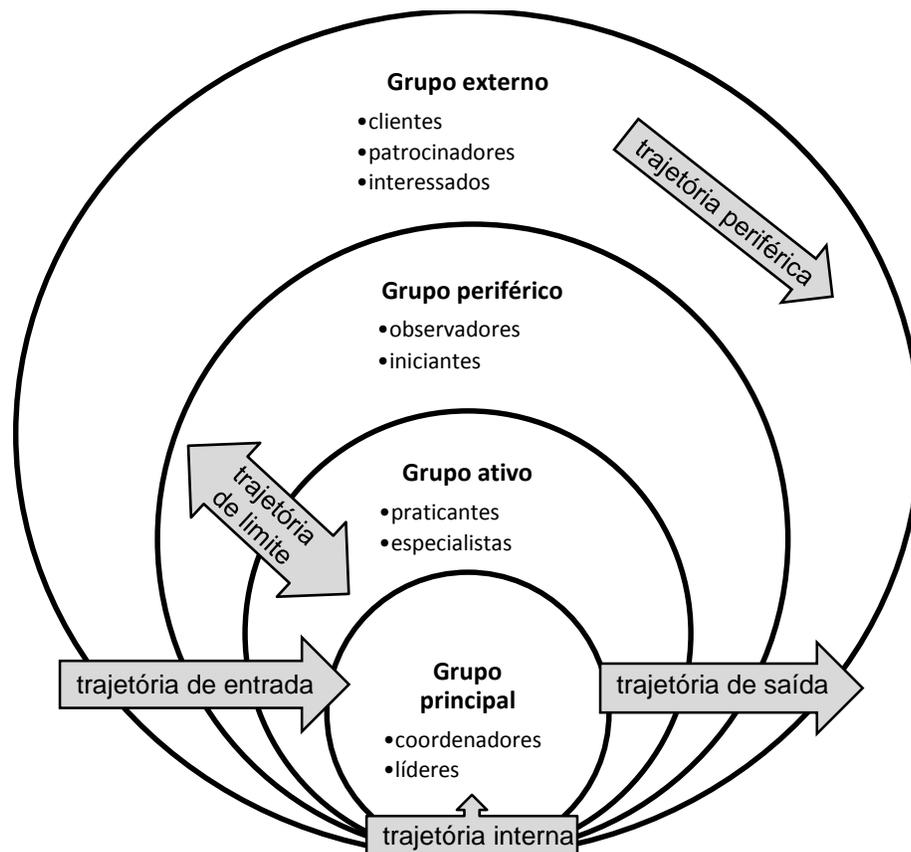


Figura 7 – Níveis de participação e trajetórias em uma comunidade de prática (adaptado de Wenger; Trayner, 2011)

A interação de pessoas com interesses e trajetórias distintos dentro de uma comunidade dificulta a criação de um ponto de encontro unificado para este grupo de

usuários. Como uma única ferramenta pode não satisfazer as necessidades de todos os grupos de usuários, é preferível disponibilizar atividades e canais em que a prática possa ser estimulada, como fóruns de discussão, teleconferências, listas de e-mails e projetos comunitários (SNYDER; BRIGGS, 2003, p. 13-15).

## 3. Estado da Arte

Neste capítulo é realizada a revisão do estado da arte em repositórios de jogos para o ensino do Scrum. Ao pesquisar e analisar os repositórios existentes, com jogos para o ensino do Scrum, busca-se a verificação do cumprimento dos requisitos necessários para possibilitar a execução das principais operações em um repositório de jogos.

### 3.1. Requisitos

Através da busca efetuada nos capítulos anteriores, juntamente com a definição de funcionalidades fundamentais para um repositório (HEERY, 2005), foram identificados os requisitos relevantes para um repositório de jogos (objetos de aprendizagem) para o ensino do Scrum, listados a seguir:

REQ. 1 - Exibição dos jogos para o ensino do Scrum e dos metadados, com imagens e descrição das características.

REQ. 2 - Busca por jogos, através dos seguintes critérios:

- a. Gerenciamento ágil: princípios;
- b. Objetivos educacionais: domínios cognitivo, afetivo e psicomotor;
- c. Scrum: cerimoniais, papéis e artefatos;
- d. Jogos: categoria (competição, sorte e simulação), grau de disciplina, duração e espaço físico necessário.

REQ. 3 - Adição, modificação e remoção de objetos de aprendizagem.

REQ. 4 - Avaliação dos jogos, em relação ao objetivo de aprendizagem proposto.

REQ. 5 - Controle de acesso para usuários.

### 3.2. Critérios de inclusão e exclusão

Os limites sobre localização, amplitude, tema e linguagens considerados na revisão do estado da arte, na tentativa de manter a objetividade e eficácia, são listados a seguir:

- **Localização:** sistema de buscas de páginas da Web da Google.com. Os *sites*

que não estão indexados no sistema da Google.com não são avaliados.

- **Amplitude:** a análise dos resultados encontrados considera os 100 primeiros resultados. Esta análise é feita em até dois níveis de profundidade, sendo que o primeiro nível é representado pelos resultados diretos e o segundo nível é representado por possíveis referências internas encontradas na página avaliada.
- **Ordenação:** os resultados estão na ordem das páginas mais referenciadas, ou que são referenciadas por outras páginas relevantes (com muitas referências), para as menos referenciadas, de acordo com o mecanismo Page Rank da Google.com.
- **Restrições:** os resultados que, pela identificação do título ou pela breve descrição disponibilizada pela ferramenta Google se mostrarem inadequados, não são avaliados. Esta restrição estende-se a repositórios para uso em redes internas em que o cadastro não está disponibilizado para o público geral.
- **Tema:** a busca está limitada aos repositórios que contiverem apenas jogos, para o ensino do Scrum ou práticas do gerenciamento ágil que possam ser aplicadas para o aprendizado ou suporte do Scrum.
- **Linguagem:** apenas os resultados que se encontram nas línguas portuguesa ou inglesa são avaliados.

### 3.3. Execução da busca

Nesta seção são apresentados os parâmetros utilizados e os resultados obtidos nas buscas por repositórios de jogos para o ensino do Scrum.

Com base nos requisitos identificados foi realizada uma consulta no sistema de busca de páginas da Google.com, no dia 8 de agosto de 2013, com os seguintes parâmetros: [*scrum AND ("objetos de aprendizagem" OR "learning objects" OR jogos OR games OR repository OR repositório)*].

Esta consulta retornou aproximadamente 43.200.000 registros de páginas, indexados através do mecanismo de Page Rank da Google.com.

Na revisão e análise dos 100 primeiros resultados, foi encontrado apenas o repositório

**TastyCupCakes** (2013), que oferece jogos para o ensino em geral, sem considerar especificamente a metodologia Scrum.

Após a constatação da dificuldade de encontrar repositórios que utilizassem critérios extraídos do Scrum para as operações de busca e exibição dos jogos, foi efetuado o relaxamento destes requisitos (REQ. 1 e REQ. 2), passando a considerar jogos em geral. O processo de análise foi então refeito, através do mesmo mecanismo de busca, com os seguintes parâmetros: [("*project management*" OR "*gerenciamento de projetos*" OR "*gerência de projetos*" OR "*scrum*" OR "*ágil*" OR "*agile*") AND (*instrução* OR *instruction* OR *educação* OR *education* OR "*objetos de aprendizagem*" OR "*learning objects*" OR *jogos* OR *games* OR *repository* OR *repositório*)].

Nesta segunda busca, com retorno de aproximadamente 806.000.000 registros, foi encontrado apenas um repositório que cumprisse com os novos requisitos estabelecidos: o **Instructional Games Repository** (GQS, 2013), referenciado por um tópico de discussão no *site* Nasaga (2012).

Por considerar como insuficiente o retorno de dois registros para as buscas realizadas, uma terceira foi efetuada com os mesmo parâmetros (da segunda), mas desconsiderando os cinco requisitos estabelecidos, passando a considerar simplesmente "listas de jogos para o ensino de Gerenciamento Ágil de Projetos, que contém mais de dez registros".

A terceira busca teve como retorno, além dos dois registros encontrados anteriormente, o **tópico de discussão Games for Scrum**, escrito por Sahota (2009); e o **grupo de discussão AgileGames** (HANOUILLE, 2013), referenciado por Löffler (2010).

### 3.4. Resultados Observados

Nesta seção são apresentados os resultados observados através da descrição das características gerais, de ilustrações sobre as principais funcionalidades e de uma análise de conformidade com os requisitos levantados.

#### 3.4.1. Características gerais e principais funcionalidades

O *blog* TastyCupCakes (MCCOULLOUGH; MCGREAL, 2013) apresenta um grande

conjunto de objetos de aprendizagem categorizados pelos assuntos sobre Gerenciamento Ágil em Geral, *Lean* e Gerenciamento de projetos. As listas sobre cada categoria (figura 8) apresentam um resumo básico sobre o objeto e a página em que o objeto está armazenado (figura 9) possui um padrão recomendado, mas opcional, para a descrição do mesmo.

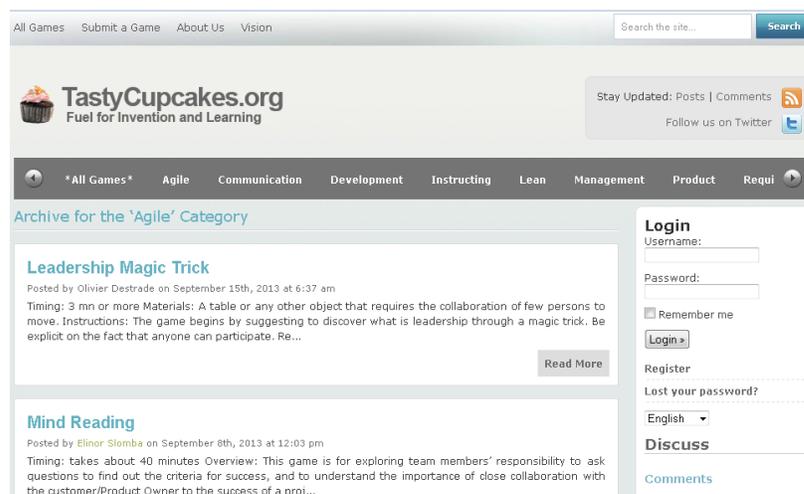


Figura 8 - Lista de jogos da categoria Agile, exibição de categorias e do campo de busca no repositório TastyCupCakes (MCCOULLOUGH; MCGREAL, 2013)

Filed Under: [cross-communication](#), [feed-back](#), [leadership](#) 0 Comments

## Leadership Magic Trick

Posted by Olivier Destrade on September 15th, 2013 at 6:37 am

**Timing:** 3 mn or more

**Materials:**  
A table or any other object that requires the collaboration of few persons to move.

**Instructions:**  
The game begins by suggesting to discover what is leadership through a magic trick. Be explicit on the fact that anyone can participate. Request the participants to come close the table.  
Announce that in order to prepare for the game, the table (or any object not easy to handle) should be moved to another location that you indicate in the room.  
Let participants spontaneously move the table.  
Notice that what just happened illustrates the first principle of leadership: announcing your intention:

- You announced your intention to the group
- This was received as a request for help. A few persons have given help spontaneously
- Other members of the group noticed that the action was taking place normally without the need for any more help. This was made possible BECAUSE THEY WERE INFORMED

**Learning Points:**  
This game can be used to initiate a group to collaboration within an auto-organised team. It can be used to introduce notions of leadership, cross-communication and feed-back.  
When facilitating the game, it is essential that you don't give orders nor use any commandment expression.  
You can increase the time of the game by asking more questions to the participants and discuss with them various aspects of the experience:

- Announcing intention
- Cross-communication
- Consequences of feed back and of lack of feed-back
- Notions of engagement and desengagement which associates with feed-back
- When does manipulation begins
- and so on.

Rate This  
★★★★★  
Rating: 5.0/5 (2 votes cast)

[Print](#) [PDF](#)

 Written by: [Olivier Destrade](#) on September 15, 2013.

Figura 9 – Exemplo de exibição de um jogo no repositório TastyCupCakes (MCCOULLOUGH; MCGREAL, 2013)

O *Instructional Games Repository* (GQS, 2013), ou simplesmente IGR, é um repositório de objetos de aprendizagem para suporte ao ensino de várias áreas da computação, como Qualidade de *Software*, Integração de Sistemas e Gerenciamento de Projetos, no qual o ensino do Scrum é, apenas, uma subárea desta última. Para buscar jogos neste repositório, existem parâmetros baseados não apenas nas áreas de ensino, mas em relação a objetivos de aprendizagem, classificação e características dos jogos (figura 10). A página de exibição do jogo apresenta os metadados considerados na busca e informações sobre a execução do jogo, material necessário e avaliação (figura 11).

Find and share educational games for teaching computing.

### Search

Use this form to define you search using any kind of combination of the attributes.

Name

#### APPLICATION DOMAIN

Computing Area

#### GAME CLASSIFICATION

Format

Type

Availability

#### GAME EXECUTION

Duration

Language

#### LEARNING OBJECTIVE

Cognitive Level

#### LEARNERS CHARACTERISTICS

In Groups of

Context

Domain

SEARCH

Figura 10 - Parâmetros para a busca por jogos no Instructional Games Repository (GQS, 2013).

## Detective Game - what killed the project?

### PHOTO



### INFORMATION

NAME	Detective Game - what killed the project?
CREATOR	Bruno Rausis, Gustavo Soares, Prof. Dr. rer. nat. Christiane Gresse von Wangenheim, PMP
PUBLICATION REFERENCE	
KEYWORDS	Earned value management
WEBSITE	<a href="http://www.gqs.ufsc.br/detective-game-what-killed-the-project/">http://www.gqs.ufsc.br/detective-game-what-killed-the-project/</a>

### CLASSIFICATION

FORMAT	Manual
TYPE	Other
LEARNERS CONSTELLATION	6 Students
COMPLEXITY	Medium
AVAILABILITY	Free

### EVALUATION

EVALUATION(S) PERFORMED	Yes
EVALUATION FOCUS	Impact on motivation, user experience and learning
TYPE OF EVALUATION	Case study
LEVEL OF EVALUATION	Level 1
NO. OF LEARNERS INVOLVED IN THE EVALUATION	From 50 to 100 Person
PRINCIPAL RESULTS OF THE EVALUATION	An evaluation of the game in project management courses at computing programs at the Federal University of Santa Catarina based on the students perception indicates the potential of such a game to contribute positively to the learning of EVM. In addition, the study indicated the very positive effect of the game's application on the motivation, social interaction, attention and its relevance to the course objectives. As such, the game offers a low-budget alternative instructional strategy that can easily be applied during a lecture within the classroom to complement (not to substitute) other instructional strategies, such as, expository lessons, to be adopted beforehand to present basic knowledge on EVM.

### RATING



Total votes: 2

### COMMENTS

THIAGO	Jogo muito bem elaborado e intuitivo!! Uma forma mais pratica de ensino que consegue prender a atenção do aluno! Parabéns aos autores.
--------	--



Figura 11 - Exemplo de exibição de um jogo no IGR (GQS, 2013).

O grupo AgileGames (HANOULLE, 2013) possui alguns tópicos com referências para outros repositórios de jogos, sem nenhuma estrutura ou metadados associados. Como se trata de um grupo de discussão (figura 12), na grande maioria dos casos, os jogos são obtidos através de solicitações de determinados usuários em um novo tópico, que são eventualmente respondidas por outros usuários (figura 13).

**AgileGames** Compartilhado publicamente  
60 de 189 tópicos (99+ não lidos) ★  

Membros · Sobre 

Hello,  
**Welcome** to the agile games mailing list.

I would like to invite the people who have not yet introduced themselves to answer the next 4 questions:

- Who you are?
- what was the first agile game you played?
- when you played that?
- what was your experience?

and anything you like to add .. ;-)

Yves Hanouille

---

 **are you interested in Pairwork?** (1)  
Por Yves Hanouille - 1 postagem - 3 visualizações 16 de set

---

 **Any suggestion for intro games for computer science college students, large (100+) group, Agile virgins?** (7)  
Por Kulawat Wongsaraj - 7 postagens - 13 visualizações 16 de set

---

 **Recommendation for 1 hour game: small stories, continuous improvement, very large group** (7)  
Por Ted M. Young - 8 postagens - 75 visualizações 11 de set

Figura 12 - Lista de tópicos de discussão sobre jogos no grupo AgileGames (HANOULLE, 2013).

**AgileGames** >  
**Recommendation for 1 hour game: small stories, continuous improvement, very large group**  
8 postagens de 5 autores  

★ **Ted M. Young** I'm looking for a game that would show how small stories tend to be better than larger stories in terms of delivering value sooner and making it easier to do 8 de jul

 **Ted M. Young** 2 de ago  

★ [Traduzir mensagem para português](#)

Still looking for suggestions, if you have any. Looking for a game that doesn't necessarily split people into roles.

;ted  
- mostrar texto das mensagens anteriores -

---

 **Mark Kilby** 2 de ago  

★ **Re: [AgileGames] Re: Recommendation for 1 hour game: small stories, continuous improvement, very large group**

[Traduzir mensagem para português](#)

Ted:

Sorry... I missed your earlier post. I've done many variations of the Ball Point Game.

Consider using paper wads

When I know I have some scrap paper (trying to recycle), I'll use wads of paper and I don't worry about making them a consistent size. If I know they have been doing some form of agile, I'll even deliberately make them different sizes with most the same size. You will see how the "flow of the game" changes as they trying to move the larger items through the team. They almost always bring up this "problem" and then we can have a discussion of what it would be like to have all the work the same size. Sometimes, the teams will even ask if they can do something with the large paper wad. I leave it to their discretion and they usually try to make it a similar size to the other paper wads.

Figura 13 - Indicação de um jogo por um participante do grupo AgileGames (HANOULLE, 2013).

A *wiki* Games for Scrum (SAHOTA, 2009) apresenta em uma mesma página, uma lista sobre jogos para o ensino de Scrum, contendo o nome e, em alguns casos, uma referência para o *site* em que o jogo está hospedado (figura 14).

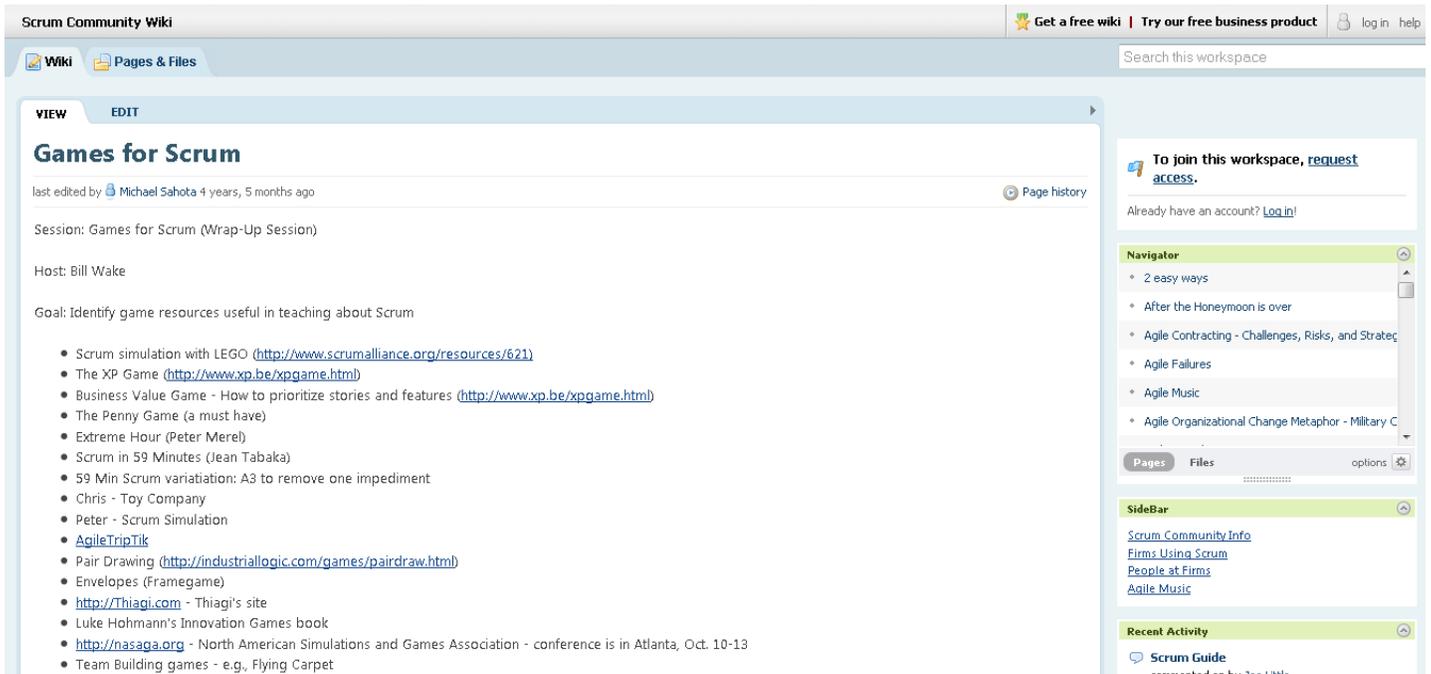


Figura 14 - Lista de jogos na página da *wiki* Games for Scrum (SAHOTA, 2009).

Com as características e ilustrações apresentadas é possível observar, de forma superficial, as funcionalidades apresentadas por cada um dos sistemas considerados. Para possibilitar uma análise concreta, os requisitos levantados a priori são comparados com as funcionalidades encontradas, através da análise na subseção a seguir.

### 3.4.2. Análise da conformidade com os requisitos levantados

A conformidade das funcionalidades disponibilizadas nos sistemas TastyCupCakes, IGR, AgileGames e Games for Scrum, com os requisitos levantados, junto com o formato e a arquitetura do sistema, são apresentados na tabela 16.

		<b>TastyCupCakes (MCCOULLOUGH; MCGREAL, 2013)</b>	<b>IGR (GQS, 2013)</b>	<b>AgileGames (HANOUILLE, 2013),</b>	<b>Games for Scrum (SAHOTA, 2009)</b>
<b>Características</b>	Formato	<i>Blog</i> (Wordpress)	Repositório de objetos de aprendizagem	Grupo de discussão (Google)	<i>Wiki</i> (PbWorks)
	Arquitetura	Índices centralizados, objetos com referência externa	Índices centralizados, objetos com referência externa	-----	-----
<b>Requisitos</b>	REQ01 - Exibição dos jogos	Semiestruturada. Com imagens e textos.	Estruturada. Com imagens, classificação, descrição dos objetivos de aprendizagem e descrição da execução.	Livre. Nome do jogos, descrição breve e referência à <i>site</i> externo para obtenção do jogo e de mais informações.	Livre. Nome do jogo e referência a <i>site</i> externo para obtenção do jogo e de mais informações (em alguns casos, apenas).
	REQ02 - Busca de jogos	Textual simples. O termo da busca é procurado nos campos de todos os jogos.	Estruturada. Com os campos de nome, área de domínio, classificação do jogo, duração, objetivo de aprendizagem, características do grupo em que o jogo pode ser aplicado.	Textual simples. O termo da busca é procurado nos textos de todos os tópicos do grupo.	-----
	REQ03 - Adição, modificação e remoção de objetos de aprendizagem	Semiestruturada. Moderada. Sem necessidade de identificação do usuário. O jogo passa por aprovação para ser aceito no <i>site</i> .	Estruturada. Moderação para jogos de outros usuários. Necessita de identificação. O usuário pode manipular apenas os jogos que adicionou.	Adição de tópicos e respostas de forma livre. Necessita de identificação. O usuário pode manipular as conversas que criou.	Edição livre da lista de jogos. Necessita de identificação.
	REQ04 - Sistema de avaliação	Pontuação de zero a cinco, sem critério.	Pontuação de zero a cinco, com campo para justificar.	-----	-----
	REQ05 - Controle de acesso	Cadastro de usuário moderado.	Cadastro de usuário livre. Revogação de acesso efetuada pelo moderador	Cadastro de usuário dos grupos do Google, autorizações efetuadas pelo dono do grupo. O usuário pode ser banido caso ofenda outro usuário ou publique material inadequado.	Cadastro de usuário da <i>wiki</i> , autorizações e revogações efetuadas pelo moderador.

Tabela 16 – Requisitos considerados e características gerais dos “repositórios” avaliados

Após a análise dos repositórios encontrados, em que o *blog* TastyCupCakes.org apresenta parcialmente as características esperadas, com exceção do uso padronizado de metadados, e o repositório IGR que, ao desconsiderar o requisito de jogo para o ensino do Scrum, qualifica-se como resultado que mais se aproximou dos requisitos levantados. Os

outros dois resultados encontrados não apresentaram funcionalidades que cumprissem com os requisitos.

Com a flexibilização do requisito do domínio (jogos para ensino do Scrum), o IGR apresenta os requisitos esperados para um repositório, candidatando-se à adaptação para suportar a busca e a apresentação de informações do Scrum. Esta adaptação é realizada através da agregação dos metadados do IGR com os metadados propostos neste trabalho (chamados de modelo SGR, Scrum Games Repository), disponibilizados no Apêndice A.

## **4. Solução desenvolvida**

Neste capítulo são abordadas a especificação dos metadados, com a definição e validação do conjunto de metadados a ser utilizado, a especificação dos requisitos, que contém os requisitos funcionais e não-funcionais, os perfis de acesso e os casos de uso, a arquitetura e modelagem do sistema, a descrição do padrão para as interfaces do usuário, a implementação do sistema e os testes realizados nesta implementação.

### **4.1. Especificação dos metadados dos jogos**

#### **4.1.1. Análise para validação dos metadados propostos**

Nesta seção são apresentados os requisitos utilizados na análise e validação dos metadados propostos no presente trabalho.

Na fundamentação teórica sobre Gerenciamento Ágil de projetos, Scrum, Aprendizagem e Jogos, foi possível perceber características que podem ser aplicadas na busca e na classificação de jogos. Estes grupos de classificação que farão parte dos metadados utilizados para referenciar os jogos no repositório são sumarizados a seguir:

- Informações gerais;
- Características da versão;
- Classificações gerais do jogo;
- Classificações em relação à metodologia ágil Scrum;
- Classificações de aprendizagem;
- Execução do jogo;
- Avaliação;
- Comentários.

#### **Critérios de inclusão e exclusão considerados na validação**

Os limites sobre localização da busca, amplitude, tema e linguagens aceitas, foram

estabelecidos na tentativa de manter a objetividade e eficácia, e são listados a seguir:

- **Tema:** a busca está limitada aos jogos para o ensino do Scrum ou dos princípios do gerenciamento ágil que suportem o Scrum.
- **Resultados avaliados:** os cinco primeiros jogos encontrados serão avaliados.
- **Localização:** sistema de buscas de páginas da Web da Google.com. Os *sites* que não estão indexados no sistema da Google.com não são avaliados.
- **Amplitude:** a análise dos resultados encontrados considera os 100 primeiros resultados. Esta análise será feita em até dois níveis de profundidade, sendo que o primeiro nível é representado pelos resultados diretos e o segundo nível é representado por possíveis referências internas encontradas na página avaliada.
- **Ordenação:** os resultados estão na ordem das páginas mais referenciadas, ou que são referenciadas por outras páginas relevantes (com muitas referências), para as menos referenciadas, de acordo com o mecanismo Page Rank da Google.com.
- **Restrições:** os resultados que, pela identificação do título ou pela breve descrição disponibilizada pela ferramenta Google, se mostrarem inadequados, não serão avaliados. Esta restrição estende-se à jogos que não possuem uma descrição suficiente para o entendimento básico do plano de execução.
- **Linguagem:** apenas os resultados que se encontrarem na língua portuguesa ou na língua inglesa serão avaliados.

## **Execução da busca**

Utilizando o sistema Google.com, foi realizada uma busca, no dia 13 de agosto de 2013, com os seguintes parâmetros: [(*jogos OR games*) AND *Scrum*].

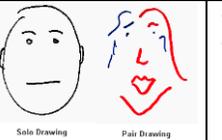
Foram encontrados aproximadamente 5.400.000 resultados compatíveis com os termos da busca, indexados através do mecanismo de Page Rank da Google.com.

Ao realizar a busca nos registros resultantes, os seguintes jogos foram encontrados: *Scrum From Hell* (WAKE, 2004), *The Scrum Game* (WAKE; COHN, 2007), *PairDraw* (KERIEVSKY, 2001), *Agile TripTik* (SEFFERNICK, 2008), *Scrum Simulation with LEGO Briks*

(KRIVITSKY, 2011).

## Classificação dos jogos encontrados no modelo de metadados propostos

A distribuição das informações referentes à caracterização, execução e avaliação dos jogos nos metadados propostos, é exibida na tabela 17.

Informações gerais	Nome	<i>Scrum From Hell</i>	<i>The Scrum Game</i>	<i>PairDraw</i>	<i>AgileTripTik</i>	<i>Scrum Simulation with LEGO Bricks</i>
	Imagem	 (extraído de Kruchten, 2010)			(Não disponível)	
	Criador	William Wake	William Wake e Mike Cohn	Joshua Kerievsky	Tom Seffernick	Alexey Krivitsky
	Referência de publicação	(Não disponível)	(Não disponível)	(Não disponível)	(Não disponível)	(Não disponível)
	Palavras-chave	Scrum, Reunião de acompanhamento diário, Daily Meeting, Distração	Scrum, Reunião de revisão, Reunião de acompanhamento diário	Programação em par, trabalho em equipe	Priorização, planejamento	Planejamento, trabalho em equipe, LEGO.
Características da versão	Data da publicação	20/10/2004	28/04/2007	03/07/2001	13/05/2008	28/02/2009
	Site	<a href="http://xp123.com/articles/scrum-from-hell/">http://xp123.com/articles/scrum-from-hell/</a>	<a href="http://www.mountaingoatsoftware.com/pages/28-the-scrum-game-a-fun-interactive-tool-for-learning-scrum-amp-agile">http://www.mountaingoatsoftware.com/pages/28-the-scrum-game-a-fun-interactive-tool-for-learning-scrum-amp-agile</a>	<a href="http://www.industriallogic.com/blog/pairdraw-2/">http://www.industriallogic.com/blog/pairdraw-2/</a>	<a href="http://scrumcommunity.pbworks.com/w/page/10148889/AgileTripTik">http://scrumcommunity.pbworks.com/w/page/10148889/AgileTripTik</a>	<a href="http://www.scrumalliance.org/system/resources/files/0000/3689/Scrum-Simulation-with-LEGO-Bricks-v2.0.pdf">http://www.scrumalliance.org/system/resources/files/0000/3689/Scrum-Simulation-with-LEGO-Bricks-v2.0.pdf</a>
	Material Necessário	<ul style="list-style-type: none"> <li>• Vinte cartas para responder às questões da Reunião de acompanhamento diário;</li> <li>• Dez cartas para comportamentos inadequados.</li> </ul>	<ul style="list-style-type: none"> <li>• Tabuleiro do jogo;</li> <li>• Cartas de impedimento;</li> <li>• Cartas de oportunidade;</li> <li>• Um pino para cada jogador;</li> <li>• Dois dados de movimentação;</li> <li>• Três dados de progresso.</li> <li>•</li> </ul>	<ul style="list-style-type: none"> <li>• Bloco de folhas no formato A4;</li> <li>• Canetas de cores diferentes, uma para cada participantes.</li> </ul>	<ul style="list-style-type: none"> <li>• Mapa com pontos de parada;</li> <li>• Uma carta para cada ponto de parada;</li> <li>• Bloco de notas;</li> <li>• Canetas.</li> </ul>	<ul style="list-style-type: none"> <li>• Um baralho de cartas do Planning Poker para cada participante;</li> <li>• <i>Post-its</i> para escrever as histórias e as estimativas</li> <li>• Cartaz branco para estruturar o <i>Sprint Backlog</i> e colocar os <i>Post-its</i></li> <li>• Conjuntos de peças de lego para construir as estruturas</li> </ul>
Gerenciamento	Princípios	Trabalho em	Trabalho em	Trabalho em	Trabalho em	Trabalho em equipe,

ágil de projetos		equipe, conversação direta, auto-organização	equipe, conversação direta.	equipe, conversação direta, adaptação à mudanças	equipe, conversação direta, adaptação à mudanças	conversação direta, adaptação à mudanças
Objetivos de Aprendizagem	Cognitivo	Análise, síntese e avaliação	Conhecimento e compreensão	Compreensão, análise e avaliação	Compreensão e análise	Conhecimento, compreensão e aplicação
	Afetivo	Recepção e Resposta	Resposta	Recepção e resposta	Recepção e resposta	Recepção e resposta
	Psicomotor	Percepção e Adaptação	Percepção	Adaptação	Adaptação	Percepção e adaptação
Scrum	Papéis	Scrum Master e Time	Time	Time	Time	<i>Product Owner, Scrum Master</i> e Time
	Competências	Conhecimento sobre o processo, comunicação, colaboração e envolvimento	Conhecimento sobre o processo, comunicação, colaboração e envolvimento	Colaboração e envolvimento	Colaboração e envolvimento	Conhecimento sobre o processo, comunicação, colaboração e envolvimento
	Fases	Game	Game	Relacionado à Game	Game e Postgame	Pregame e Game
	Cerimonias	Reunião diária de acompanhamento	Reunião diária de acompanhamento e revisão do produto	----	Revisão do produto	Reunião de planejamento, reunião diária de acompanhamento e revisão do produto
	Artefatos	Quadro de tarefas	Gráfico de acompanhamento e lista de funcionalidades (estimativa)	----	Lista de funcionalidades (estimativa)	Lista de funcionalidades
Jogos	Categoria	Simulação	Sorte	Simulação	Simulação	Simulação
	Público alvo	Ensino superior e corporativo	Ensino superior e corporativo	Ensino superior	Ensino superior e corporativo	Ensino superior e corporativo
	Quantidade de jogadores	Até oito jogadores	Até oito jogadores	Até quatro jogadores	Até oito jogadores	Mais de oito jogadores
	Relação entre jogadores	Cooperativa e competitiva	Cooperativa	Cooperativa	Cooperativa	Cooperativa e competitiva
	Duração	Curta	Longa	Curta	Curta	Longa
	Espaço físico	Médio	Pequeno	Pequeno	Médio	Médio - Grande

Tabela 17 – Distribuição das informações dos jogos encontrados dos metadados propostos.

Ao analisar os jogos encontrados, foi constatado que os metadados levantados atendem as necessidades de caracterização e classificação de jogos no domínio do Scrum. Para que a descrição de um jogo seja considerada completa são necessários outros metadados, como de avaliação por objetivos educacionais e de pontuação por alunos e professores.

O sistema desenvolvido no presente trabalho tem como base o repositório IGR, que contempla os metadados faltantes na análise realizada nesta seção.

#### 4.1.2. Definição dos metadados e comparação com outros modelos

A descrição geral de um jogo (objeto de aprendizagem), com informações que possibilitem a identificação e distinção entre os demais, é realizada com o auxílio dos metadados do grupo de Metadados gerais (tabela 18).

Para identificar o jogo, visualmente, os modelos do IGR e do SGR propõem a utilização de um metadado para armazenar uma imagem dos componentes do jogo (humano e material). Este metadado não é previsto inicialmente no modelo LOM, mas pode ser criado através da especificação de metadados utilizando a categoria de Meta-metadados [3]<sup>9</sup>. O mesmo processo pode ser realizado na criação do metadado

Apesar da falta de um metadado para esta representação visual, o modelo LOM especifica outros metadados que são considerados excedentes para o presente trabalho, listados a seguir:

- Identificador [1.1]: utilizado para referenciar objetos de outros sistemas de catalogação, não é especificada no SGR, porque não existem repositórios ou catálogos de jogos para Scrum (segundo a pesquisa realizada no presente trabalho).
- Descrição [1.4]: utilizado para descrever o objeto sumariamente. Não é especificado no SGR pela dificuldade e custo na criação deste tipo de informação, para considerar a semântica de todos os elementos do objeto de aprendizagem (informações contidas na descrição completa do jogo [5.10]).
- Cobertura [1.5]: utilizado para descrever o contexto histórico, cultural e geográfico em que o jogo foi criado. Como os jogos do SGR são para o ensino do Scrum, e esta é uma metodologia recente, ligada ao gerenciamento de projetos e desenvolvimento de *software*, este metadado não é especificado no modelo do presente trabalho.
- Estrutura [1.7] e nível de agregação [1.8]: utilizados para descrever a estrutura organizacional (atômica e coleção, por exemplo) e a granularidade do objeto. Como todos os objetos do SGR são jogos, atômicos e de alta granularidade, os metadados para armazenar estas informações tornam-se desnecessários.

---

<sup>9</sup> Número de identificação da categoria, sub categoria ou elemento (metadado) do modelo LOM (IEEE, 2002).

1. Metadados gerais				
Metadado			Tipo de campo e possíveis valores no SGR	Exemplos
SGR	LOM	IGR (GQS, 2013)		
Imagem	--	Imagem	Campo para envio de imagem dos componentes ou da execução do jogo	
Referência de publicação	Descrição; metadados gerais [1.4] <sup>10</sup>	Referência de publicação	Campo texto com a referência à uma publicação em artigo ou conferência, relacionada ao jogo	<p>“Fernandes JM, Sousa SM; PlayScrum - a card game to learn the Scrum agile method, 2nd International Conference on Games and Virtual Worlds for Serious Applications (VS-GAMES 2010), Braga, Portugal, IEEE Computer Society Press, pp. 52-59” ou</p> <p>“<a href="http://sbgames.org/sbgames2010/proceedings/computing/short/Computing_short35.pdf">http://sbgames.org/sbgames2010/proceedings/computing/short/Computing_short35.pdf</a>”</p>
Nome	Título; metadados gerais [1.2]	Nome	Campo texto com o nome do jogo	“Scrum Simulation with Lego Briks”
Línguas disponíveis	Língua; metadados gerais [1.3]	Línguas disponíveis	Campo de seleção para as línguas em que o jogo é distribuído: <ul style="list-style-type: none"> <li>• Inglês;</li> <li>• Português;</li> <li>• Francês;</li> <li>• Espanhol;</li> <li>• Russo;</li> <li>• Alemão;</li> <li>• Italiano;</li> <li>• Outros</li> </ul>	“Português” e “Inglês”
Palavras-chave	Palavras-chave; metadados gerais [1.5]	Palavras-chave	Campo texto com as palavras que identificam os assuntos abordados no jogo	“Scrum”; “Gerenciamento de projetos”

Tabela 18 - Grupo de metadados sobre informações gerais.

Uma das características básicas dos objetos de aprendizagem é a possibilidade de

<sup>10</sup> A descrição de uma referência de publicação no LOM, pode ser feita no metadado de Descrição (metadados gerais [1.4]), se tiver caráter informativo, apenas; pode ser feita no metadado Identificador (metadados gerais [1.1]), se for utilizado para identificar o objeto em um sistema de classificação; e pode ser feita com a especificação de um novo metadado, por meio do grupo de meta-metadados [3].

reutilização de parte ou de sua totalidade. Com a utilização por outros instrutores ou em um contexto diferente do que serviu como base para a criação, o objeto pode precisar de adaptações, que devem ser associadas ao objeto original. Para possibilitar o rastreamento deste grupo de informações sobre o ciclo de vida do objeto foram criados os metadados da categoria 2 do modelo LOM (IEEE, 2002).

Os jogos, por possuírem características educativas (e lúdicas) intrínsecas no mecanismo de funcionamento, não possuem esta facilidade de customização. Um jogo modificado pode não cumprir mais com seus objetivos de aprendizagem (definidos anteriormente), ou pode necessitar de um tempo ou espaço físico maior para executá-lo. Por este motivo o modelo SGR não disponibiliza metadados para controlar a versão [2.1], o estado do objeto [2.2], considerando-o como final no momento em que for registrado no repositório, e as informações dos contribuintes [2.3]. Para referenciar um jogo modificado, o SGR sugere a criação de um novo registro de objeto de aprendizagem.

Os metadados do SGR utilizados para o início do ciclo de vida dos objetos de aprendizagem, a criação, pode ser encontrados na tabela 19.

2. Metadados de criação do objeto				
Metadado			Tipo de campo e possíveis valores no SGR	Exemplos
SGR	LOM (IEEE)	IGR (GQS, 2013)		
<b>Criador</b>	Entidade; contribuinte; metadados do ciclo de vida [2.3.2]	Criador	Campo texto com o nome do criador	“João da Silva”
<b>Data da publicação</b>	Data; contribuinte; metadados do ciclo de vida [2.3.3]	--	Dia em que o cadastro do jogo foi realizado	“10/10/2010”

Tabela 19 – Grupo de metadados sobre informações da criação do objeto.

O modelo LOM disponibiliza o grupo de Meta-metadados [3] para adicionar metadados não especificados no modelo original. Os metadados de Imagem e Referências de publicação do SGR (no grupo de metadados gerais), por exemplo, poderiam fazer parte de um meta-modelo, entretanto, devido à dificuldade em desenvolver e manter um repositório baseado em uma abstração deste nível, a especificação do grupo de Meta-metadados não é utilizada.

Para especificar as características técnicas (digitais) do objeto de aprendizagem, o modelo LOM disponibiliza o grupo de metadados técnicos [4], com especificação para armazenar as seguintes informações:

- formato [4.1]: vídeo/avi, áudio/mp3 ou text/pdf, por exemplo;
- tamanho [4.2]: expresso em bytes, “1048576 bytes” é o tamanho descrito para um arquivo de 1mb);
- requisitos [4.4]: requisitos que compreendem destas configurações de *hardware* e sistema operacional, até versão do navegador e de aplicativos.
- descrição da instalação [4.5]: por exemplo, “ descompacte o ‘arquivo.rar’ e execute o arquivo ‘video-aula.mpeg’ localizado dentro da pasta descompactada”;
- outros requisitos [4.6]: informações sobre outros componentes de *hardware* e *software*, que não podem ser expressos através do metadado 4.4; e
- duração [4.7]: tempo de execução contínua do objeto digital, utilizada, principalmente para recursos multimídia, como uma vídeo-aula (com duração de 1h30), por exemplo.

Como estes metadados se referem, principalmente, a objetos no formato digital, e o SGR contém apenas jogos manuais (não virtuais), somente dois dos metadados do grupo técnico do LOM são utilizados, apresentados na tabela 20.

4. Metadados técnicos				
SGR	Metadado		Tipo de campo e possíveis valores no SGR	Exemplos
	LOM (IEEE)	IGR (GQS, 2013)		
<b>Formato</b> <sup>11</sup>	Formato; metadados técnicos [4.1]	Formato <sup>12</sup>	Campo de seleção para o formato do jogo: <ul style="list-style-type: none"> <li>• Baseado em computador;</li> <li>• Manual</li> </ul>	“Manual”
<b>Site</b>	Localização; metadados técnicos [4.3]	<i>Site</i>	Campo texto com a referência ao <i>site</i> onde o jogo pode ser encontrado	“ <a href="http://www.gqs.ufsc.br/detective-game-what-killed-the-project/">http://www.gqs.ufsc.br/detective-game-what-killed-the-project/</a> ”

Tabela 20 – Grupo de metadados sobre informações técnicas

A descrição mais completa das características físicas e de execução de um objeto de aprendizagem no SGR, pode ser encontrada nos metadados do grupo educacional [5], do

<sup>11</sup> Embora o presente trabalho considere apenas jogos manuais, o metadado e os valores de “Formato” foram mantidos para viabilizar a adaptação ao repositório IGR.

<sup>12</sup> Embora o presente trabalho considere apenas jogos manuais, o metadado e os valores de “Formato” foram mantidos para viabilizar a adaptação ao repositório IGR.

LOM. Parte deste modelo de referência não é utilizada, porque os jogos, utilizados no SGR, possuem algumas características em comum, não havendo necessidade de diferenciação através dos metadados.

- tipo de interação [5.1]: “mista”, “expositiva” ou “ativa”, onde os jogos, como objetos de aprendizagem experiencial, utilizam-se deste último tipo de interação;
- grau de interação [5.3]: muito alto para jogos;
- densidade semântica [5.4]: muito alta para jogos, que reúnem diversos elementos, educacionais e técnicos, em um mesmo recurso;
- público alvo [5.5]: os alunos em geral, ou aprendizes, são o público alvo dos jogos do SGR;
- contexto [5.6]: nos jogos para ensino do Scrum, a informação descritiva sobre o ambiente onde o objeto é utilizado é desnecessária; mas se considerarmos como semântica o nível educacional do público alvo, ela é utilizada, no grupo de metadados para classificação [9];
- faixa etária [5.7]: a restrição de idade para o ensino do Scrum está mais relacionada à necessidade de conhecimento do aluno sobre gerenciamento de projetos, do que em relação à conteúdo ofensivo; portanto, o aluno que tiver este conhecimento (aos 12, 18 ou 30 anos, por exemplo) estará apto à participação dos jogos do SGR;
- duração [5.9]: para simplificar a apresentação dos metadados no SGR, a utilização do tempo de duração expresso em número discreto (dias, horas, minutos e segundos) é dispensado, em função do metadado para classificação de restrições [9], onde a utilização de intervalos de tempo possibilita, ao instrutor, uma escolha de acordo com as restrições de tempo de sua aula ou curso; e
- língua [5.11]: os jogos do SGR são destinados à alunos que entendam a língua em que o este foi construído; não constando nos propósitos do repositório a utilização de jogos para o ensino de línguas (um jogo em inglês, sendo aplicado à um público com pouca proficiência na língua, para que aprendam o Scrum e o inglês, por exemplo).

Um dos metadados do grupo educacional do LOM, para a descrição de características, instruções, avisos, e outras informações pertinentes à execução do jogo [5.10], é transformado, no modelo do SGR, em outros cinco metadados, cada qual com uma função específica. Este e outros metadados utilizados no SGR são apresentados na tabela 21.

5. Metadados educacionais				
SGR	Metadado		Tipo de campo e possíveis valores no SGR	Exemplos
	LOM (IEEE)	IGR (GQS, 2013)		
<b>Tipo</b>	Tipo de recurso de aprendizagem; metadados educacionais [5.2]	Tipo	Campo de seleção para o tipo de jogo <sup>13</sup> : <ul style="list-style-type: none"> <li>• Competição;</li> <li>• Sorte;</li> <li>• Simulação;</li> <li>• Dramatização;</li> <li>• Jogo de tabuleiro;</li> <li>• Estudo de caso</li> <li>• Outros.</li> </ul>	“Competição”
<b>Complexidade<sup>14</sup></b>	Dificuldade; metadados educacionais [5.8]	Complexidade	Campo de seleção para a complexidade do jogo: <ul style="list-style-type: none"> <li>• Alta;</li> <li>• Média;</li> <li>• Baixa</li> </ul>	“Alta”
<b>Customizações disponíveis</b>	Descrição; metadados educacionais [5.10]	Customizações disponíveis	Campo texto com a descrição de customizações possíveis para o jogo	“Utilizando mais um baralho, é possível adicionar cerca de 2 pessoas em cada grupo”
<b>Presença do instrutor</b>	Descrição; metadados educacionais [5.10]	Presença do instrutor	Campo de seleção para indicar a necessidade do instrutor na execução: <ul style="list-style-type: none"> <li>• Necessária</li> <li>• Opcional</li> </ul>	“Necessária”
<b>Descrição</b>	Descrição; metadados educacionais [5.10]	Descrição	Campo de texto para a descrição geral do jogo	“Os participantes, em grupos de seis pessoas, vão criar um <i>Product Backlog...</i> ”
<b>Passos básicos</b>	Descrição; metadados educacionais [5.10]	Passos básicos	Campo de texto para a descrição e para a estimativa de duração de cada passo da execução	<ul style="list-style-type: none"> <li>• Explicação inicial – 15 min.</li> <li>• Realização das estimativas – 5min.</li> <li>• Reunião de planejamento – 5 min.</li> </ul>
<b>Comentários para o aluno</b>	Descrição; metadados	Comentários para o aluno	Campo de texto para questionamento,	A conclusão da execução serve para

<sup>13</sup> Os tipos de jogos especificados no IGR foram mantidos com o intuito de possibilitar a adaptação do com o modelo do SGR. Os valores possíveis para este metadado são os valores conjuntos dos modelos, alterando a semântica dos tipos de “Competição”, “Sorte” e “Simulação”, para que não considerem mais jogos de tabuleiros.

<sup>14</sup> A complexidade está diretamente relacionada à dificuldade de utilização de um jogo, considera-se que um jogo de complexidade baixa é fácil de usar.

	educacionais [5.10]		anotações e dicas para a execução do jogo.	refletir na utilização do Scrum no lugar de outras metodologias de projeto.
--	---------------------	--	--	---

Tabela 21 - Grupo de metadados sobre classificações gerais do jogo.

A construção de um repositório de jogos para o ensino do Scrum tem como objetivo facilitar o acesso de instrutores e alunos, a estes recursos. Isso não significa que todos os jogos poderão ser utilizados livremente. Para preservar os direitos e especificar os eventuais custos de compra ou utilização de um jogo, o modelo LOM especifica um grupo de metadados sobre direitos autorais [6]. No SGR, considera-se que um metadado para indicar (com valores “sim” ou “não”) se direitos autorais e restrições de uso são aplicados ao objeto, é desnecessário. Diferentemente do indicativo, o metadado para descrição [6.3] destas informações é utilizado. Estes metadados, sobre direitos autorais, e exemplos de valores podem ser encontrados na tabela 22.

6. Metadados sobre direitos autorais				
Metadado			Tipo de campo e possíveis valores no SGR	Exemplos
SGR	LOM (IEEE)	IGR (GQS, 2013)		
<b>Disponibilidade</b>	Custo; metadados sobre direitos autorais [6.1]	Disponibilidade	Campo de seleção para a disponibilidade do jogo: <ul style="list-style-type: none"> <li>• Gratuito;</li> <li>• Pago</li> </ul>	“Gratuito”
<b>Custo</b>	Descrição; metadados sobre direitos autorais [6.3]	Custo	Campo texto com o valor (expresso em US\$) referente ao custo de compra do jogo ou dos materiais para desenvolvê-lo	“R\$50,00”
<b>Licença</b>	Descrição; metadados sobre direitos autorais [6.3]	Licença	Campo de seleção para indicar o tipo de licença: <ul style="list-style-type: none"> <li>• COPYRIGHT</li> <li>• CC BY</li> <li>• CC BY-SA</li> <li>• CC BY-ND</li> <li>• CC BY-NC</li> <li>• CC BY-NC-SA</li> <li>• CC BY-NC-ND</li> </ul>	“Copyright” e “Creative Commons”

Tabela 22 – Grupo de metadados sobre direitos autorais

O custo e os direitos autorais mencionados anteriormente podem ser aplicados apenas à parte de um objeto de aprendizagem. Se este objeto for composto por outros, e um deles (um baralho de cartas, por exemplo), tem restrições de uso; considera-se, nos metadados do grupo 6, os direitos autorais mais restritivos com o custo somado.

As relações entre objetos de aprendizagem são especificadas no grupo de metadados

sobre relacionamento [7] do modelo LOM. A utilização destes metadados é útil para repositórios onde são armazenadas figuras, vídeos, textos, manuais, e sobretudo, objetos de alta granularidade. Os jogos para o ensino do Scrum não fazem parte deste cenário, são compostos por partes indissociáveis, não reutilizáveis, o que elimina a necessidade de especificar metadados para o relacionamento entre objetos.

Os comentários para avaliar as partes ou o todo de um objeto, realizados por alunos ou professores (que não tenham utilizado de um método científico) são uteis para entender por outra perspectiva, se os objetivos estão sendo alcançados por meio da diferença nos resultados obtidos através da aplicação dos métodos formais de avaliação e dos comentários dos alunos.

Para pessoas que desejam contribuir com este tipo de informações, o modelo LOM disponibiliza o grupo de metadados para anotação [8] que, no caso do SGR, é estendido através da inclusão de um metadado para pontuação do jogo em que o comentário está associado (tabela 23).

8. Metadados para anotação				
SGR	Metadado		Tipo de campo e possíveis valores no SGR	Exemplos
	LOM (IEEE)	IGR (GQS, 2013)		
<b>Autor</b>	Entidade; metadados para anotação [8.1]	Autor	Campo com o nome do usuário que realizou o comentário.	"Paulo"
<b>Data</b>	Data, metadados para anotação [8.2]	Data	Data da criação do comentário.	"01/01/2013"
<b>Texto</b>	Descrição; metadados para anotação [8.3]	Texto	Campo de texto para a descrição do comentário.	"Jogo bem elaborado e intuitivo. A aplicação dele com os alunos do Curso de computação foi excelente!"
<b>Pontuação individual</b>	--	Pontuação individual	Campo de seleção para indicar uma nota de avaliação (uma a cinco estrelas).	★ ★ ★

Tabela 23 - Grupo de metadados sobre comentários.

Os metadados apresentados anteriormente servem para descrever o objeto em seu formato digital ou físico, descrever a execução e os pontos a serem observados, ou adicionar comentários relevantes ao objeto. Estas informações podem auxiliar na escolha de um jogo,

mas não serão decisivas, para isso o instrutor precisa de que os jogos sejam classificados de acordo com objetivos de aprendizagem, restrições (tempo e espaço) e características conceituas.

Na tabela 24 são apresentados os metadados herdados do modelo IGR, utilizados para parte da classificação no SGR, onde cada metadado deste grupo reúne as informações de identificação [9.1] e taxionomia [9.2] do modelo LOM, no mesmo registro. Esta modelagem foi adotada a fins de diminuir a complexidade do SGR, e por este mesmo motivo, os metadados de descrição [9.3] e palavras-chave [9.4] foram suprimidos.

9.1 Metadados para classificação geral				
Metadado			Tipo de campo e possíveis valores	Exemplos
SGR	LOM (IEEE)	IGR (GQS, 2013)	SGR	
<b>Domínio</b>	(Classificação de disciplina) metadados para classificação [9]	Domínio	Campo de seleção para o domínio da comunidade: <ul style="list-style-type: none"> <li>• “Ciência da computação”;</li> <li>• “Engenharia de <i>Software</i>”;</li> <li>• “Administração”;</li> <li>• “Outros”</li> </ul>	“Engenharia de <i>Software</i> ”
<b>Área da computação</b> <sup>15</sup>	(Classificação de conceito) metadados para classificação [9]	Área da computação	Campo de seleção para a área da computação: <ul style="list-style-type: none"> <li>• “Gerenciamento de projetos”;</li> <li>• “Integração de sistemas”;</li> <li>• “Sistemas inteligentes”;</li> <li>• “Análise de requisitos de negócio”;</li> <li>• ... (outras áreas)</li> </ul>	“Gerenciamento de projetos”
<b>Objetivos do domínio cognitivo</b>	(Classificação de objetivo educacional) metadados para classificação [9]	Objetivos do domínio cognitivo	Campo de seleção com os objetivos do domínio cognitivo, apresentados no jogo: <ul style="list-style-type: none"> <li>• Conhecimento;</li> <li>• Compreensão</li> <li>• Aplicação</li> <li>• Análise;</li> <li>• Síntese</li> <li>• Avaliação</li> </ul>	“Conhecimento”
<b>Duração por grupo</b>	(Classificação de restrições) metadados para classificação [9] <sup>16</sup>	Duração por grupo	Campo de seleção para a duração do jogo por grupo de participantes: <ul style="list-style-type: none"> <li>• 0 – 1/2 hora;</li> <li>• 1/2 hora – 1 hora;</li> <li>• 1 – 2 horas;</li> <li>• 2 – 4 horas;</li> <li>• 4 – 6 horas;</li> <li>• 6 – 8 horas;</li> <li>• Mais de 8 horas;</li> </ul>	“2-4 horas”
<b>Quantidade</b>	(Classificação)	Quantidade de participantes	Campo de seleção para a quantidade	“Três ou

<sup>15</sup> Embora os jogos encontrados no SGR sejam, principalmente, do domínio da Engenharia de Software, e da área de gerenciamento de projeto, os metadados e os valores do “Domínio” e da “Área da Computação” foram mantidos para viabilizar a adaptação ao repositório IGR.

<sup>16</sup> O tempo de duração do objeto de aprendizagem pode ser catalogado de duas formas distintas dependendo de seu valor: através dos metadados de classificação se for um número intervalar, com semântica restritiva e classificatória; e através dos metadado educacionais se for um número discreto, com semântica descritiva

de participantes por grupo	de pré-requisitos metadados para classificação [9]	por grupo		de participantes por grupo: <ul style="list-style-type: none"> <li>• um;</li> <li>• dois;</li> <li>• três ou quatro;</li> <li>• cinco ou seis;</li> <li>• sete ou oito;</li> <li>• mais de oito</li> </ul>	quatro”
(Não utilizado)	Relação entre os participantes (intra e inter-grupos)	(Não utilizado)	Relação entre os participantes (intra e inter-grupos)	Campo de seleção para a relação entre os participantes: <ul style="list-style-type: none"> <li>• Cooperativa;</li> <li>• Competitiva</li> <li>• Cooperativa e Competitiva</li> <li>• Sem interação entre participantes</li> </ul>	“Cooperativa”
(Não utilizado)	Espaço necessário por grupo	(Não utilizado)	Espaço necessário por grupo	Campo de seleção para o espaço necessário: <ul style="list-style-type: none"> <li>• Pequeno (até 4m<sup>2</sup>);</li> <li>• Médio (até 30m<sup>2</sup>);</li> <li>• Grande (acima de 30m<sup>2</sup>)</li> </ul>	“Pequeno (até 4 m <sup>2</sup> )”
Habilidades interpessoais / Atitudes	Objetivos do Domínio Afetivo	Habilidades interpessoais / Atitudes	Objetivos do Domínio Afetivo	Campo de seleção com os objetivos do domínio afetivo, apresentados no jogo: <ul style="list-style-type: none"> <li>• Recepção;</li> <li>• Resposta</li> <li>• Valorização</li> <li>• Organização</li> <li>• Internalização de valores</li> </ul>	“Recepção”
Habilidades interpessoais / Atitudes	Objetivos do Domínio Psicomotor	Habilidades interpessoais / Atitudes	Objetivos do Domínio Psicomotor	Campo de seleção com os objetivos do domínio psicomotor, apresentados no jogo: <ul style="list-style-type: none"> <li>• Percepção;</li> <li>• Preparação;</li> <li>• Resposta orientada;</li> <li>• Resposta mecânica;</li> <li>• Resposta completa evidente; ou</li> <li>• Reorganização.</li> </ul>	“Percepção”

Tabela 24 - Grupo de metadados sobre classificações de aprendizagem.

Os metadados apresentados anteriormente são para classificações em geral e não trazem abordam conceitos específicos dos jogos para o ensino do Scrum. Os metadados para esta classificação, presentes apenas no modelo SGR, são apresentados na tabela 25.

9.2 Metadados para classificação de acordo com a metodologia Scrum				
SGR	Metadados		Tipo de campo e possíveis valores no SGR	Exemplos
	LOM (IEEE)	IGR (GQS, 2013)		
Princípios ágeis	(Classificação de conceito) metadados para classificação [9]	--	Campo de seleção para o princípio ágil abordado no jogo (BECK et al, 2001): <ul style="list-style-type: none"> <li>• Atendimento à solicitações do cliente;</li> <li>• Favorecimento a mudanças;</li> <li>• Entregas de <i>software</i> frequentes;</li> <li>• Desenvolvimento em equipe;</li> <li>• Ambiente motivador;</li> <li>• Conversação direta;</li> <li>• Funcionamento do <i>Software</i> como métrica;</li> <li>• Desenvolvimento constante e</li> </ul>	“Atendimento à solicitações do cliente”,

			<ul style="list-style-type: none"> <li>sustentável;</li> <li>• Excelência técnica;</li> <li>• Simplicidade;</li> <li>• Times auto organizados;</li> <li>• Melhoria contínua.</li> </ul>	
<b>Papéis do Scrum</b>	(Classificação de conceito) metadados para classificação [9]	--	Campo de seleção para os papéis do scrum abordados no jogo: <ul style="list-style-type: none"> <li>• <i>Scrum Master</i>;</li> <li>• <i>Product Owner</i>;</li> <li>• Time de desenvolvimento</li> </ul>	"Scrum Master"
<b>Fases do Scrum</b>	(Classificação de conceito); metadados para classificação [9]	--	Campo de seleção para as fases do Scrum abordadas no jogo (SCHWABER, 1996, p. 12): <ul style="list-style-type: none"> <li>• Pregame;</li> <li>• Game;</li> <li>• Postgame</li> </ul>	"Pregame"
<b>Cerimoniais do Scrum</b>	(Classificação de conceito) metadados para classificação [9]	--	Campo de seleção para os cerimoniais do Scrum abordados no jogo (RUBIN, 2012): <ul style="list-style-type: none"> <li>• Reunião de planejamento;</li> <li>• Reunião de controle diário;</li> <li>• Reunião de revisão;</li> <li>• Reunião de retrospectiva</li> </ul>	"Reunião de planejamento"
<b>Artefatos do Scrum</b>	(Classificação de conceito) metadados para classificação [9]	--	Campo de seleção para os artefatos do Scrum utilizados no jogo (COHN, 2010): <ul style="list-style-type: none"> <li>• <i>Product backlog</i></li> <li>• <i>Sprint backlog</i></li> <li>• Quadro de tarefas</li> <li>• Gráfico de <i>burndown</i>.</li> <li>• Sumário da retrospectiva</li> </ul>	"Product backlog"
<b>Competências do Scrum</b>	(Classificação de competências) metadados para classificação [9]	--	Campo de seleção para as competências do Scrum desenvolvidas com o jogo (RUBIN, 2012; COHN, 2010): <ul style="list-style-type: none"> <li>• Conhecimento do negócio;</li> <li>• Conhecimento do processo do Scrum;</li> <li>• Comunicação e influência;</li> <li>• Poder e autoridade;</li> <li>• Disponibilidade;</li> <li>• Responsabilidade;</li> <li>• Humildade;</li> <li>• Colaboração;</li> <li>• Comprometimento;</li> <li>• Confiança;</li> <li>• Melhoria contínua;</li> <li>• Simplicidade;</li> <li>• Envolvimento</li> </ul>	"Conhecimento do negócio"

Tabela 25 – Grupo de metadados sobre classificações em relação à metodologia ágil Scrum.

Um dos grandes benefícios do uso de jogos como objetos de aprendizagem é o aproveitamento posterior de resultados obtidos através de um método de avaliação, aplicado à um grupo de alunos. Um jogo avaliado através de uma metodologia científica, com resultados negativos, pode ser dispensado em função de outro jogo com uma avaliação melhor. Para armazenar as informações sobre avaliações realizadas em um jogo, o modelo IGR utiliza os metadados apresentados na tabela 26. Apesar do modelo LOM não

apresentar, explicitamente, metadados para avaliação, é possível adicioná-los ao modelo base, utilizando o grupo de Meta-metadados [3].

10. Metadados para avaliação					
Metadado			Tipo de campo e possíveis valores		Exemplos
SGR	LOM (IEEE)	IGR (GQS, 2013)	IGR (GQS, 2013)	SGR	
<b>Avaliação realizada</b>	--	Avaliação realizada	Campo de seleção para indicar se a avaliação foi realizada antes de cadastrar o jogo: <ul style="list-style-type: none"> <li>• Sim</li> <li>• Não</li> </ul>		“Sim”
<b>Foco da avaliação</b>	--	Foco da avaliação	Campo de texto para descrever o objetivo da avaliação		“Eficiência” e “Usabilidade”
<b>Tipo de Avaliação</b>	--	Tipo de Avaliação	Campo de seleção para indicar qual o tipo de avaliação foi realizada: <ul style="list-style-type: none"> <li>• Experimental;</li> <li>• Quasi-experimental;</li> <li>• Estudo de caso;</li> <li>• Outros.</li> </ul>		“Experimental”
<b>Nível da avaliação</b>	--	Nível da avaliação	Campo de texto para indicar o nível considerado na avaliação (segundo modelo de Kirpatric)		“Nível 1”
<b>Quantidade de alunos envolvidos</b>	--	Quantidade de alunos envolvidos	Campo de seleção para indicar a quantidade de alunos envolvidos no processo de avaliação: <ul style="list-style-type: none"> <li>• 0 – 10 pessoas;</li> <li>• 10 – 50 pessoas;</li> <li>• 50 – 100 pessoas;</li> <li>• Mais de 100 pessoas.</li> </ul>		“0 – 10 pessoas”
<b>Principais resultados</b>	--	Principais resultados	Campo de texto para descrever os principais resultados obtidos na avaliação		“O resultado da aplicação do jogo foi positivo ao reforçar o conhecimento nos processos do Scrum.”

Tabela 26 - Grupo de metadados sobre avaliação.

Este grupo de metadados apresentado, do modelo SGR, especifica as informações necessárias para encontrar e identificar jogos para o ensino do Scrum, de acordo com características descritivas, como a descrição da execução; classificatórias, como duração e objetivos educacionais; e qualitativas, no caso dos resultados da aplicação de uma avaliação.

O modelo SGR não é especificado para ser estático, final; ao contrário, espera-se que a comunidade de prática, em volta do repositório, identifique e desenvolva melhorias que possam ser aplicadas ao modelo inicial.

No tópico a seguir são apresentados os requisitos baseados nas funcionalidades

necessárias para um repositório e na apresentação e busca de jogos através dos metadados mencionados anteriormente.

## 4.2. Especificação dos requisitos e casos de uso

Ao analisar o estado da arte em repositórios, ficou evidenciado o cumprimento dos requisitos gerais por parte do *Instructional Games Repository* (GQS, 2013). Para que este repositório pudesse ser utilizado, os critérios de busca e as informações exibidas na página de apresentação do jogo, foram adaptados para trabalhar com os metadados propostos pelo presente trabalho.

Os requisitos mapeados no *Instructional Games Repository*, por Bonetti (2011) foram atualizados para contemplar esta modificação.

No tópico a seguir, é apresentada uma análise comparativa entre os metadados das categorias do modelo de referência LOM (IEEE, 2012), com os metadados do modelo do IGR e do SGR. Os metadados do modelo LOM descritos na análise, são referenciados pelo nome, seguido da categoria (e sub categorias) em que está contido, e pelo número de identificação.

### 4.2.1. Requisitos funcionais

Nesta subseção são apresentados os requisitos essenciais para o repositório de jogos para o ensino do Scrum.

Os requisitos necessários para a criação de uma comunidade de prática, como descrita no capítulo da fundamentação teórica, não são abordados neste trabalho. O repositório pode servir como uma ferramenta para uma comunidade, embora não ofereça meios de integração.

Identificação	Nome	Descrição
RF01	Busca e visualização	Busca avançada por parâmetros relacionados aos metadados e visualização das informações

RF02	Manipulação de jogos	Criação, modificação e remoção de jogos
RF03	Pontuação e comentários	Atribuição de pontos para indicar a qualidade de um jogo e de comentários para fornecer uma análise textual sobre o jogo.
RF04	Controle de acesso	Restrições para que usuários comuns não possam comprometer a qualidade do sistema e para que usuários avançados possam monitorá-los.
RF05	Cadastro de usuário	Criação, modificação e remoção dos usuários

Tabela 27 – Requisitos funcionais

#### 4.2.2. Requisitos não-funcionais

Com o intuito de manter a qualidade do sistema, em relação ao seu desenvolvimento e funcionamento, foram atualizados os requisitos levantado por Bonetti (2011), para o repositório IGR, descritos na tabela 28.

Identificação	Categoria	Descrição
RNF01	Linguagem de programação e armazenamento de dados	Desenvolvimento com a linguagem Oracle Java 6, para o SGBD Oracle MySQL.
RNF02	Paradigma de programação	Orientado à objetos
RNF03	Documentação	Descrição das classes através da documentação Java. Descrição das funcionalidades através dos casos de uso.
RNF04	<i>Design</i> de interface do usuário	Utilização do tema de elementos visuais do grupo GQS – UFSC.

RNF05	Desempenho	O tempo de processamento do cadastro de usuário e da busca, cadastro, remoção e edição de jogos, deve ser de, no máximo, 5 segundos.
-------	------------	--

Tabela 28 – Requisitos não funcionais do SGR.

A utilização do repositório SGR é realizada por grupos de usuários que contemplam níveis de interesses, habilidades e envolvimento distintos. Essa classificação de perfis é descrita em seguida.

#### **4.2.3. Perfis de usuários**

O SGR apresenta três perfis de usuários com permissões distintas para cadastro, edição e visualização de usuários e jogos. O formato de distribuição de permissões é cumulativo, sendo que o usuário possui todas as permissões que um visitante, além de outras particulares de seu perfil; e o administrador, por sua vez, possui todas as permissões de um usuário comum, além de permissões específicas de administração da comunidade (figura 15).



Figura 15 – Perfis e permissões de usuário no SGR.

Com a definição dos perfis de usuário do sistema, é possível detalhar as funcionalidades e os passos necessários para realizá-las. Este detalhamento é feito por meio dos casos de uso, descritos a seguir.

#### 4.2.4. Casos de uso

A partir da especificação dos requisitos funcionais é realizada a identificação dos casos de uso para os usuários. Como o repositório não possui interação ativa com outros sistemas ou usuários<sup>17</sup> foram identificados apenas os usuários finais da aplicação *Web*.

Os casos de uso foram divididos em dois grupos: os requisitos que podem ser realizados sem a necessidade de controle de usuários (autenticação) e os que precisam deste controle; conforme descrito no diagrama de casos de uso na figura 16.

<sup>17</sup> Ao considerar que a interação do SGBD é passiva, que apenas realiza operações solicitadas pela aplicação, este sistema não pode ser considerado como um ator nos casos de uso.

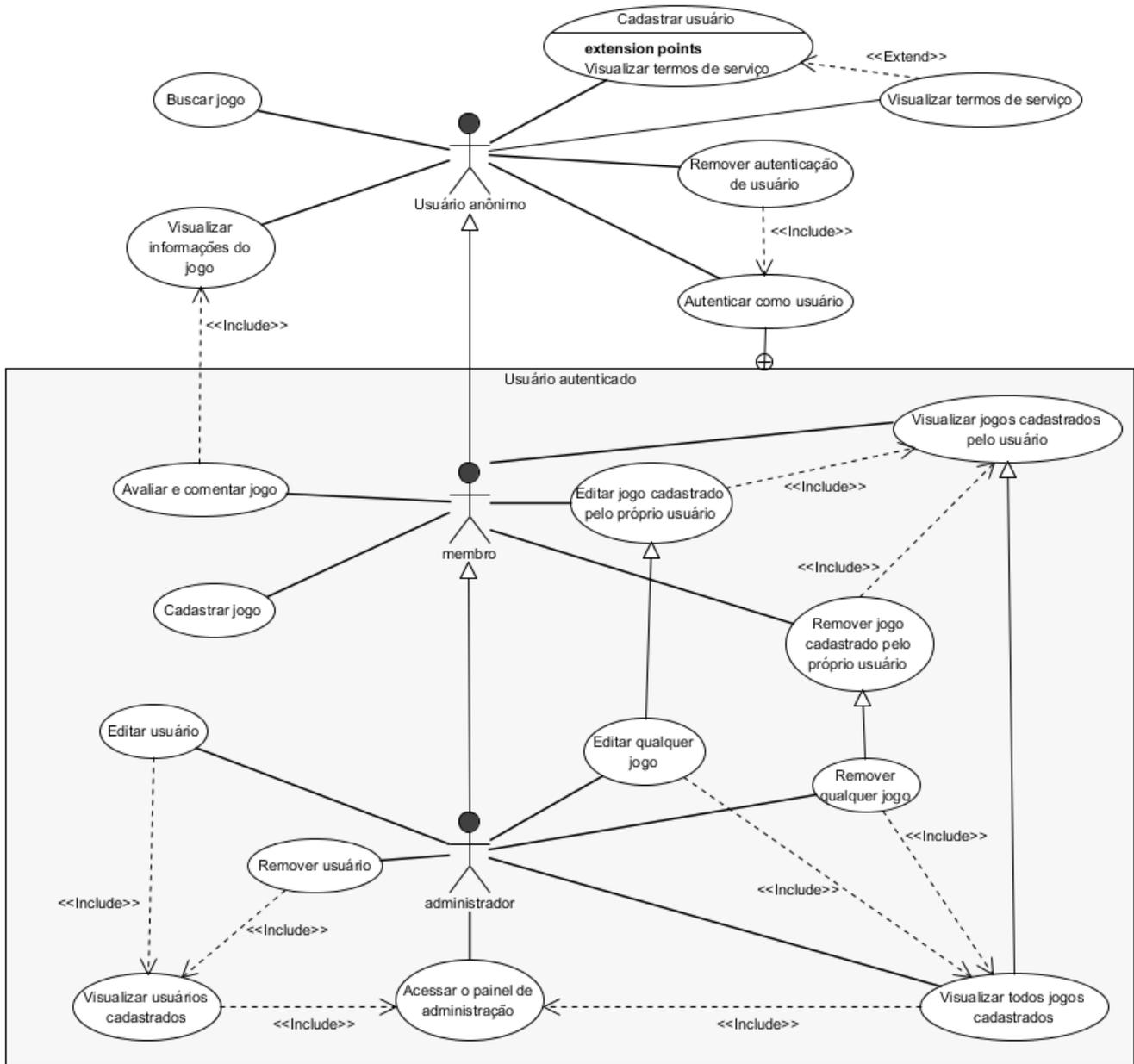


Figura 16 - Atores e casos de uso do SGR

Para descrever os detalhes dos casos de uso, é utilizada a estrutura proposta por Cockburn (2000), que sugere a descrição do ator primário, de pré-condições (inclusive a adição de um caso de uso precedente), a descrição dos passos do cenário principal e a descrição de passos adicionais. Alguns itens sugeridos na estrutura da Cockburn não são utilizados, como o nível, porque todos os casos de uso são do nível de detalhamento de usuário; os interessados, que são os usuários do repositório, e as garantias mínimas e de sucesso, porque nenhuma funcionalidade precisa de aprovação para ser concretizada. Os

detalhes dos casos de uso levantados são descritos a seguir:

### Caso de uso 1 - buscar jogo

**Ator primário:** usuário anônimo, membro ou administrador.

**Pré-condições:** nenhuma.

**Cenário principal:**

1. o usuário preenche os filtros de busca;
2. o usuário clica no botão de pesquisa (“*SEARCH*”); e
3. o sistema exibe a página com os jogos que cumprem com os requisitos filtrados.

**Extensões:**

- 1a. o usuário não precisa preencher os filtros de busca se estiver em busca de uma lista com todos os jogos, podendo pular o passo 1.

### Caso de uso 2 - visualizar informações do jogo

**Ator primário:** usuário anônimo, membro ou administrador.

**Pré-condições:**

- 1a. o usuário deve estar na página de visualização dos jogos por ele cadastrados; ou
- 1b. o usuário deve estar na página de resultados da busca.

**Cenário principal:**

1. o usuário clica no botão (“*MORE*”) para visualizar as informações; e
2. o sistema redireciona o usuário para a página de informações do jogo;

**Extensões:** nenhuma.

### Caso de uso 3 – cadastrar usuário

**Ator primário:** usuário anônimo ou administrador.

**Pré-condições:** nenhuma.

**Cenário principal:**

1. o usuário clica no botão para criar uma nova conta (“*CREATE ACCOUNT*”);
2. o sistema direciona o usuário para o formulário de criação de conta, que contém informações como nome, e-mail, nome de usuário e senha;
3. o usuário preenche estas informações;

4. o usuário clica no botão “*Terms of Service*”, para acessar os termos de serviço;
5. o usuário marca o botão que indica concordância com os termos de serviço;
6. o usuário clica no botão para confirmar a criação da conta (“*REGISTER*”); e
7. o sistema direciona o usuário para a página de autenticação.

**Extensões:**

1a. para que uma nova conta possa ser criada por um administrador, ele deve estar autenticado, precisa estar na página de administração do sistema e deve clicar no botão “*REGISTER*”, dentro do grupo de contas de usuário (“*ACCOUNTS*”); e

5a. caso o usuário deixe o botão de concordância com os termos de serviço desmarcado, ao clicar no botão de confirmação da criação da conta, o sistema exibe uma informação de que a concordância é necessária.

**Caso de uso 4 - visualizar termos de serviço**

**Ator primário:** usuário anônimo, membro ou administrador.

**Pré-condições:** nenhuma.

**Cenário principal:**

1. o usuário clica no botão para acessar a página com os termos de serviço; e
2. o sistema direciona o usuário para esta página.

**Extensões:**

1a. o usuário pode acessar os termos de serviço por meio da página de cadastro de jogos, ao clicar no respectivo botão.

**Caso de uso 5 - autenticar como usuário**

**Ator primário:** usuário anônimo, membro ou administrador.

**Pré-condições:** nenhuma.

**Cenário principal:**

1. o usuário clica no botão para autenticar no repositório (“*LOGIN*”);
2. o sistema direciona o usuário para um formulário com as informações necessárias para a autenticação;

3. o usuário preenche as informações de autenticação;
4. o usuário clica no botão para confirmar a autenticação (“*LOGIN*”); e
5. o sistema direciona o usuário para a página inicial

**Extensões:** nenhuma.

#### **Caso de uso 6 - remover autenticação de usuário**

**Ator primário:** usuário anônimo, membro ou administrador.

**Pré-condições:** o usuário precisa estar autenticado.

**Cenário principal:**

1. o usuário clica no botão para remover a autenticação no repositório (“*LOGOUT*”);  
e
2. o sistema direciona o usuário para a página inicial.

**Extensões:** nenhuma.

#### **Caso de uso 7 - avaliar e comentar jogo**

**Ator primário:** membro ou administrador.

**Pré-condições:**

1. o usuário precisa estar autenticado; e
2. o usuário deve estar visualizando as informações de um jogo;

**Cenário principal:**

1. o usuário clica no botão para avaliar e pontuar o jogo (“*Want to rate or comment this game? Click Here.*”);
2. o sistema exibirá um painel com campos para pontuar e comentar o jogo;
3. o usuário preenche os campos de pontuação e comentário;
4. o usuário salva a pontuação e comentário clicando no botão de confirmação (“*Confirm*”); e
5. o sistema exibe a página de informações do jogo com a pontuação e comentários realizados.

**Extensões:** nenhuma.

#### **Caso de uso 8 - cadastrar jogo**

**Ator primário:** membro ou administrador.

**Pré-condições:** o usuário precisa estar autenticado.

**Cenário principal:**

1. o usuário clica no botão para cadastrar o jogo (“REGISTER A GAME”);
2. o usuário preenche as informações relacionadas ao jogo;
3. o usuário clica no botão para confirmar o cadastro (“CONFIRM”); e
4. o sistema direciona o usuário para a página de exibição do jogo.

**Extensões:**

- 1a. o administrador pode cadastrar um jogo através do painel de administração, clicando no botão “REGISTER”; e
- 2a. nem todas as informações presentes no formulário precisam ser preenchidas.

### **Caso de uso 9 - editar jogo cadastrado pelo próprio usuário**

**Ator primário:** membro ou administrador.

**Pré-condições:**

1. o usuário precisa estar autenticado; e
2. o usuário deve estar na página de visualização dos jogos por ele cadastrados.

**Cenário principal:**

1. o usuário seleciona o jogo que deseja editar apertando no respectivo botão (“EDIT”);
2. o usuário altera as informações que deseja;
3. o usuário confirma a edição, clicando no respectivo botão (“CONFIRM”); e
4. o sistema direciona o usuário para a página de exibição do jogo.

**Extensões:** nenhuma.

### **Caso de uso 10 - visualizar jogos cadastrados pelo próprio usuário**

**Ator primário:** membro ou administrador.

**Pré-condições:** o usuário precisa estar autenticado.

**Cenário principal:**

1. o usuário clica no botão para visualizar os jogos por ele cadastrados (“MY ACCOUNT”); e

2. o sistema apresenta a lista de jogos cadastrados pelo usuário.

**Extensões:** nenhuma.

### **Caso de uso 11 - remover jogo cadastrado pelo próprio usuário**

**Ator primário:** membro ou administrador.

**Pré-condições:**

1. o usuário precisa estar autenticado;
2. o usuário deve estar na página de visualização dos jogos por ele cadastrados; e
3. o sistema apresenta a lista de jogos cadastrados pelo usuário.

**Cenário principal:**

4. o usuário seleciona o jogo que deseja remover apertando no respectivo botão (“*DELETE*”); e
5. o usuário confirma o desejo de remover o jogo, clicando no botão “OK”.

**Extensões:**

- 2a. o usuário pode desistir da remoção do jogo, clicando no botão “Cancel”.

### **Caso de uso 12 - acessar o painel de administração**

**Ator primário:** administrador.

**Pré-condições:** O usuário precisa estar autenticado.

**Cenário principal:**

1. o administrador clica no botão para acessar o painel de administração do repositório (“*ADMINISTRATION*”);
2. o sistema direciona o usuário para o painel de administração.

**Extensões:** nenhuma.

### **Caso de uso 13 - visualizar todos jogos cadastrados**

**Ator primário:** administrador.

**Pré-condições:**

1. o administrador deve estar autenticado; e
2. o administrador deve estar no painel de administração.

**Cenário principal:**

1. o administrador clica no botão para gerenciar os jogos cadastrados, no painel de administração de jogos (“*MANAGEMENT*”); e
2. o sistema exibe uma lista com os jogos cadastrados abaixo do painel de administração.

**Extensões:** nenhuma.

**Caso de uso 14 - remover qualquer jogo**

**Ator primário:** administrador

**Pré-condições:**

1. o administrador deve estar autenticado;
2. o administrador deve estar no painel de administração; e
3. o administrador deve estar visualizando os jogos cadastrados.

**Cenário principal:**

1. o administrador escolhe um jogo e clica no botão para removê-lo (“*DELETE*”);
2. o administrador confirma a ação clicando no botão “*OK*”; e
3. o sistema direciona o administrador para a página de administração.

**Extensões:** nenhuma.

**Caso de uso 15 - editar qualquer jogo**

**Ator primário:** administrador

**Pré-condições:**

1. o administrador deve estar autenticado;
2. o administrador deve estar no painel de administração; e
3. o administrador deve estar visualizando os jogos cadastrados.

**Cenário principal:**

1. o administrador escolhe um jogo e clica no botão para editá-lo (“*EDIT*”);
2. o administrador realiza as alterações desejadas;
3. o administrador confirma a edição, clicando no respectivo botão (“*CONFIRM*”); e
4. o sistema direciona o administrador para a página de exibição do jogo.

**Extensões:** nenhuma.

### **Caso de uso 16 - visualizar usuários cadastrados**

**Ator primário:** administrador.

**Pré-condições:**

1. o administrador deve estar autenticado; e
2. o administrador deve estar no painel de administração.

**Cenário principal:**

1. o administrador clica no botão para gerenciar os usuários cadastrados, no painel de administração de usuários (“*MANAGEMENT*”); e
2. o sistema exibe uma lista com os usuários cadastrados abaixo do painel de administração.

**Extensões:** nenhuma.

### **Caso de uso 17 - remover usuário**

**Ator primário:** administrador

**Pré-condições:**

1. o administrador deve estar autenticado;
2. o administrador deve estar no painel de administração; e
3. o administrador deve estar visualizando os usuários cadastrados.

**Cenário principal:**

1. o administrador escolhe um usuário e clica no botão para removê-lo (“*DELETE*”); e
2. o sistema atualiza a lista de usuários cadastrados.

**Extensões:** nenhuma.

### **Caso de uso 18 - editar usuário**

**Ator primário:** administrador.

**Pré-condições:**

1. o administrador deve estar autenticado;
2. o administrador deve estar no painel de administração; e

3. o administrador deve estar visualizando os usuários cadastrados.

**Cenário principal:**

1. o administrador escolhe um usuário e clica no botão para editá-lo (“EDIT”);
2. o sistema direciona o administrador para a página de edição do usuário;
3. o administrador realiza as alterações desejadas;
4. o administrador clica no botão para confirmar as edições (“EDIT”); e
5. o sistema direciona o administrador para a lista de usuários cadastrados, no painel de administração.

**Extensões:** nenhuma.

Com os detalhes das funcionalidades previstas para o repositório, é possível criar uma arquitetura e uma modelagem de banco de dados que possibilitem o desenvolvimento do sistema de acordo com as melhores práticas e tecnologias existentes. Estas estruturas são descritas na seção a seguir.

### **4.3. Arquitetura e modelagem do sistema**

O repositório desenvolvido no presente trabalho é uma adaptação do IGR (BONETTI, 2011; GQS, 2013), para suportar, principalmente, as operações de busca, criação, edição e visualização de jogos com os conceitos e classificações do Scrum.

Esta adaptação é realizada através de modificação das telas do repositório, adicionando campos e comportamentos visuais; das classes de apresentação, controle e manipulação de dados; e do banco de dados, através da adição de tabelas e relacionamentos, e inserção de massa de dados para o domínio do Scrum. Os detalhes destas modificações são descritos a seguir.

#### **4.3.1. Modelagem objeto-relacional**

O sistema de gerenciamento de banco de dados utilizado é o MySQL 5.1 (objeto-relacional), com tabelas do tipo InnoDB, para suporte a transações e restrições de chaves

estrangeiras, e com a codificação de caracteres Latin1.

A modelagem das objetos e dos relacionamentos é realizada através da ferramenta MySQL Workbench 6.0, utilizando o modelo do IGR como base, obtido com a operação de engenharia reversa. A partir deste modelo extraído, são adicionadas as tabelas e relacionamentos para a manipulação das informações levantadas para o repositório SGR, listados na tabela 29:

Tabela origem (nova)	Tipo de relacionamento	Tabela destino	Valores
"affectivelevel"	N:M através da entidade fraca "affectivlevel_characteristic"	"characteristics"	Valores do nível afetivo da taxionomia de objetivos educacionais
"psychomotorlevel"	N:M através da entidade fraca "psychomotorlevel_characteristic"	"characteristics"	Valores do nível psicomotor da taxionomia de objetivos educacionais
"ceremony"	N:M através da entidade fraca "ceremony_characteristic"	"characteristics"	Cerimoniais do Scrum
"artifact"	N:M através da entidade fraca "artifact_characteristic"	"characteristics"	Artefatos do Scrum
"role"	N:M através da entidade fraca "role_characteristic"	"characteristics"	Papéis do Scrum
"competency"	N:M através da entidade fraca "competency_characteristic"	"characteristics"	Competências do Scrum
"agileprinciple"	N:M através da entidade fraca "agileprinciple_characteristic"	"characteristics"	Princípios das metodologias ágeis
"authorlicense"	N:M através da entidade fraca "authorlicense_resources"	"resources"	Tipos de licença para distribuição de jogos (direitos autorais)
"availablespace"	N:M através da entidade fraca "availablespace_classificationgame"	"classificationgame"	Escala de espaço necessário para que o jogo passa ser executado

Tabela 29 - Tabelas e relacionamentos adicionados no modelo de banco de dados.

A integração das tabelas e relacionamentos adicionados pode ser encontrada na

modelagem que contém os relacionamentos da tabela do jogo (“Method”) com as principais tabelas do banco de dados, na figura 17.

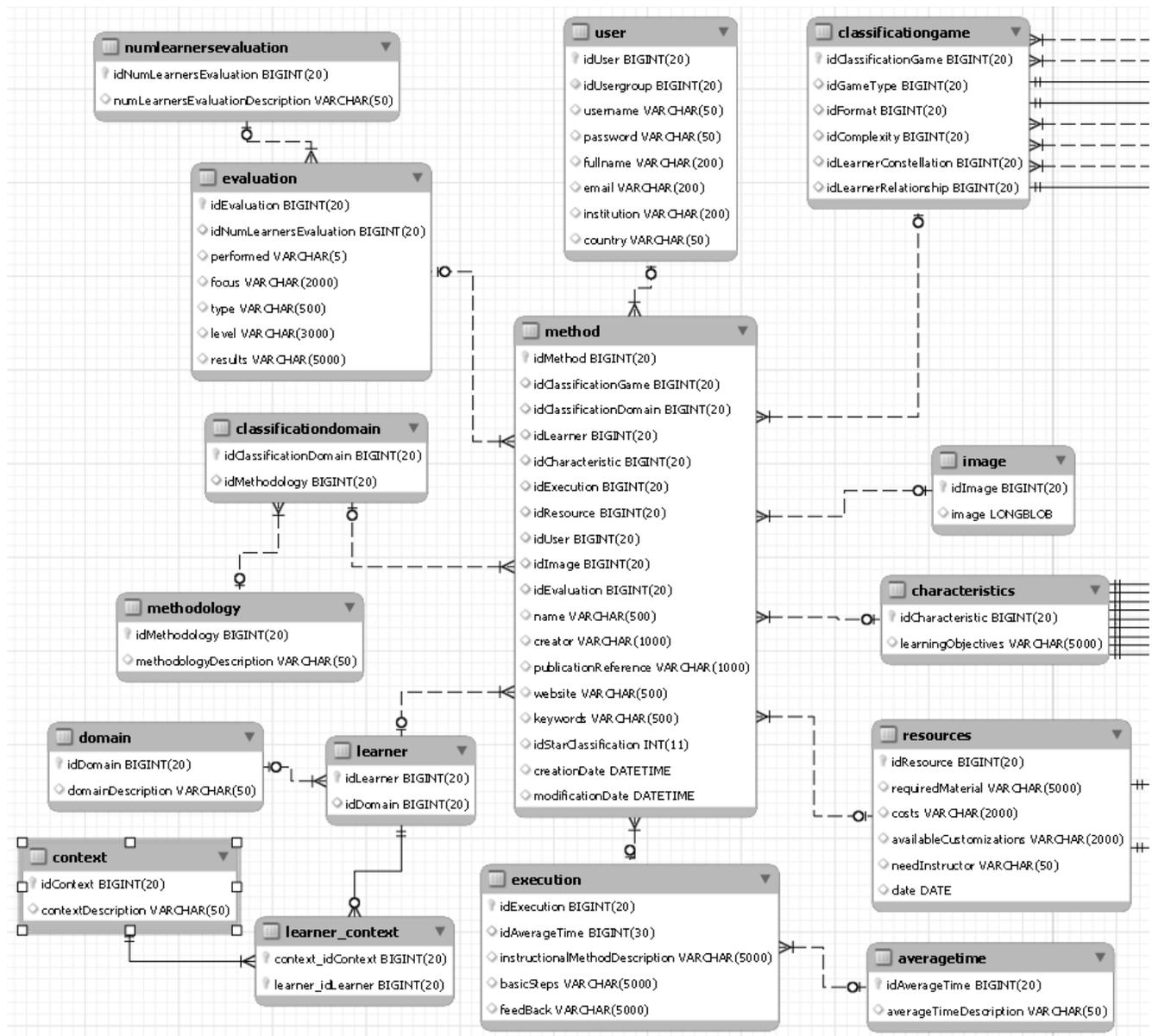


Figura 17 - Principais tabelas e relacionamentos na modelagem objeto-relacional do SGR.

As tabelas e relacionamentos que não estão diretamente ligados a um jogo (tabela “Method”), são descritos nas figuras 18, 19, 20 e 21.

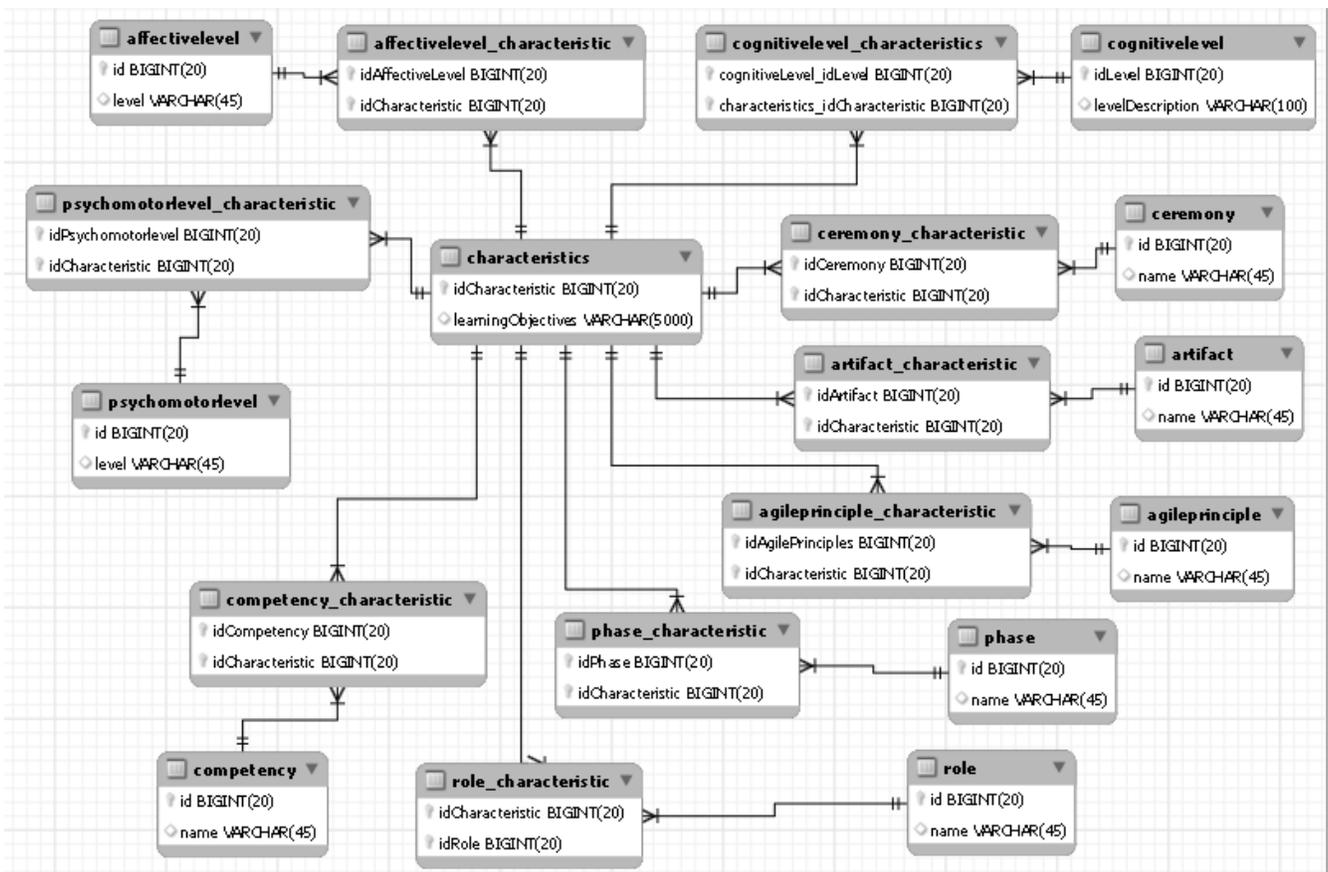


Figura 18 - Tabelas e relacionamentos para descrever as características do jogo na modelagem objeto-relacional do SGR

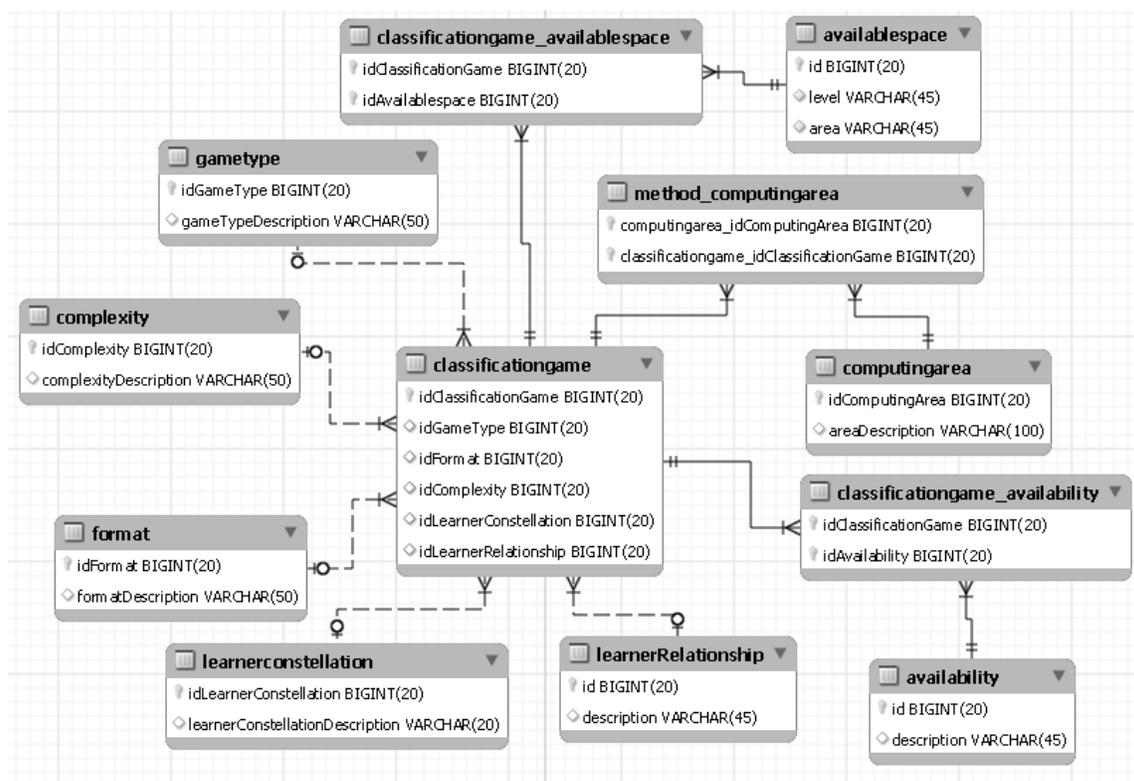


Figura 19 - Tabelas e relacionamentos para classificação de jogos na modelagem objeto-relacional do SGR.

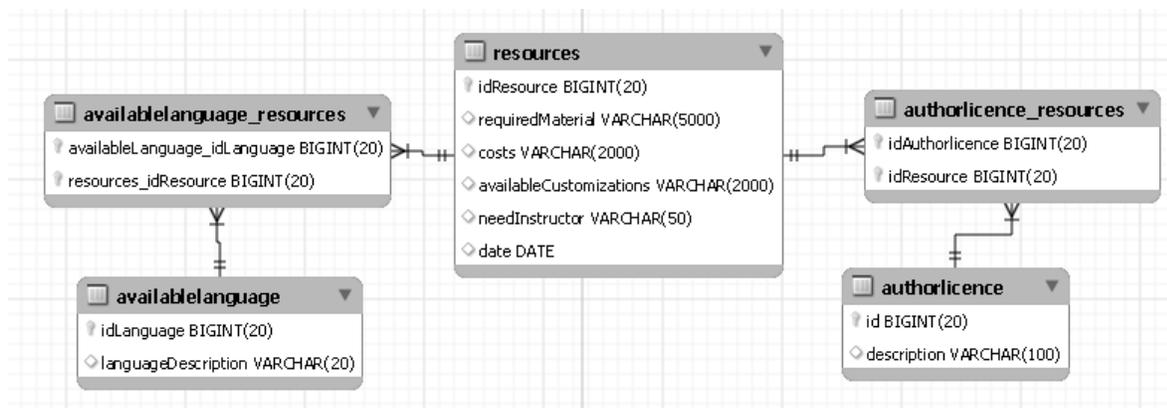


Figura 20 – Tabelas e relacionamentos para descrever os recursos dos jogos na modelagem objeto-relacional do SGR

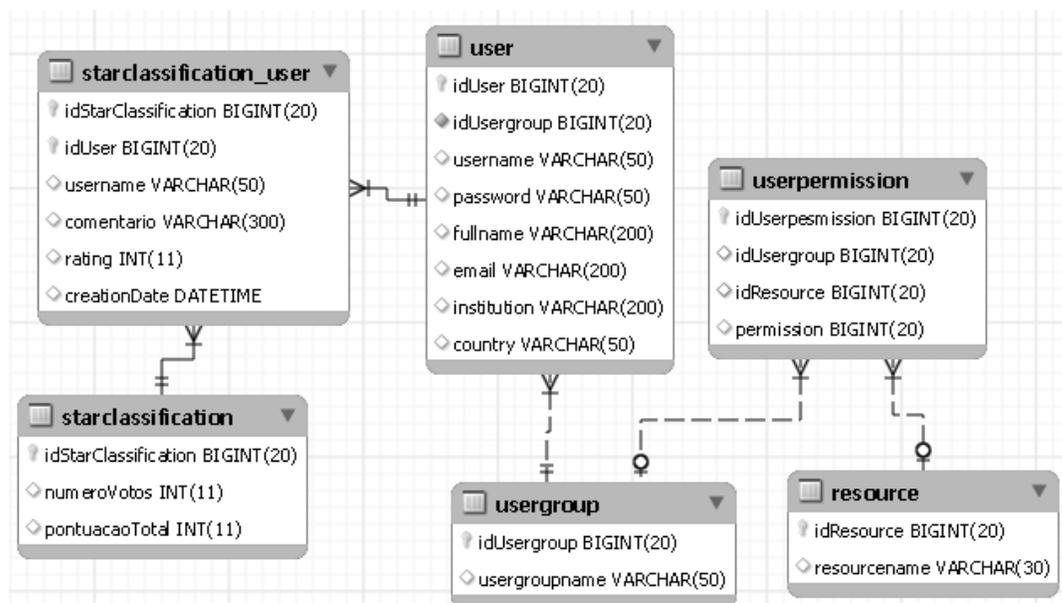


Figura 21 – Tabelas e relacionamentos para caracterização do usuário na modelagem objeto-relacional do SGR

Os registros adicionados nas tabelas, para classificação dos jogos, foram apresentados anteriormente junto com a descrição dos metadados. As modificações do banco de dados atingem não apenas as novas tabelas, mas outras que tiveram seus valores de classificação revistos, como, por exemplo, para tempo médio de duração do jogo (“averagetime”) e para número de jogadores necessários por grupo (“learnerconstellation”).

A integração do modelo e das modificações mencionadas, que compõe os requisitos levantados no projeto, é realizada através da codificação de funções, objetos e métodos de manipulação de dados, descritos a seguir.

#### 4.3.2. Estrutura de classes

Para desenvolver as novas funcionalidades, é utilizada a linguagem Oracle Java, através da Kit de desenvolvimento *Standard Edition*, na versão 5.0 *Update 7*. A utilização de uma versão mais nova da linguagem não é possível, porque a versão citada é a última suportada pelo servidor Apache Tomcat, versão 6.0.9, disponibilizado pelo departamento de Informática e Estatística (UFSC), que serve como servidor de “produção” para o artefato de

software desenvolvido neste trabalho.

O sistema desenvolvido tem seu código organizado em três camadas (MVC): de **apresentação**, para informações de visualização, como JSP, HTML, JavaScript e CSS; de **domínio**, para funções e objetos que trabalham com a lógica das operações; e de **fonte de dados**, para objetos e bibliotecas que realizam a comunicação com o banco de dados (FOWLER, 2003, p. 20, 330).

Nas camadas de domínio e fonte de dados, são utilizados padrões de design considerados como boas práticas, e convenções de programação da linguagem Java, como formato de indentação de código, utilização de chaves para identificar blocos, nome de pacotes e classes, e número máximo de linhas em classes e métodos (SUN MICROSYSTEMS, 1997). Os padrões de design utilizados são listados a seguir:

- *Front Controller*: consolida a conversação entre a parte cliente da aplicação (navegador, HTML e JavaScript) e a parte servidor (código de domínio e fonte de dados, principalmente) canalizando todas as requisições HTTP para que sejam tratadas por um único objeto (FOWLER, 2003, p. 344);
- *Data Access Object (Gateway)*: mantém a lógica de acesso ao banco de dados isolada do sistema; permitindo o acesso de tabelas através de interfaces simples para buscar, salvar, alterar ou excluir dados, que encapsulam operações complexas que consideram chaves e restrições (FOWLER, 2003, p. 144);
- *Value Object*: são objetos simples com poucos atributos e que tem sua igualdade determinada através da comparação entre os seus valores e os valores de outro objeto do mesmo tipo, ignorando qualquer diferença entre a posição em memória (referência) ou identificador artificial (FOWLER, 2003, p. 486);
- *Template Method*: define a estrutura de uma operação com passos a serem seguidos; que são implementados por subclasses, provendo comportamento concreto (GAMMA et al, 1995, p. 325); e
- *Builder*: separa a construção de um objeto complexo de sua representação, para que seja possível customizar o processo de criação e gerar diferentes representações (GAMMA et al, 1995, p. 97).

A aplicação dos padrões de design mencionados pode ser encontrada na classe “br.ufsc.incod.servlet.SGRController.java” (*Front Controller*), em todas as classes do pacote “br.ufsc.incod.dao” (*Data Access Object*), em todas as classes do pacote “br.ufsc.incod.entity” (*Value Object*) e nos construtores de operações SQL, “br.ufsc.incod.util.InsertQueryBuilder”, “br.ufsc.incod.util.UpdateQueryBuilder” e “br.ufsc.incod.util.QueryBuilder” (*Builder* e *Template Method*).

O padrões de código e a modelagem do banco de dados suportam o desenvolvimento das interfaces de usuário e dos fluxos das operações, descritos nos subtópicos a seguir.

#### **4.4. Interfaces de usuário**

As interfaces de usuário foram adaptadas do sistema IGR, que segue o padrão de cores, formas e organização de página estabelecidos para os sistemas desenvolvidos em conjunto com o Grupo de Qualidade de *Software* – INCOD – INE - UFSC (2013).

Utilizando este padrão, o SGR apresenta as páginas de busca, cadastro e informações de jogos e usuários, em um formato contendo os três grupos de elementos visuais listados a seguir:

- cabeçalho [1]: com a identificação do grupo de pesquisa (GQS), do sistema (Scrum Games Repository) e das entidades que suportam seu desenvolvimento (laboratório INCoD e UFSC);
- corpo da página [2]: contém as informações de uma página do repositório, como por exemplo, a busca de jogos por parâmetros de classificação; e
- rodapé [3]: contém informações de localização de outras páginas do repositório, como ajuda, termos de compromisso de uso e informações gerais.

Os grupos de elementos visuais podem ser encontrados na figura 22.

The image shows a web interface for searching educational games. It includes a navigation bar with 'HOME', 'LOGIN', and 'CREATE ACCOUNT' buttons. A search bar is located at the top. Below the search bar, there is a 'Search' section with a form containing several filter categories: 'APPLICATION DOMAIN' (Computing Area), 'GAME CLASSIFICATION' (Format, Type, Availability), 'GAME EXECUTION' (Duration, Language, Required Space), 'LEARNING OBJECTIVE' (Cognitive Level, Affective Level, Psychomotor Level), and 'LEARNERS CHARACTERISTICS' (In Groups of, Interrelationship, Context, Domain). A 'SEARCH' button is positioned at the bottom right of the form. The footer contains contact information for GQS and logos for INCoD and CNPq.

Figura 22 - Grupos de elementos visuais no SGR

## 4.5. Implementação do sistema

A implementação do sistema é realizada utilizando a linguagem de programação Oracle Java Enterprise Edition 6, com um servidor de aplicação Apache Tomcat 7 e banco de dados MySQL.

Os resultados da implementação são apresentados por meio da indicação dos passos do cenário principal, realizada na especificação dos requisitos deste trabalho, nas telas capturadas durante a execução das funcionalidades. Este fluxo dos passos e das telas é apresentado a seguir.

### Caso de uso 1 - buscar jogo

#### Cenário principal:

1. o usuário preenche os filtros de busca;
2. o usuário clica no botão de pesquisa (“SEARCH”); e

3. o sistema exibe a página com os jogos que cumprem com os requisitos filtrados.



Figura 23 – Telas no caso de uso 1 - buscar jogo

## Caso de uso 2 - visualizar informações do jogo

### Cenário principal:

1. o usuário clica no botão ("MORE") para visualizar as informações; e
2. o sistema redireciona o usuário para a página de informações do jogo;

HOME LOGIN CREATE ACCOUNT Scrum From Hell [2]

**Search Results**  
 SEARCH CRITERIA: All methods  
 3 Game(s) Found  
 NEW SEARCH

**Scrum From Hell**  
 COMPUTING AREA: Project Management  
 COGNITIVE LEVELS: Evaluating, Analyzing  
 WEBSITE: <http://xp123.com/articles/scrum-from-hell/>  
 MORE

[1]

**PHOTO**



**INFORMATION**

NAME	Scrum From Hell
CREATOR	William Wake
PUBLICATION REFERENCE	Not Informed
KEYWORDS	scrum, daily meeting, distraction
WEBSITE	<a href="http://xp123.com/articles/scrum-from-hell/">http://xp123.com/articles/scrum-from-hell/</a>

**APPLICATION DOMAIN CLASSIFICATION**

COMPUTING AREA	Project Management
----------------	--------------------

**SCRUM CLASSIFICATION**

AGILE PRINCIPLES	Self-organizing teams Team work
------------------	------------------------------------

Figura 24 – Telas no caso de uso 2 – visualizar informações do jogo

### Caso de uso 3 – cadastrar usuário

#### Cenário principal:

1. o usuário clica no botão para criar uma nova conta (“*CREATE ACCOUNT*”);
2. o sistema direciona o usuário para o formulário de criação de conta, que contém informações como nome, e-mail, nome de usuário e senha;
3. o usuário preenche estas informações;
4. o usuário clica no botão “*Terms of Service*”, para acessar os termos de serviço;
5. o usuário marca o botão que indica concordância com os termos de serviço;
6. o usuário clica no botão para confirmar a criação da conta (“*REGISTER*”); e
7. o sistema direciona o usuário para a página de autenticação.

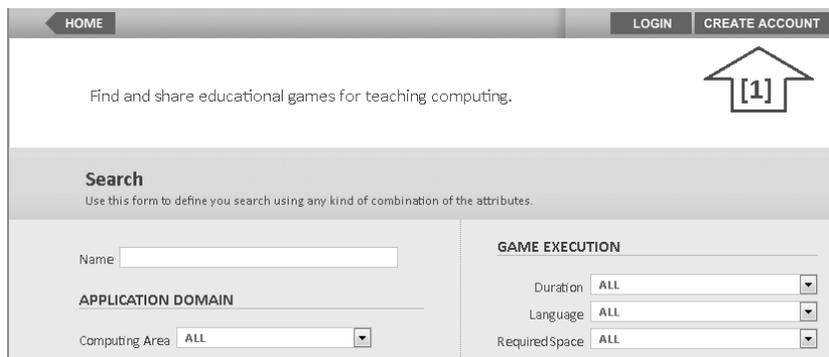


Figura 25 – Tela inicial no caso de uso 3 – cadastrar usuário

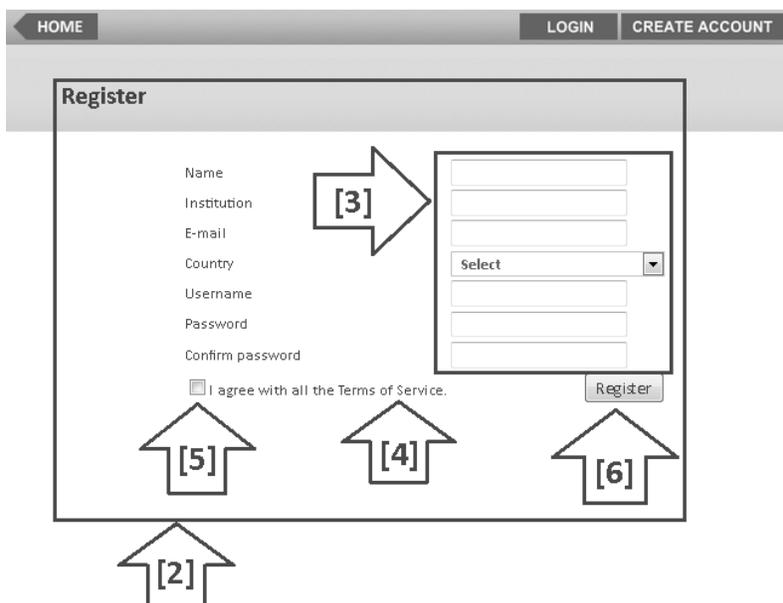


Figura 26 - Tela de cadastro no caso de uso 3 – cadastrar usuário

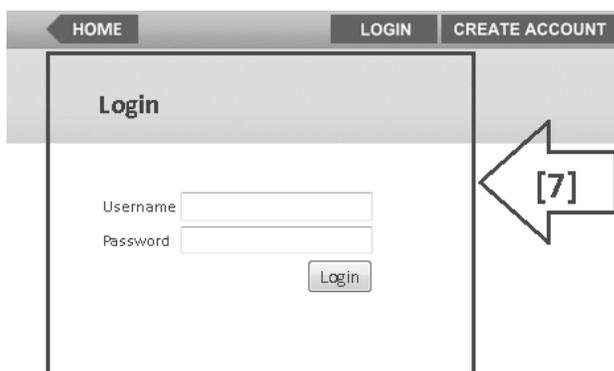
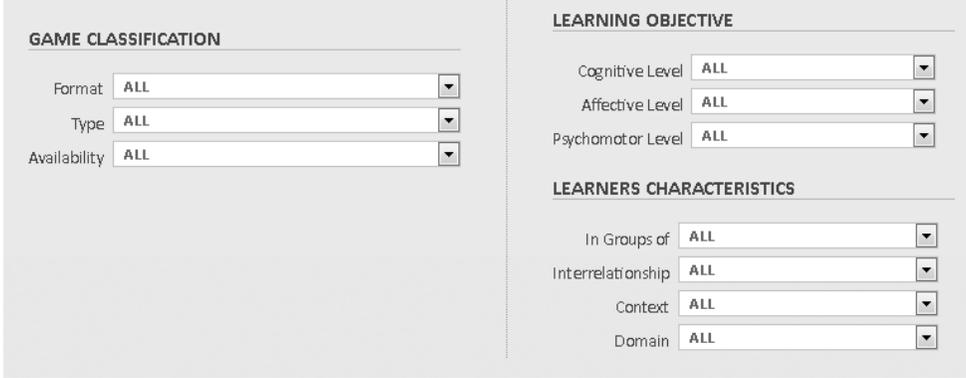


Figura 27 - Tela de autenticação no caso de uso 3 – cadastrar usuário

## Caso de uso 4 - visualizar termos de serviço

## Cenário principal:

1. o usuário clica no botão para acessar a página com os termos de serviço; e
2. o sistema direciona o usuário para esta página.



**GAME CLASSIFICATION**

Format: ALL  
Type: ALL  
Availability: ALL

**LEARNING OBJECTIVE**

Cognitive Level: ALL  
Affective Level: ALL  
Psychomotor Level: ALL

**LEARNERS CHARACTERISTICS**

In Groups of: ALL  
Interrelationship: ALL  
Context: ALL  
Domain: ALL

SEARCH

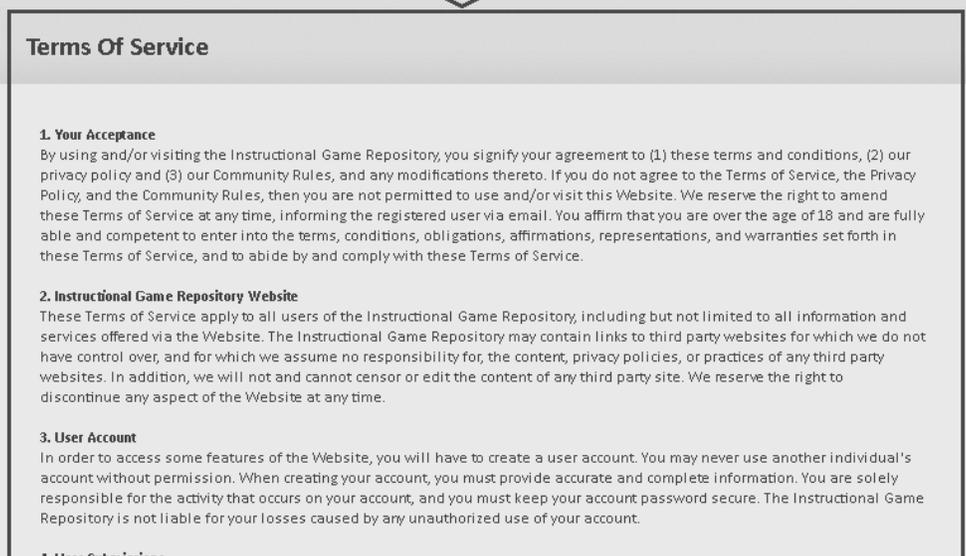
**Software Quality Group - GQS**  
National Institute for Research and Technology on Digital Convergence - INODD  
Department of Informatics and Statistics - INE  
Federal University of Santa Catarina - UFSC  
88049-200 Florianópolis - SC Brazil  
incod@incod.ufsc.br

[1]

SUPPORT  
CNPq  
60 ANOS

Welcome | Community Rules | Privacy Policy | Terms of Service

Figura 28 - Tela inicia no caso de uso 4 – visualizar termos de serviço



[2]

## Terms Of Service

**1. Your Acceptance**  
By using and/or visiting the Instructional Game Repository, you signify your agreement to (1) these terms and conditions, (2) our privacy policy and (3) our Community Rules, and any modifications thereto. If you do not agree to the Terms of Service, the Privacy Policy, and the Community Rules, then you are not permitted to use and/or visit this Website. We reserve the right to amend these Terms of Service at any time, informing the registered user via email. You affirm that you are over the age of 18 and are fully able and competent to enter into the terms, conditions, obligations, affirmations, representations, and warranties set forth in these Terms of Service, and to abide by and comply with these Terms of Service.

**2. Instructional Game Repository Website**  
These Terms of Service apply to all users of the Instructional Game Repository, including but not limited to all information and services offered via the Website. The Instructional Game Repository may contain links to third party websites for which we do not have control over, and for which we assume no responsibility for, the content, privacy policies, or practices of any third party websites. In addition, we will not and cannot censor or edit the content of any third party site. We reserve the right to discontinue any aspect of the Website at any time.

**3. User Account**  
In order to access some features of the Website, you will have to create a user account. You may never use another individual's account without permission. When creating your account, you must provide accurate and complete information. You are solely responsible for the activity that occurs on your account, and you must keep your account password secure. The Instructional Game Repository is not liable for your losses caused by any unauthorized use of your account.

**4. User Submissions**

Figura 29 - Tela dos termos de serviço no caso de uso 4 – visualizar termos de serviço

## Caso de uso 5 - autenticar como usuário

### Cenário principal:

1. o usuário clica no botão para autenticar no repositório (“LOGIN”);
2. o sistema direciona o usuário para um formulário com as informações necessárias para a autenticação;
3. o usuário preenche as informações de autenticação;
4. o usuário clica no botão para confirmar a autenticação (“LOGIN”); e
5. o sistema direciona o usuário para a página inicial

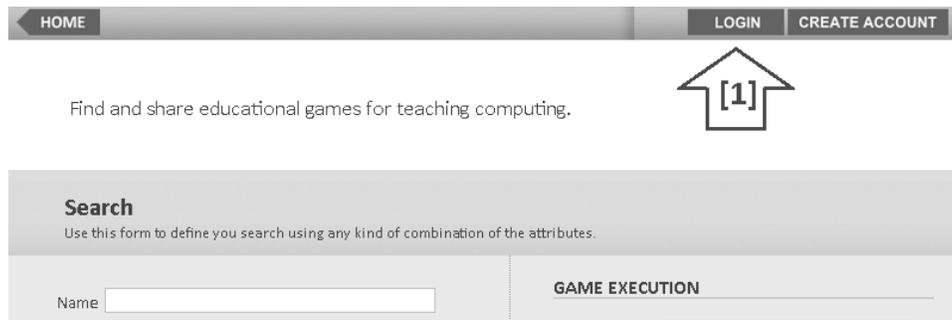


Figura 30 - Tela inicial no caso de uso 5 – autenticar como usuário

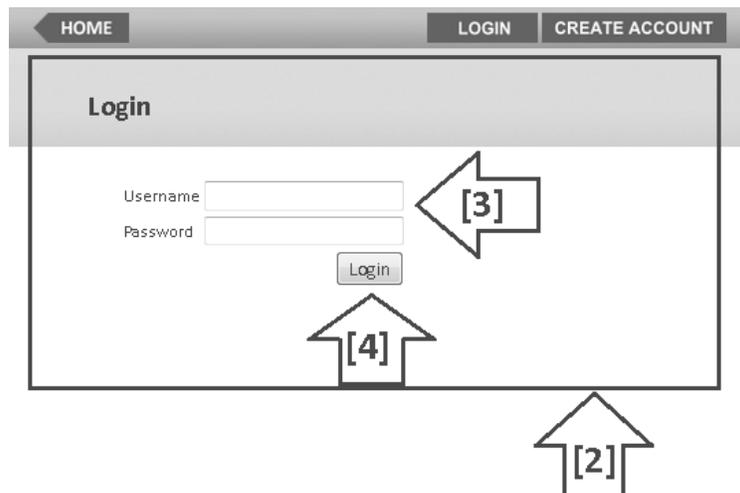


Figura 31 - Tela de autenticação no caso de uso 5 – autenticar como usuário

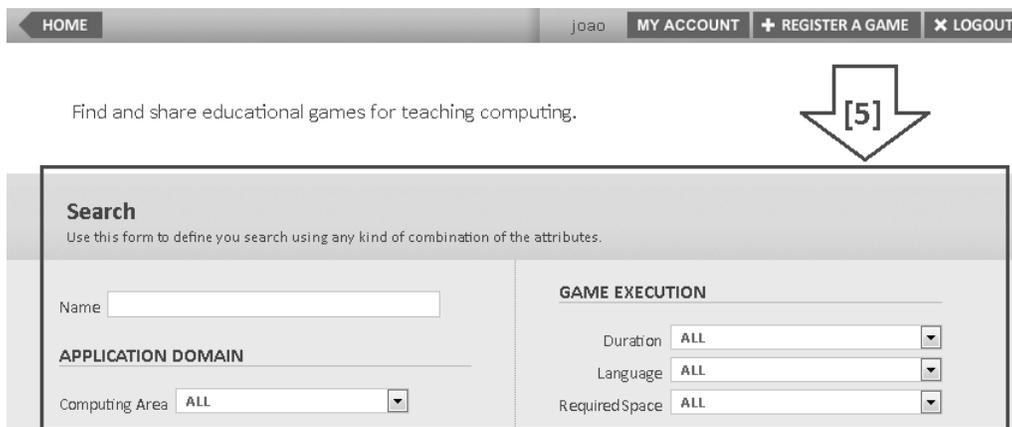


Figura 32 - Tela inicial no caso de uso 5 – autenticar como usuário

## Caso de uso 6 - remover autenticação de usuário

### Cenário principal:

1. o usuário clica no botão para remover a autenticação no repositório (“LOGOUT”);  
e
2. o sistema direciona o usuário para a página inicial.

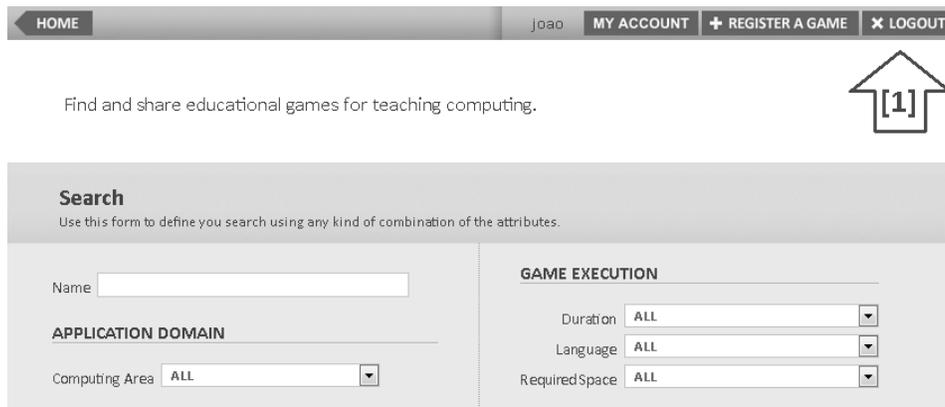


Figura 33 - Tela inicial com usuário autenticado no caso de uso 6 – remover autenticação de usuário

HOME LOGIN CREATE ACCOUNT

Find and share educational games for teaching computing.

[2]

**Search**  
Use this form to define you search using any kind of combination of the attributes.

Name

**APPLICATION DOMAIN**

Computing Area ALL

**GAME CLASSIFICATION**

Format ALL  
Type ALL  
Availability ALL

**GAME EXECUTION**

Duration ALL  
Language ALL  
Required Space ALL

**LEARNING OBJECTIVE**

Cognitive Level ALL  
Affective Level ALL  
Psychomotor Level ALL

Figura 34 - Tela inicial sem usuário autenticado no caso de uso 6 – remover autenticação de usuário

### Caso de uso 7 - avaliar e comentar jogo

#### Cenário principal:

1. o usuário clica no botão para avaliar e pontuar o jogo (*“Want to rate or comment this game? Click Here.”*);
2. o sistema exibirá um painel com campos para pontuar e comentar o jogo;
3. o usuário preenche os campos de pontuação e comentário;
4. o usuário salva a pontuação e comentário clicando no botão de confirmação (*“Confirm”*); e
5. o sistema exibe a página de informações do jogo com a pontuação e comentários realizados.

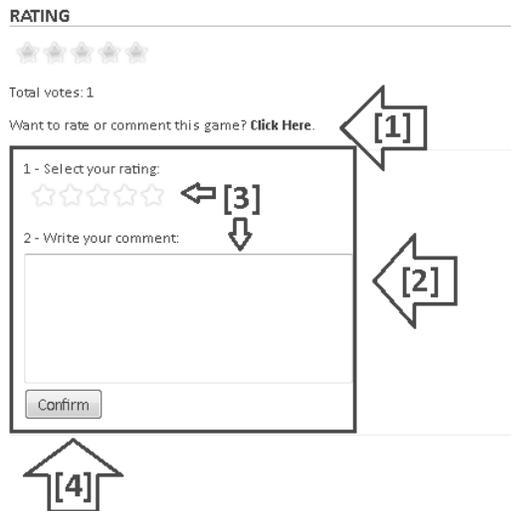


Figura 35 - Tela de avaliação e comentários no caso de uso 7 – avaliar e comentar jogo

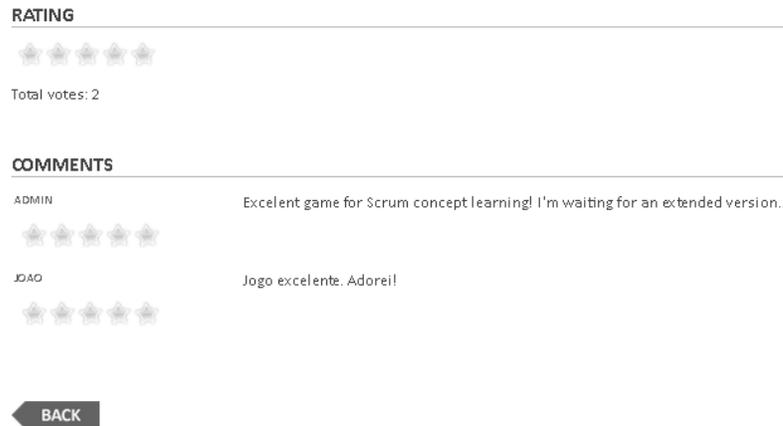


Figura 36 – Tela com as avaliação e comentários do jogo no caso de uso 7 – avaliar e comentar jogo

### Caso de uso 8 - cadastrar jogo

#### Cenário principal:

1. o usuário clica no botão para cadastrar o jogo (“REGISTER A GAME”);
2. o usuário preenche as informações relacionadas ao jogo;
3. o usuário clica no botão para confirmar o cadastro (“CONFIRM”); e
4. o sistema direciona o usuário para a página de exibição do jogo.

HOME joao MY ACCOUNT + REGISTER A GAME X LOGOUT

Find and share educational games for teaching computing.

**Search**  
Use this form to define you search using any kind of combination of the attributes.

Name

**APPLICATION DOMAIN**

**GAME EXECUTION**  
Duration ALL

Figura 37 - Tela inicial no caso de uso 8 – cadastrar jogo

HOME joao MY ACCOUNT + REGISTER A GAME X LOGOUT

**Game Registration**

**PHOTO**

UPLOAD A NEW PHOTO  Nenhum arquivo selecionado.

*Send a picture before you complete the game informations. At this moment, the upload of only one photo is possible.*

**INFORMATION**

NAME  Indicate the name of the instructional method

CREATOR  Indicate the name(s) of the creator(s)

PUBLICATION REFERENCE  Provide reference(s) to publications on the instructional method

KEYWORDS  A keyword or phrase describing the topic of this learning object

WEBSITE  Indicate a web site on the instructional method for further informations

**APPLICATION DOMAIN CLASSIFICATION**

COMPUTING AREA  Programming Fundamentals  
Integrative Programming  
Algorithms and Complexity  
Operating Systems Principles & Design

**CLASSIFICATION**

FORMAT  Select

TYPE  Select

INGROUPS OF  Select



Figura 38 - Tela de cadastro no caso de uso 8 – cadastrar jogo



Figura 39 – Tela com as informações do jogo no caso de uso 8 – cadastrar jogo

### Caso de uso 9 - editar jogo cadastrado pelo próprio usuário

#### Cenário principal:

1. o usuário seleciona o jogo que deseja editar apertando no respectivo botão (*“EDIT”*);
2. o usuário altera as informações que deseja;
3. o usuário confirma a edição, clicando no respectivo botão (*“CONFIRM”*); e
4. o sistema direciona o usuário para a página de exibição do jogo.

## Profile information:

Name: admin  
Institution: admin  
E-mail: admin@admin.com  
Country: AL

## Your registered games:

Scrum From Hell

COGNITIVE LEVELS: Evaluating, Analyzing.

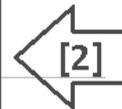
WEBSITE: <http://xp123.com/articles/scrum-from-hell/>

[MORE](#) [EDIT](#) [DELETE](#)



Figura 40 - Tela com a lista de jogos no caso de uso 9 – editar jogo cadastrado pelo próprio usuário

FEEDBACK TO THE LEARNER	<input type="text"/> Describe briefly if and what kind of feedback is given to the learner and/or if a debriefing activity is done (indicating the issues to be discussed).
DURATION	<input type="text" value="0 to 1/2 hour"/>
<b>EVALUATION</b>	
EVALUATION(S) PERFORMED	<input type="text"/>
EVALUATION FOCUS	<input type="text" value="Indicate the aspects being evaluated (e.g. learning effectiv"/>
TYPE OF EVALUATION	<input type="text"/>
LEVEL OF EVALUATION	<input type="text" value="Indicate the level considered in accordance to Kirkpatrick's"/>
NO. OF LEARNERS INVOLVED IN THE EVALUATION	<input type="text"/>
PRINCIPAL RESULTS OF THE EVALUATION	<input type="text" value="Cite the principal results obtained."/>



CONFIRM

Figura 41 - Tela para edição de informações no caso de uso 9 – editar jogo cadastrado pelo próprio usuário



Figura 42 - Tela com as informações editadas no caso de uso 9 – editar jogo cadastrado pelo próprio usuário

### Caso de uso 10 - visualizar jogos cadastrados pelo próprio usuário

#### Cenário principal:

1. o usuário clica no botão para visualizar os jogos por ele cadastrados (“MY ACCOUNT”); e
2. o sistema apresenta a lista de jogos cadastrados pelo usuário.



Figura 43 - Tela para acesso a lista de jogos no caso de uso 10 – visualizar jogos cadastrados pelo próprio usuário

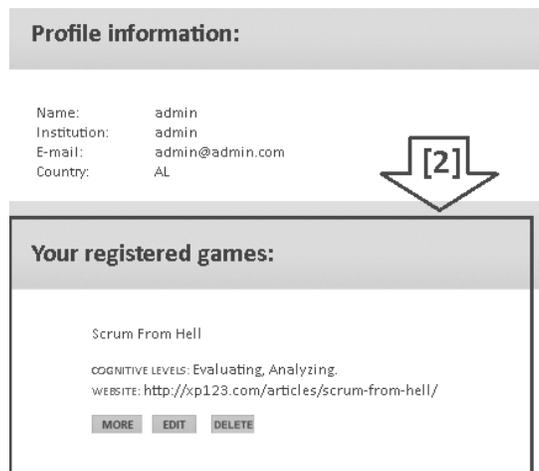


Figura 44 - Tela para visualização da lista de jogos no caso de uso 10 – visualizar jogos cadastrados pelo próprio usuário

### Caso de uso 11 - remover jogo cadastrado pelo próprio usuário

#### Cenário principal:

1. o usuário seleciona o jogo que deseja remover apertando no respectivo botão (“DELETE”);
2. o usuário confirma o desejo de remover o jogo, clicando no botão “OK”; e
3. o sistema apresenta a lista de jogos cadastrados pelo usuário.

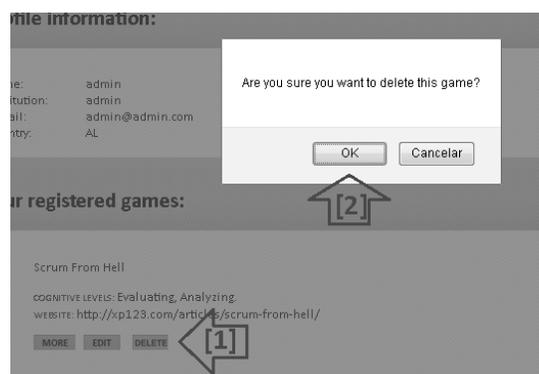


Figura 45 - Tela com a lista de jogos no caso de uso 11 – remover jogo cadastrado pelo próprio usuário.

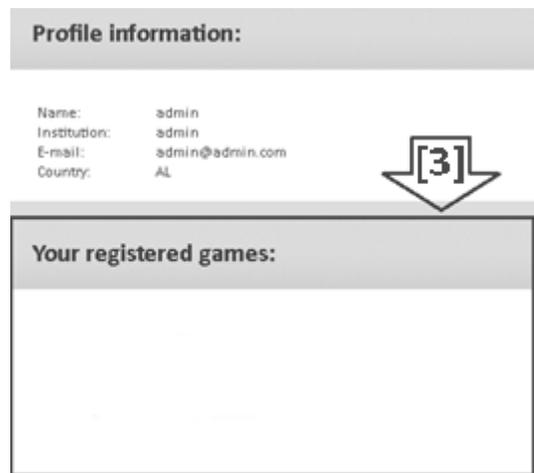


Figura 46 - Tela com o jogo removido da lista no caso de uso 11 – remover jogo cadastrado pelo próprio usuário.

## Caso de uso 12 - acessar o painel de administração

### Cenário principal:

1. o administrador clica no botão para acessar o painel de administração do repositório (“ADMINISTRATION”);
2. o sistema direciona o usuário para o painel de administração.

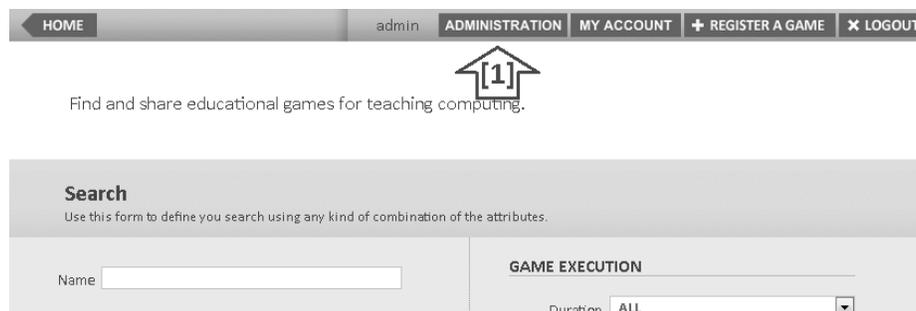


Figura 47 - Tela com para acesso o painel administrativo no caso de uso 12 – acessar o painel de administração.

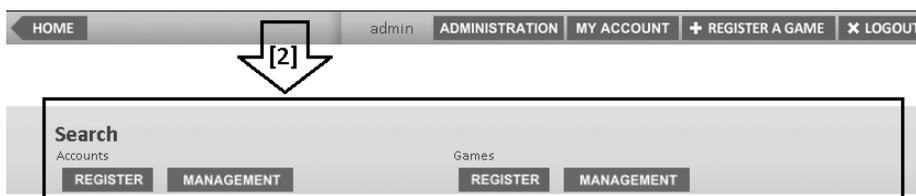


Figura 48 - Tela com o painel administrativo no caso de uso 12 – acessar o painel de administração.

## Caso de uso 13 - visualizar todos jogos cadastrados

### Cenário principal:

1. o administrador clica no botão para gerenciar os jogos cadastrados, no painel de administração de jogos (“MANAGEMENT”); e
2. o sistema exibe uma lista com os jogos cadastrados abaixo do painel de administração.

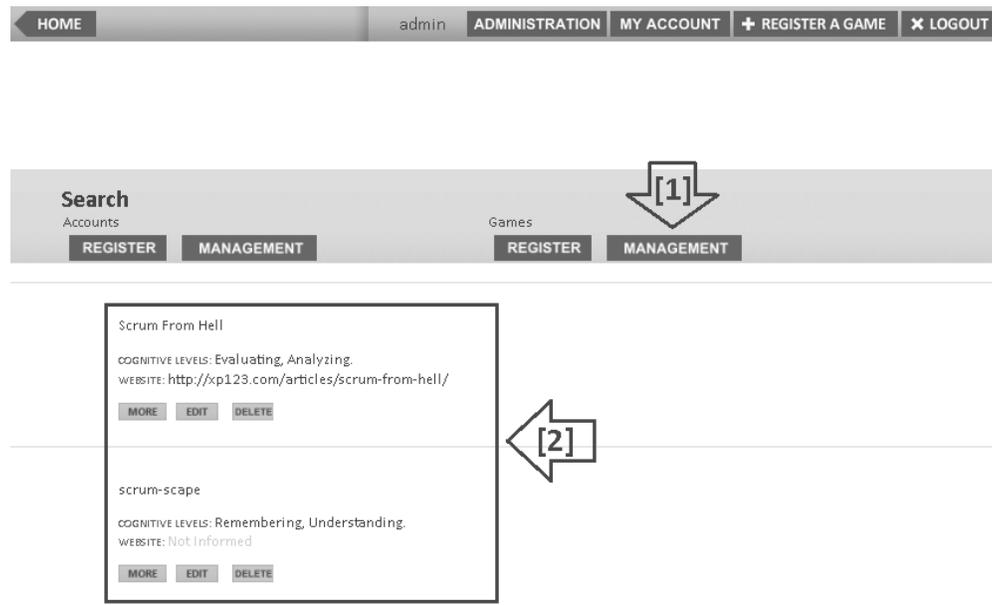


Figura 49 - Tela do caso de uso 13 – visualizar todos os jogos cadastrados.

### Caso de uso 14 - remover qualquer jogo

#### Cenário principal:

1. o administrador escolhe um jogo e clica no botão para removê-lo (“DELETE”);
2. o administrador confirma a ação clicando no botão “OK”; e
3. o sistema direciona o administrador para a página de administração.

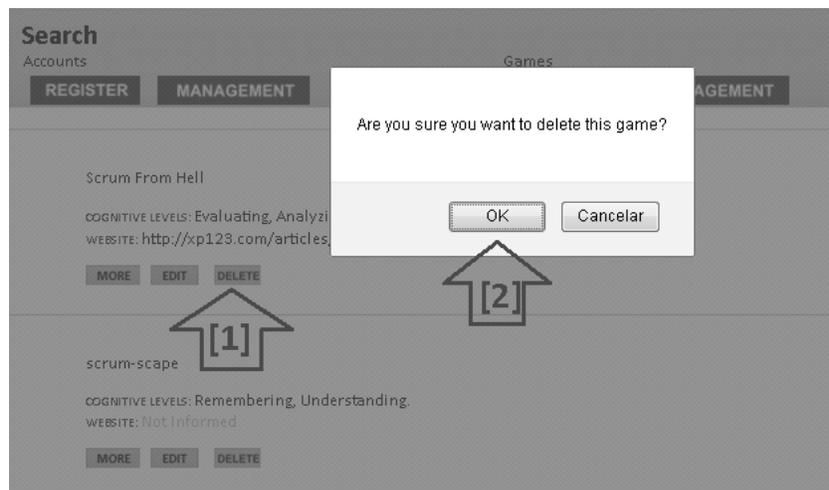


Figura 50 - Tela para remoção de jogo no caso de uso 14 – remover qualquer jogo.



Figura 51 – Tela pós-remoção de jogo no caso de uso 14 – remover qualquer jogo.

### Caso de uso 15 - editar qualquer jogo

#### Cenário principal:

1. o administrador escolhe um jogo e clica no botão para editá-lo (“EDIT”);
2. o administrador realiza as alterações desejadas;
3. o administrador confirma a edição, clicando no respectivo botão (“CONFIRM”); e
4. o sistema direciona o administrador para a página de exibição do jogo.

**Extensões:** nenhuma.

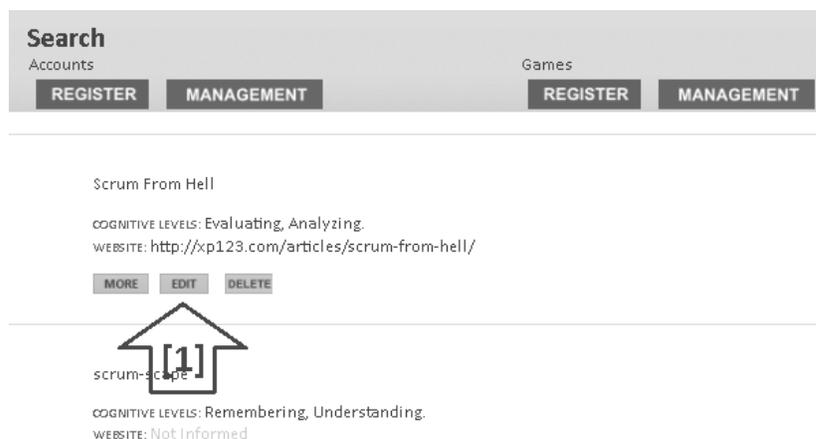


Figura 52 - Tela seleção do jogo para edição no caso de uso 15 – editar qualquer jogo.

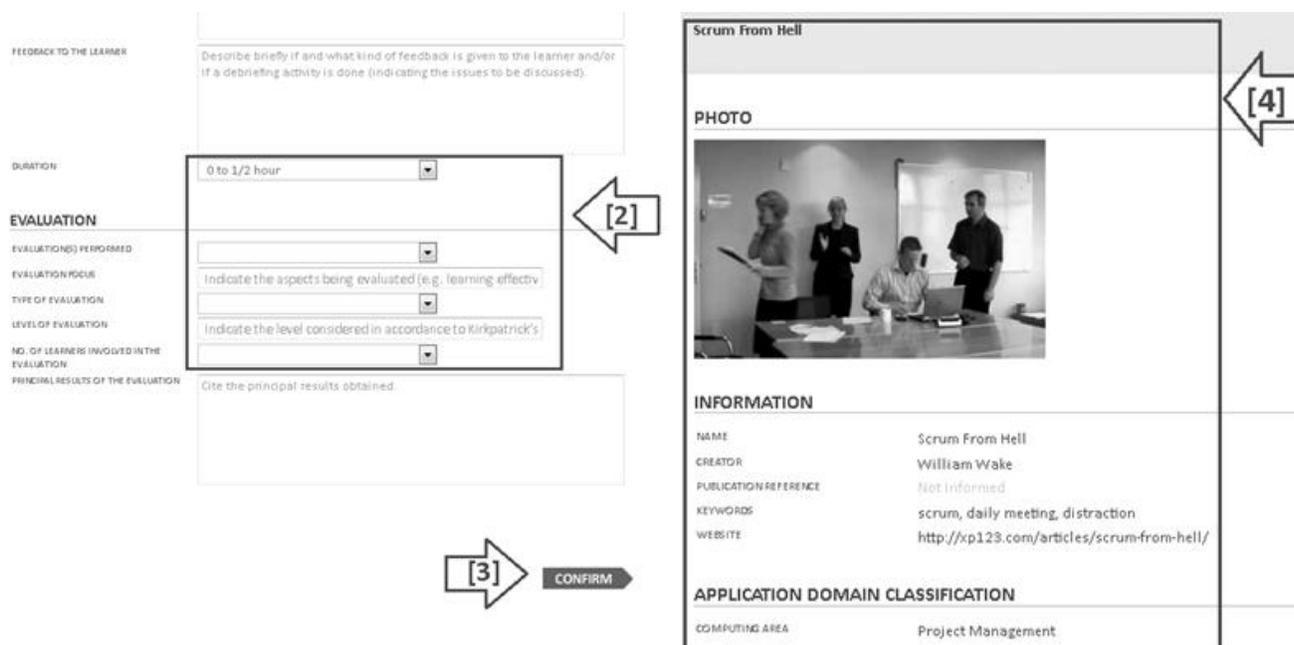


Figura 53 - Telas de edição e visualização no caso de uso 15 – editar qualquer jogo.

## Caso de uso 16 - visualizar usuários cadastrados

### Cenário principal:

1. o administrador clica no botão para gerenciar os usuários cadastrados, no painel de administração de usuários (“MANAGEMENT”); e
2. o sistema exibe uma lista com os usuários cadastrados abaixo do painel de administração.

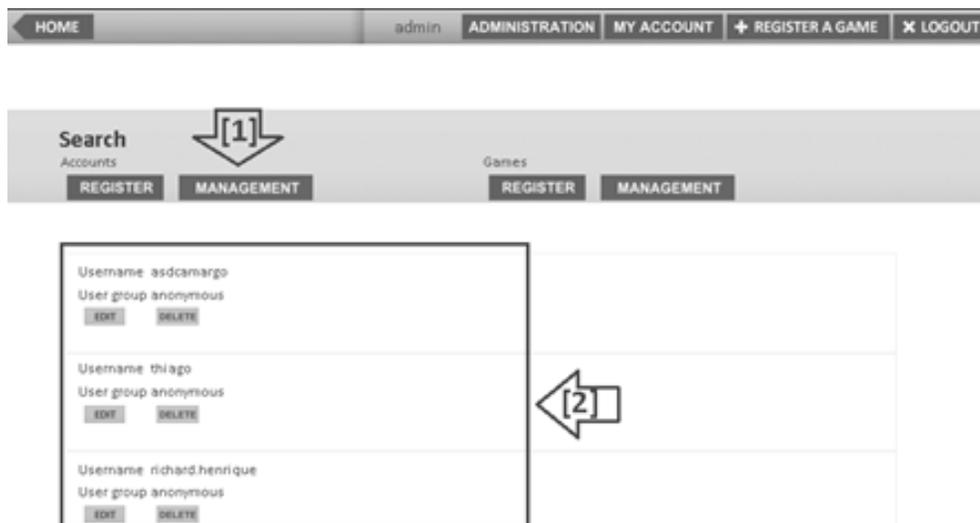


Figura 54 - Telas de edição e visualização no caso de uso 16 – visualizar usuários cadastrados.

### Caso de uso 17 - remover usuário

#### Cenário principal:

1. o administrador escolhe um usuário e clica no botão para removê-lo (“DELETE”); e
2. o sistema atualiza a lista de usuários cadastrados.

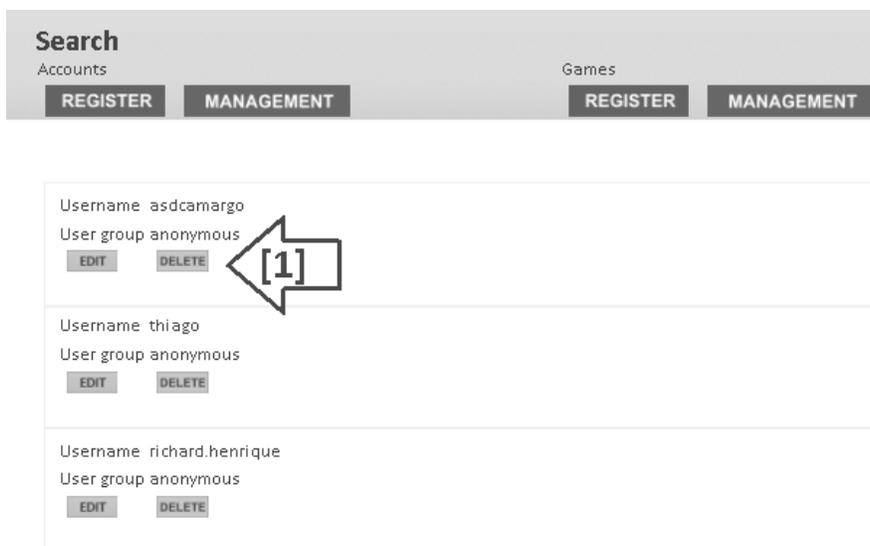


Figura 55 - Telas de seleção de usuário para remoção no caso de uso 17 – remover usuário.

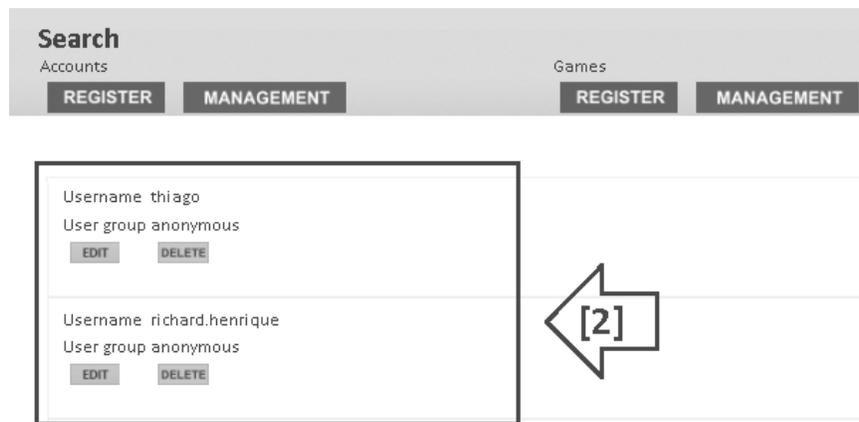


Figura 56 - Tela de exibição dos usuários restantes no caso de uso 17 – remover usuário.

### Caso de uso 18 - editar usuário

#### Cenário principal:

1. o administrador escolhe um usuário e clica no botão para editá-lo (“EDIT”);
2. o sistema direciona o administrador para a página de edição do usuário;
3. o administrador realiza as alterações desejadas;
4. o administrador clica no botão para confirmar as edições (“EDIT”); e
5. o sistema direciona o administrador para a lista de usuários cadastrados, no painel de administração.

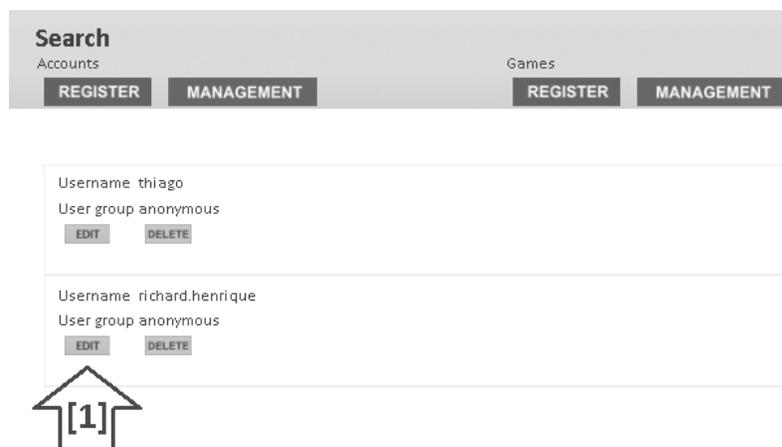


Figura 57 – Tela para escolha de usuário para edição no caso de uso 18 – editar usuário.

**User edit: richard.henrique** [2]

Name	Richard Henrique de Sot
Username	richard.henrique
Institution	UFSC
E-mail	richard.henrique@ufsc.
Country	BR
Password	••••••
Retype Password	••••••
User group	1

[3]

Edit [4]

Figura 58 - Tela para edição do cadastro do usuário no caso de uso 18 – editar usuário.

**Search** [5]

Accounts REGISTER MANAGEMENT Games REGISTER MANAGEMENT

Username	thiago	User group	anonymous	EDIT	DELETE
Username	richard.henrique	User group	anonymous	EDIT	DELETE

Figura 59 - Tela com todos os usuários cadastrados no caso de uso 18 – editar usuário.

Com a implementação dos casos de uso apresentados, é possível realizar os testes de aceitação no repositório. Estes testes são descritos na seção a seguir.

## 4.6. Testes do sistema

Para garantir que as funcionalidades implementadas estão em conformidade com os requisitos e casos de uso definidos, são realizados testes de aceitação por meio da execução dos passos descritos nos cenários principais de cada caso de uso, descritos na tabela 30 (os cenários dos casos de uso não são, necessariamente, executados na ordem em que estão listados na tabela).

Caso de uso	Dados do teste	Resultado do teste
1 - buscar jogo	Busca por meio dos filtros de “Product backlog” ( <i>Artifacts</i> ) e “Sprint planning meeting” ( <i>Ceremonies</i> ), selecionando a área da computação como “Project Management” e a metodologia como “Scrum”	OK
2 - visualizar informações do jogo	Seleção do jogo “Scrum From Hell”	OK
3 – cadastrar usuário	Cadastro do usuário “joao”, com nome “joao” e senha “1234”.	OK
4 - visualizar termos de serviço	Acesso a página de termos de serviço por meio do botão na parte inferior da página inicial.	OK
5 - autenticar como usuário	Acesso a página de autenticação e inserção dos dados de usuário (“joao”) e senha (“1234”).	OK
6 - remover autenticação de usuário	Autenticação como usuário “joao” e remoção da autenticação por meio de clique no botão “Logout”.	OK
7 - avaliar e comentar jogo	Adição do comentário “Excelente jogo para exemplificar o que não deve ser feito em Daily Meetings!” e da pontuação 5 no jogo “Scrum From Hell”.	OK
8 - cadastrar jogo	Cadastro das informações do jogo “Scrum From Hell”, como complexidade “Low”, artefatos “Task Board” e ceremonies “Daily scrum meeting”.	OK
9 - editar jogo cadastrado pelo próprio usuário	Adição dos papéis abordados no jogo “Scrum From Hell”: “Scrum Master” e “Team Member”.	OK
10 - visualizar jogos cadastrados pelo próprio usuário	Acesso por meio do botão “My Account” na página inicial.	OK
11 - remover jogo cadastrado pelo próprio usuário	Remoção do jogo “Scrum From Hell”.	OK

12 - acessar o painel de administração	Autenticação com o usuário “admin”, senha “admin” e clique no botão “Administration” na página inicial.	OK
13 - visualizar todos jogos cadastrados	Acesso por meio do botão “MANAGEMENT” no painel de administração.de jogos.	OK
14 - remover qualquer jogo	Clique no botão “DELETE” do jogo “Scrum-scape”, após ter listado todos os jogos cadastrados.	OK
15 - editar qualquer jogo	Edição do jogo “SMMYD” por meio do painel de administração de jogos.	OK
16 - visualizar usuários cadastrados	Acesso por meio do botão “MANAGEMENT” no painel de administração.de usuários.	OK
17 - remover usuário	Remoção do usuário “joao” após listagem de todos os usuários cadastrados por meio do painel de administração de usuários.	OK
18 - editar usuário	Edição das informações do usuário “joão” após a listagem de todos os usuários cadastrados por meio do painel de administração de usuários.	OK

Tabela 30 – Resultados dos testes de aceitação dos casos de uso.

A garantia de que o código está funcionando conforme foi concebido é realizada por meio da execução de testes unitários para as classes utilitárias do sistema, descritos na tabela 31.

Classe de teste	Descrição	Resultado
MethodIDsHandlerTest.java	Testa o algoritmo de união dos jogos filtrados por cada	OK

	parâmetro na busca.	
QueryBuilderTest.java	Testa o algoritmo utilizado para construção de queries SQL em geral.	OK
UpdateQueryBuilderTest.java	Testa o algoritmo utilizado para a construção de queries SQL de atualização das tabelas.	OK
InsertQueryBuilderTest.java	Testa o algoritmo utilizado para a construção de queries SQL de inserção nas tabelas.	OK

Tabela 31 – Resultados dos testes unitários nas classes utilitárias do SGR.

Com os resultados de sucesso descritos nesta seção, o sistema encontra-se em um estado adequado para ser disponibilizado em um ambiente de produção ou de testes que permita a avaliação do mesmo por usuários. Esta avaliação, com a descrição dos objetivos, dos dados e dos resultados é descrita no capítulo seguinte.

## 5. Avaliação

Neste capítulo são apresentadas a aplicação do repositório desenvolvido e a avaliação sobre requisitos de usabilidade, para obter respostas e percepções de usuários, possibilitando uma análise do repositório e do modelo de metadados propostos no presente trabalho.

### 5.1. Definição

Com o objetivo de avaliar a usabilidade das funcionalidades do Scrum Games Repository, é realizada uma avaliação de verificação de aspectos da usabilidade por um painel de especialistas.

Para que o sistema possa ser considerado usável, ele deve cumprir com os requisitos levantados em um determinado nível, de acordo com os seguintes aspectos (RUBIN; CHISNELL, 2008, p. 4):

- **Utilidade:** grau de cumprimento dos objetivos do usuário. O sistema pode ser fácil de usar e aprender e pode oferecer uma navegação satisfatória, mas se não atingir os objetivos específicos dos diferentes usuários, ele não será utilizado.
- **Eficiência:** velocidade em que o usuário consegue realizar as operações de forma precisa e completa.
- **Eficácia:** conformidade com os resultados esperados pelo usuário ao executar uma operação, considerando inconsistências e falhas.
- **Apreensibilidade (*learnability*):** capacidade de realizar operações no sistema após uma breve utilização ou treinamento.
- **Satisfação:** conjunto de sensações, percepções e opiniões do usuário sobre um sistema, que normalmente leva em consideração os outros atributos de usabilidade.
- 

A avaliação aplicada neste trabalho tem como objetivo verificar se o produto está em conformidade com os padrões e requisitos estabelecidos na fase prévia ao desenvolvimento.

Para guiar a definição, execução e análise desta avaliação, é utilizado o plano descrito a seguir (adaptado de RUBIN; CHISNELL, 2008, p. 35;67):

- identificação dos objetivos, questões e métricas;
- criação do plano de execução e definição dos participantes;
- execução da pesquisa e coleta de dados;
- avaliação dos dados e apresentação dos resultados.

Estes quatro passos do plano de avaliação são executados nas seções a seguir.

## 5.2. Identificação dos objetivos, questões e métricas

A elaboração de métricas para a avaliação é realizada através da especificação de objetivos e formulação de questões, utilizando o modelo GQM (*Goal, Question and Metric*). A definição de objetivos é considerada crítica para o sucesso da aplicação deste modelo, segundo Basili e Caldiera (1994, p. 529) e orienta-se a caracterização das dimensões de propósito, aspecto, objeto (ou processo) e ponto de vista<sup>18</sup>. A partir dos objetivos, são elaboradas questões para caracterizar o objeto no aspecto descrito. Para responder estas questões de forma quantitativa, são escolhidas métricas que permitam a extração de um conjunto de dados através da análise dos resultados do teste.

Os aspectos de usabilidade considerados, os objetivos formulados considerando cada aspecto, as questões propostas para avaliar o SGR dentro de cada objetivo, as hipóteses levantadas sobre as respostas de cada questão e as métricas extraídas no processo de avaliação são apresentados na tabela 32.

Aspectos de usabilidade	Objetivos	Questões	Hipóteses	Métricas
-------------------------	-----------	----------	-----------	----------

---

<sup>18</sup> Um exemplo de caracterização pode ser encontrado na definição do objetivo O1, onde o verbo transitivo direto “Avaliar” define o porquê da avaliação do objeto (propósito); a parte do predicativo “a utilidade”, define o qual o atributo é analisado (aspecto), o objeto direto da oração, “modelo de metadados”, define o que é analisado (objeto) e a outra parte do predicativo, “do ponto de vista do usuário”, define o público alvo da avaliação.<sup>1</sup>

<b>Utilidade</b> (RUBIN; CHISNELL, 2008) / Funcionalidade (ISO/IEC 25010)	O1. Avaliar a utilidade do SGR do ponto de vista do usuário.	Q1. A busca por jogos atende as necessidades do usuário?	Ao menos 75% dos entrevistados consideram que a busca por jogos no SGR atende suas necessidades.	M1. 75% das respostas são de concordância total em relação a utilidade da busca (1º quartil); M2. Metade das respostas são de concordância total em relação a utilidade da busca (mediana);
		Q2. As informações exibidas na página do jogo atendem as necessidades do usuário?	Ao menos 75% dos entrevistados consideram que as informações exibidas na página do jogo atendem suas necessidades.	M3. 75% das respostas são de concordância total em relação a utilidade das informações do jogo (1º quartil); M4. Metade das respostas são de concordância total em relação a utilidade das informações do jogo (mediana);
<b>Eficiência</b> (RUBIN; CHISNELL, 2008) / Performance e Usabilidade (ISO/IEC 25010)	O2. Avaliar o tempo de resposta do processamento das operações no ponto de vista do usuário.	Q3. O resultado da busca foi apresentado em menos de cinco segundos?	Ao menos 75% dos entrevistados consideram que o resultado da busca foi apresentado em menos de cinco segundos.	M5. 75% das respostas são de concordância total em relação ao tempo de resposta do processamento da busca (1º quartil); M6. Metade das respostas são de concordância total em relação ao tempo de resposta do processamento da busca (mediana);
		Q4. A exibição do jogo após seu cadastro foi apresentada em menos de cinco segundos?	Ao menos 75% dos entrevistados consideram que o resultado do cadastro foi apresentado em menos de cinco segundos.	M7. 75% das respostas são de concordância total em relação ao tempo de resposta do processamento do cadastro de um jogo (1º quartil); M8. Metade das respostas são de concordância total em relação ao tempo de resposta do processamento do cadastro de um jogo (mediana);
	O3. Avaliar a dificuldade na execução das operações disponibilizadas ao usuário.	Q5. É fácil realizar a busca por jogos?	75% dos entrevistados afirmam ter realizado a busca com facilidade.	M9. 75% das respostas são de concordância total em relação a facilidade na realização da operação de busca de jogos (1º quartil); M10. Metade das respostas são de concordância total em relação a facilidade na realização da operação de busca de jogos (mediana);
		Q6. É fácil realizar o cadastro de jogos?	50% dos entrevistados afirmam ter cadastrado um jogo com facilidade.	M11. Metade das respostas são de concordância total em relação a facilidade no cadastro de um jogo (mediana); M12. 25% das respostas são de concordância total em relação a facilidade no cadastro de um jogo (3º quartil);
<b>Eficácia</b> (RUBIN; CHISNELL, 2008) / Confiabilidade e Usabilidade (ISO/IEC 25010)	O4. Avaliar a completude das operações disponibilizadas ao usuário	Q7. É possível cadastrar um usuário?	Ao menos 75% dos entrevistados conseguem cadastrar um usuário com suas informações pessoais.	M13. 75% das respostas são de concordância total em relação a completude da operação de cadastro de um usuário (1º quartil); M14. Metade das respostas são de concordância total em relação a completude da operação de cadastro de um usuário (mediana);

		Q8. É possível realizar a busca por jogos?	Ao menos 75% dos entrevistados conseguem realizar a busca por jogos.	M15. 75% das respostas são de concordância total em relação a completude da operação de busca de um jogo (1º quartil); M16. Metade das respostas são de concordância total em relação a completude da operação de busca de um jogo (mediana);
		Q9. É possível cadastrar um jogo?	Ao menos 75% dos entrevistados conseguem realizar o cadastro de jogos.	M17. 75% das respostas são de concordância total em relação a completude da operação de cadastro de um jogo (1º quartil); M18. Metade das respostas são de concordância total em relação a completude da operação de cadastro de um jogo (mediana);
		Q10. É possível alterar as informações de um jogo cadastrado?	Ao menos 75% dos entrevistados conseguem alterar as informações de um jogo por eles cadastrado.	M19. 75% das respostas são de concordância total em relação a completude da operação de edição de informações de um jogo cadastrado pelo próprio usuário (1º quartil); M20. Metade das respostas são de concordância total em relação a completude da operação de edição de informações de um jogo cadastrado pelo próprio usuário (mediana);
		Q11. É possível excluir um jogo cadastrado?	Ao menos 75% dos entrevistados conseguem excluir um jogo por eles cadastrado.	M21. 75% das respostas são de concordância total em relação a completude da operação de exclusão de um jogo cadastrado pelo próprio usuário (1º quartil); M22. Metade das respostas são de concordância total em relação a completude da operação de exclusão de um jogo cadastrado pelo próprio usuário (mediana);
	O5. Avaliar a confiabilidade das operações disponibilizadas ao usuário.	Q12. São encontrados erros nas informações exibidas após o cadastro de um jogo?	Nas mais do que 25% dos entrevistados afirmam ter encontrado erros nas informações disponibilizadas após o cadastro de um jogo.	M23. 25% das respostas são de concordância total em relação a confiabilidade das informações exibidas após o cadastro de um jogo pelo próprio usuário (3º quartil); M24. Metade das respostas são de concordância total em relação a confiabilidade das informações exibidas após o cadastro de um jogo pelo próprio usuário (mediana);
<b>Apreensibilidade</b> (RUBIN; CHISNELL, 2008)	O6. Avaliar o aprendizado da utilização no ponto de vista do usuário.	Q13. A aprendizagem na utilização do sistema é rápida?	75% dos entrevistados afirmam ter aprendido a utilizar o SGR rapidamente.	M25. 75% das respostas são de concordância total em relação a rapidez do aprendizado na utilização do SGR (1º quartil); M26. Metade das respostas são de concordância total em relação a rapidez do aprendizado na utilização do SGR (mediana);

<b>Satisfação</b> (RUBIN; CHISNELL, 2008) / Usabilidade (ISO/IEC 25010)	O7. Avaliar a satisfação na execução das operações no ponto de vista do usuário.	Q14. O design gráfico do SGR é atraente?	75% dos entrevistados consideram que o design gráfico do SGR é atraente.	M27. 75% das respostas são de concordância total em relação a satisfação com o design gráfico das interfaces de usuário do SGR (1º quartil); M28. Metade das respostas são de concordância total em relação a satisfação com o design gráfico das interfaces de usuário do SGR (mediana);
		Q15. O uso das operações disponibilizadas pelo SGR é satisfatório?	75% dos entrevistados se sentem satisfeitos com a busca de jogos no SGR.	M29. 75% das respostas são de concordância total em relação a satisfação com a busca de jogos no SGR (1º quartil); M30. Metade das respostas são de concordância total em relação a satisfação com a busca de jogos no SGR (mediana);
			75% dos entrevistados afirmam que usariam o SGR na busca de jogos para o ensino do Scrum.	M31. 75% das respostas são de concordância total em relação a ao uso posterior do SGR na busca de jogos para o ensino do Scrum (1º quartil); M32. Metade das respostas são de concordância total em relação a ao uso posterior do SGR na busca de jogos para o ensino do Scrum (mediana);
			75% dos entrevistados afirmam que recomendaria o SGR para seus colegas.	M33. 75% das respostas são de concordância total em relação à recomendação do SGR colegas (1º quartil); M34. Metade das respostas são de concordância total em relação à recomendação do SGR colegas (mediana);

Tabela 32 – Aspectos de usabilidade, objetivos, questões, hipóteses e métricas utilizados na avaliação do SGR.

Com o levantamento das métricas através do método GQM, é possível estabelecer o público alvo e criar um questionário próprio para este público, com o propósito de obter os dados de avaliação. A criação do plano de execução e a seleção dos participantes, que precede a execução e análise dos dados, são descritos a seguir.

### 5.3. Criação da plano de execução e seleção dos participantes

Para possibilitar a obtenção dos valores é apresentado, no Apêndice A, um questionário com perguntas elaboradas a partir das métricas (M1 à M47), que utilizam valores qualitativos ordinais, com um extremo expresso no valor 1, que representa discordância total, e o outro extremo expresso no valor 5, representando concordância total.

A população alvo desta avaliação é composta por especialistas da área de gerenciamento de projetos. Segundo BEECHAM et al (2004), a utilização de um painel de especialistas, que possuem habilidade para avaliar métodos e técnicas utilizados em seu domínio de experiência, apresenta resultados mais assertivos quando comparados com outros métodos de avaliação, principalmente na avaliação e colaboração no desenvolvimento de modelos. Mais detalhes sobre os requisitos considerados na escolha dos participantes pode ser encontrado no plano de execução (Apêndice B).

## 5.4. Execução

A avaliação do SGR foi realizada conforme o plano descrito no Apêndice B, por um painel composto por treze especialistas da área de gerenciamento ágil de projetos e Scrum.

O período em que o SGR e o questionário foram disponibilizados para avaliação foi do dia 12/11/2013 até o dia 17/11/2013. Foram reportados pequenos problemas de estabilidade do servidor utilizado, mas que não comprometeram com o processo em geral.

A análise dos dados coletados é descrita no subcapítulo a seguir.

## 5.5. Análise dos dados

Na análise dos dados coletados através da aplicação do questionário com o painel de especialistas é considerada a mediana e o cálculo do 1º ou 3º quartil, conforme a hipótese levantada na definição dos objetivos, questões e métricas.

Cada questão é respondida com pontuação em uma escala de 1 a 5, onde o menor valor indica discordância total e o maior valor indica concordância total do participante, em relação a pergunta do questionário.

A análise, descrita a seguir, é ordenada pelos aspectos de usabilidade considerados no plano de avaliação.

### Utilidade (Funcionalidade - ISO/IEC 25010)

Questão	Mediana	1º Quartil
Q1. A busca por jogos atende as necessidades do usuário?	4	4
Q2.As informações exibidas na página do jogo atendem as necessidades do usuário?	5	4

As respostas sobre o aspecto de utilidade, considerando a busca por jogos e a exibição das informações, apesar de não confirmarem a hipótese inicial, de concordância total de ao menos 75% dos participantes, atinge este percentil em concordância parcial.

### **Eficiência (Performance e Usabilidade - ISO/IEC 25010)**

<b>Questão</b>	<b>Mediana</b>	<b>1º Quartil</b>
Q3. O resultado da busca foi apresentado em menos de cinco segundos?	5	4
Q4. A exibição do jogo após seu cadastro foi apresentada em menos de cinco segundos?	5	4

A hipótese levantada sobre a usabilidade da busca e do cadastro de jogos sobre o aspecto da eficiência, considerando o tempo de resposta de processamento destas operações, foi rejeitada; mas se considerarmos que 75% dos participantes da avaliação indicaram ao menos concordância parcial com o desempenho do processamento, é possível considerar o tempo de resposta como aceitável.

<b>Questão</b>	<b>Mediana</b>	<b>1º Quartil</b>
Q5. É fácil realizar a busca por jogos?	3	3
Q6. É fácil realizar o cadastro de jogos?	4	4

Considerando a facilidade que os usuários tiveram para realizar as operações, é possível perceber que a facilidade esperada para a realização da busca de jogos, de 75% de concordância total, não foi encontrada. Ao contrário desta análise, a operação de cadastro de jogos, apesar de sua extensão, é relatada como sendo mais fácil de usar, observando a concordância parcial (ao menos) de 75% dos participantes.

### **Eficácia (Confiabilidade e Usabilidade - ISO/IEC 25010)**

<b>Questão</b>	<b>Mediana</b>	<b>1º Quartil</b>
Q7. É possível cadastrar um usuário?	5	5
Q8. É possível realizar a busca por jogos?	5	5
Q9. É possível cadastrar um jogo?	5	5
Q10. É possível alterar as informações de um jogo cadastrado?	5	5
Q11. É possível excluir um jogo cadastrado?	5	5

A hipóteses levantadas sobre a completude das funções de busca, cadastro, edição e remoção de jogos, e cadastro de usuários, foram confirmadas: ao menos 75% dos participantes indicaram conformidade total em relação as perguntas sobre a conclusão das operações.

### **Eficácia (Confiabilidade e Usabilidade - ISO/IEC 25010)**

<b>Questão</b>	<b>Mediana</b>	<b>1º Quartil</b>
Q12. São encontrados erros nas informações exibidas após o cadastro de um jogo?	5	2

A hipótese de que 75% dos participantes não encontrariam erros nas informações exibidas após o cadastro de um jogo foi descartada. Considerando a avaliação de 40% dos participantes, que confirmaram parcialmente a apresentação de erros após esta operação, percebe-se que existe a necessidade de rever o tratamento de erros e a consistência do cadastro e da exibição de jogos.

### **Aprensibilidade (Confiabilidade e Usabilidade - ISO/IEC 25010)**

<b>Questão</b>	<b>Mediana</b>	<b>1º Quartil</b>
Q13. A aprendizagem na utilização do sistema é rápida?	5	3

A hipótese de que 75% dos participantes aprenderiam rapidamente a utilizar o sistema foi descartada. O cenário encontrado, ao avaliar as respostas, é de que existe uma necessidade de aprimorar a navegação e a exibição de botões, assim como incluir uma página de ajuda, segundo a avaliação de 40% dos participantes, que não concordaram e nem discordaram em relação a rapidez na aprendizagem do SGR.

### **Satisfação (Confiabilidade e Usabilidade - ISO/IEC 25010)**

<b>Questão</b>	<b>Mediana</b>	<b>1º Quartil</b>
Q14. O design gráfico do SGR é atraente?	4	4

A avaliação esperada para o *desing* gráfico do SGR não foi confirmada, onde 75% dos participantes indicaram que concordam parcialmente com esta questão, ao contrário do que era previsto, de 75% de concordância total.

<b>Questão</b>	<b>Mediana</b>	<b>1º Quartil</b>
Q15. O uso das operações disponibilizadas pelo SGR é satisfatório?	4	4

A avaliação sobre a satisfação do usuário no uso do SGR, composta por três métricas, que questionam se o SGR é satisfatório, se ele seria utilizado novamente para buscar jogos, e se ele seria indicado para colegas, foi positiva, mas não confirmou a hipótese inicial levantada, de que ao menos 75% dos participantes indicariam concordância total sobre estas questões.

### **Pontos fortes**

Os participantes da avaliação consideraram que o *desing* gráfico é simples mas atende os propósitos especificados, que a busca estruturada por parâmetros de classificação facilita o trabalho do professor na procura de jogos para um objetivo específico. A facilidade no aprendizado e o tempo de resposta do processamento das operações também foram apontados como vantagens do repositório.

### **Sugestões de melhoria**

Foram fornecidas sugestões de melhoria pelos participantes, na parte do *desing* gráfico, destacando o botão de cadastro de jogos para que seja mais fácil visualizá-lo; na parte de apoio ao uso do repositório, adicionando ítems de ajuda para explicar os itens do cadastro, como taxionomia de objetivos educacionais, classificação de jogos por formato, etc..; nas funcionalidades, adicionando a opção de edição das informações do perfil, cadastradas pelo próprio usuário.

### **5.5.1. Ameaças a validade**

A avaliação foi realizada por um grupo fechado de participantes (no painel de especialistas), que foi convidado por ter proximidade com o autor do trabalho ou com o grupo de pesquisa GQS – INCOD – INE – UFSC. Esta relação pode afetar os resultados da avaliação, pelo fato de que o participante pode evitar uma resposta negativa para não desqualificar o trabalho.

A quantidade de participantes e a composição do painel, heterogênea, por participantes com experiência no ensino de gerenciamento ágil de projetos e Scrum, e por participantes com experiência prática na aplicação do Scrum, pode não representar os usuário reais do repositório, invalidando a análise realizada neste trabalho.

Ao utilizar o SGR no ambiente de avaliação, disponibilizado pela UFSC, os participantes relataram momentos de lentidão e de falhas na conexão com o banco de dados. Entretanto, estas falhas não afetaram o processo de avaliação, porque todos os participantes que reportaram problemas, conseguiram finalizar o processo posteriormente.

Uma sugestão para minimizar estes riscos é a realização de uma segunda avaliação, com um grupo maior e mais representativo de participantes, em um ambiente estável e com prevenção de falhas.

## 6. Conclusão

Neste trabalho foi realizada a apresentação do contexto, abordando o cenário atual do gerenciamento de projetos, a utilização da metodologia Scrum e de jogos para o ensino dos conceitos desta.

Para a busca destes jogos foram analisados os repositórios disponíveis para acesso geral, que suportassem, principalmente, a busca, cadastro, edição, avaliação e remoção de jogos. Nesta análise foi verificado que o repositório IGR cumpria os requisitos básicos para utilização com jogos sobre gerenciamento de projetos, mas não disponibilizava ferramentas para utilização de jogos sobre o Scrum. Com este problema surgiu a necessidade de extensão do repositório IGR para suportar jogos para o ensino do Scrum.

A validação dos metadados levantados, como por exemplo, das características dos jogos e do Scrum, foi realizada através da análise do estado da arte em jogos sobre Scrum. Nesta análise foi constatado que as principais informações dos jogos podiam ser cadastradas utilizando o modelo de metadados especificado.

O sistema desenvolvido utilizou um modelo de metadados composto pelos metadados especificados neste trabalho e pelos disponibilizados no repositório IGR. As principais modificações realizadas neste sistema foram nas telas de busca, cadastro e edição de jogos.

Estas modificações e a utilidade do repositório com o suporte de busca, visualização e manipulação de jogos para o ensino do Scrum, foram validadas através de um painel de especialistas, por meio de um questionário com perguntas sobre a usabilidade do sistema.

Com este trabalho, espera-se contribuir para a utilização de jogos para o ensino do Scrum, fornecendo à instrutores, ferramentas de busca e de visualização de jogos com uma estrutura bem definida de características, classificações e avaliação da utilização.

Para trabalhos futuros, pretende-se melhorar a resposta ao usuário no cadastro de jogos, melhorar o código e a modelagem do repositório e incluir uma página de ajuda ao usuário.

## Referências

- ABES. **Mercado Brasileiro de Software**: panorama e tendências. São Paulo: ABES, 2010.
- ABES. **Mercado Brasileiro de Software**: panorama e tendêncais. São Paulo: ABES, 2011.
- ABRAHAMSSON, S. R. **Agile software development methods**. *VTT Publications 478*, 2002.
- ADAPTWORKS. **Treinamentos**. Disponível <<http://www.adaptworks.com.br/treinamentos>>. Acesso em 13 nov. 2012.
- GAMMA et al. **Design Patterns**: Elements of Reusable Object-Oriented Software. NJ: Addison-Wesley, 1995.
- BARGMEYER, B. E.; GILLMAN, D. W. **Metadata Standards and Metadata Registries**: An Overview. Bureau of Labor Statistics, 2002: Disponível em : <<http://www.bls.gov/ore/pdf/st000010.pdf>>. Acesso em 15 mai. 2012.
- BARTON, M. R.; WATERS, M. M. **Creating an Institutional Repository**: LEADIRS Workbook. Cambridge: MIT Libraries, 2005
- BASILI et. al. **Goal/Question/Metric Approach**. Encyclopedia of Software Engineering. John Wiley & Sons, 1994.
- BECK, K. **Extreme Programming Explained**: Embrace Change. 2ª ed. Addison Wesley Professional: 2004.
- BECK, K. et. al. (2001). **Manifesto para Desenvolvimento Ágil de Software**. Disponível em <<http://agilemanifesto.org/iso/ptbr/>>. Acesso em 25 mai. 2011
- Beecham, S. e. (2004). *Using an expert panel to validate a requirements process improvement model*. Hatfield, UK: University of Hertfordshire, Coolege Lane.
- BEECHAM, S., HALL, T.; RAINER, A.. **Validating the R-CMM**. *Department of Computer Science, Faculty of Engineering and Information Scienses*, 2003.
- BLOOM, B. S. **Taxionomia de objetivos educacionais**: domínio cognitivo. Porto Alegre: Editora Globo, 1979.
- BLOOM, B. S., KRATHWOHL, D. R., & BERTRAM, B. M. **Taxionomia de objetivos educacionais**: domínio afetivo. Porto Alegre: Editora Globo, 1977.
- BONETTI, T. M. **Desenvolvimento de um Repositório Colaborativo para Compartilhar Atividades de Ensino na Área de Gerenciamento de Projetos**. Florianópolis, SC, Brazil, 2011.
- BRAND, A. et. al. (). **Metadata Demystified**. The Sheridan Press & NISO Press, 2003.

BRUM, R. **Domínio psicomotor: objetivos e avaliação**. Porto Alegre: Livraria Sulina Editora, 1977.

BUGLIONE, L. **Simulation Games for Learning Organizations**, 2011. Disponível em: <<http://www.semq.eu/leng/proimplo.htm>>. Acesso em 28 jun. 2012.

BUSETTI, E. et. al. **Repositories of Learning Objects as Learning Environments for Teachers**. *Proceedings of the IEEE International Conference on Advanced Learning Technologies (ICALT'04)*, 2004.

CAELUM. *Cursos de Scrum e Agile*. Disponível em: < <http://www.caelum.com.br/cursos/agile/>>. Acesso em 25 set. 2012.

CAILLOIS, R. **The Definition of Play and The Classification of Games**. In: K. Z. Salen, *The Game design reader: a rules of play anthology* (pp. 122-155). Massachussets: The MIT Press, 2006.

CARSON, B. **Games and activites for teaching project management**. Disponível em <<http://www.nasaga.org/forum/topics/games-and-activites-for-teaching-project-management>>. Acesso em 13 mar. 2012.

CHARVAT, J. **Project Management Methodologies: Selecting, Implementing, and Supporting Methodologies and Processes for Projects**. NJ: John Wiley & Sons, 2003.

COCKBORN, A. *Writing Effective Use Cases*. NJ: Addison-Wesley, 2000.

COHN, M. **User Stories Applied: For Agile Software Development**. Boston: Addison-Wesley Professional, 2004.

COHN, M. **Succeeding with Agile: Software Development Using Scrum**. Ann Arbor: Addison-Wesley, 2010.

CREATIVE COMMONS. *About*. Disponível em: <<http://creativecommons.org/about>>. Acesso em 05 set. 2013.

DE SMET, J. **Agile games and techniques, time to share some**. 2008. Disponível em <<http://agilefun.com/?p=63>>. Acesso em 10 set. 2012.

DEEMER, P.; BENEFIELD, G. **Scrum Primer**. *Yahoo*, 2006.

DICK, W., CAREY, L., & CAREY, J. **The Systematic Design of Instruction**. New York: Addison-Wesley, 2001.

DYBA, T.; DINGSOYR, T. **What Do We Know about Agile Software Development: Voice of Evidence**. Fraunhofer Center for Experimental Software Engineering, 2009.

ELLIOT, K.; SWEENEY, K. **Quantifying the reuse of learning objects**. The University of Melbourne: 2007. Australasian Journal of Educational Technology 2008, 24(2), 137-142: Disponível em <<http://www.ascilite.org.au/ajet/ajet24/elliott.pdf>>. Acesso em 15 mai. 2011

FELICIA, P. **Handbook of Research on Improving Learning and Motivation through Educational Games: Multidisciplinary Approaches**. IGI Global, 2011.

FORLI INOCENTE, D. **Avaliação das necessidades de treinamento**: um estudo da gestão na unidade da Fiocruz na Bahia. Riberão Preto: Faculdade de Economia, Administração e Contabilidade de Riberão Preto/USP, 2006.

FOWLER, M. **Patterns of enterprise application architecture**. Boston: Pearson Education, 2003.

GADOTTI, M. **História Das Idéias Pedagógicas**. São Paulo: Editora Ática, 1997.

Gardner, H. **Frames of Mind: The Theory of Multiple Intelligences**. New York: Basic Books, 2011.

GLOBALCODE. **Academia do Agile**. Disponível em: < <http://www.globalcode.com.br/treinamentos/carreiras/academia-do-agile>>. Acesso em 13 nov. 2012.

GQS. **Instructional Games Repository**. Disponível em: < <http://srv.gqs.ufsc.br:8080/InstructionalGamesRepository/>>. Acesso em 08 fev. 2013.

GQS. **Software Engineering and Management Education**. Disponível em : <<http://www.gqs.ufsc.br/software-engineering-education/>>. Acesso em 10 set. 2012.

GREENING, J. **Renaissance of Software**. Disponível em: <<http://renaissancesoftware.net/files/articles/PlanningPoker-v1.1.pdf>>. Acesso em 07 abr. 2013

GRESSE VON WANGENHEIN, C. A. **Material da disciplina INE5617 - Gerencia de Projetos**. Florianópolis: Sistemas de Informação/INE, UFSC, 2010.

GRESSE VON WANGENHEIN, C. A. **Como ensinar com Jogos?**. 2012. Disponível em : <[http://www.inf.ufsc.br/~gresse/download/CSBC2012-ComoEnsinarComJogos\\_Wangenheim\\_vf.pdf](http://www.inf.ufsc.br/~gresse/download/CSBC2012-ComoEnsinarComJogos_Wangenheim_vf.pdf)>. Acesso em 06 dez. 2012.

GRESSE VON WANGENHEIN, C. A. **Activities for teaching project management**. Disponível em: <<http://www.inf.ufsc.br/~gresse/subpaginas/ProjectManagementTeaching.html>>. Acesso em 28 jun. 2012.

GRESSE VON WANGENHEIN, C. A. **Games and activites for teaching project management**. NASAGA. - North American Simulation and Gaming Association, 2012. Disponível em: <<http://www.nasaga.org/forum/topics/games-and-activites-for-teaching-project-management>>. Acesso em 10 fev. 2013.

HAIDT, R. C. **Curso de Didática Geral**. São Paulo: Editora Ática, 2001.

HANOULLE, Y. **Grupos do Google**. Disponível em: < <https://groups.google.com>

/forum/#!forum/agilegames>. Acesso em 10 out. 2013.

HARMAN, K.; KOOHANG, A. **Learning Objects: Standards, Metadata, Repositories, & LCMS**. Santa Rosa: Informing Science Press, 2007.

HEERY, R. **Digital Repositories Review**. Disponível em: <<http://www.ukoln.ac.uk/repositories/publications/review-200502/digital-repositories-review-2005.pdf>>. Acesso em 08 jul. 2012.

IEEE. **Draft Standard for Learning Object Metadata**. NJ: IEEE Standards Department, 2002.

INE. **Programas de Ensino**. INE-CTC-UFSC - Departamento de Informática e Estatística. Disponível em: <<http://admrede.inf.ufsc.br/interno/modulos/programas/aprovados.php>>. Acesso em 13 nov. 2012.

KERIEVSKY, J. **PairDraw**. Industrial Logic: 2001. Disponível em: <<http://www.industriallogic.com/blog/pairdraw-2/>>. Acesso em 10. mai. 2013.

KITCHENHAM, B. et al. **Systematic literature reviews in software engineering: A systematic literature review**. Information and Source Technology. 2009. Disponível em: <<http://portal.acm.org/citation.cfm?id=1466091>>. Acesso em 13 jul. 2010

KOLB, A., & KOLB, D. **Experiential Learning Theory: A Dynamic, Holistic Approach to Management Learning, Education and Development**. Handbook of Management Learning, Education and Development, 2008.

KOLB, D. A. **Experiential Learning**. Englewood Cliffs, NJ: Prentice Hall, 1984.

KOPER, R. **Combining reusable learning resources and services with pedagogical purposeful units of learning**. In: A. Littlejohn, *Reusing online resources: A sustainable approach to e-learning* (p. 47). London: Kogan Page, 2003.

KRIVITSKY, A. **Scrum Simulation with LEGO Bricks**. Disponível em: <[http://www.scrumalliance.org/system/resource\\_files/0000/3689/Scrum-Simulation-with-LEGO-Bricks-v2.0.pdf](http://www.scrumalliance.org/system/resource_files/0000/3689/Scrum-Simulation-with-LEGO-Bricks-v2.0.pdf)>. Acesso em 05 fev. 2013.

LEFRANÇOIS, G. R. **Teorias da aprendizagem**. São Paulo: Cengage Learning, 2008.

LÖFFLER, M. **What's your favorite Agile Game?** Scrumphony - Scrum, Kanban and Other Useful Stuff: 2010. Disponível em: <<http://blog.scrumphony.com/tag/agile-games-coaching-training-scrum/>>. Acesso em 10 mar. 2012.

LONGMIRE, W. **A Primer on Learning Objects**. 2000. Disponível em: <<http://vcampus.uom.ac.mu/orizons/html/Res270704/LOR-RLO/Longmire-RLO-primer.doc>>. Acesso em 10 out. 2012.

MCCULLOUGH, M.; MCGREAL, D. **TastyCupcakes.com: Fuel for Software Professionals**. Disponível em: <<http://tastycupcakes.org/>>. Acesso em 13 jul. 2012.

PENG, W.; HSIEH, G. **The influence of competition, cooperation, and player relationship in a motor performance centered computer game.** Computers in Human Behavior - Elsevier, 2100-2106, 2012.

PIAGET, J. **Psicologia e Pedagogia** 7ª Edição ed. Rio de Janeiro: Editora Forense Universitária LTDA, 1985.

PICHLER, R. **Agile Product Management with Scrum.** Boston: Pearson Education, 2010.

PMI. **Um Guia do Conhecimento em Gerenciamento de Projetos.** Newtown Square, Pennsylvania: Project Management Institute, Inc, 2008.

PMI. **Estudo de Benchmarking em Gerenciamento de Projetos:** Perspectiva Geral. 2009. Disponível em: < [http://www.pmi.org.br/benchmarking/2009/Relatorio\\_Final\\_do\\_Estudo\\_de\\_Benchmarking\\_em\\_GP\\_2009\\_-\\_Relatorio\\_Principal\\_-\\_Perspectiva\\_Geral.pdf](http://www.pmi.org.br/benchmarking/2009/Relatorio_Final_do_Estudo_de_Benchmarking_em_GP_2009_-_Relatorio_Principal_-_Perspectiva_Geral.pdf)>. Acesso em 08 fev. 2012.

PMSURVEY.ORG. **Relatório Nacional.** PMSURVEY.ORG: 2011.

PMSURVEY.ORG. **Relatório Nacional.** PMSURVEY.ORG: 2012.

PUCRS. **Estrutura Curricular.** PUCRS - Faculdade de Informática. Disponível em <<http://www3.pucrs.br/portal/page/portal/facinuni/facinuniCapa/facinuniGraduacao/facinuniGraduacaoSI/facinuniGraduacaoSIEstrutura>>. Acesso em 13 nov. 2012.

ROY, D.; SARKAR, S.; GHOSE, S. **A Comparative Study of Learning Object Metadata, Learning Material Repositories, Metadata Annotation & an Automatic Metadata Annotation Tool.** *Advances in Semantic Computing*, pp. 103-126, 2010.

RUBIN, K. S. **Essential Scrum** : a practical guide to the most popular agile process. Ann Arbor: Pearson Education, 2012.

SAHOTA, M. **Scrum Community Wiki.** 2009. Disponível em: < <http://scrumcommunity.pbworks.com/w/page/10148935/Games%20for%20Scrum>>. Acesso em 10 jun. 2012.

SASKATCHEWAN EDUCATION. **Instructional Approaches:** A Framework for Professional Practice. Canada: Saskatchewan Education, 1991.

SCHWABER, K. **SCRUM Development Process.** *OOPSLA'97 Business Object Workshop*, 1996.

SEFFERNICK, T. **The Agile Trip Tik.** Scrum Community Wiki, 2008. Disponível em: < <http://scrumcommunity.pbworks.com/w/page/10148889/AgileTripTik>>. Acesso em 20 jul. 2012.

SHORE, J., & WARDEN, S. **The Art of Agile Development.** Sebastopol: O'Really Media, 2008.

Silva, E. L. **Metodologia da pesquisa e elaboração de dissertação**. Florianópolis: UFSC, 2005.

SNYDER, W. BRIGGS, X. **Communities of Practice: A New Tool for Government Managers**. IBM Center for The Business of Government, 2003.

SPARTEZ. 2013. **Agile Poker for JIRA**. Disponível em: <<http://www.jiraplanningpoker.com/static/images/misc/play-schema.png>>. Acesso em 22 mai. 2013.

SUN MICROSYSTEMS. **Java Code Conventions**. Mountain View, CA, USA: Sun Microsystems, 1997.

SUTHERLAND, J. **Inventing and Reinventing SCRUM in Five Companies**. 2001. Disponível em: <<http://www.agilealliance.org/system/article/file/888/file.pdf>>. Acesso em 11 de set. de 2012.

SUTHERLAND, J.; SCHWABER, K. **The Scrum Papers: Nuts, Bolts, and Origins of an Agile Process**, 2007. Disponível em: <<http://jeffsutherland.com/scrumpapers.pdf>>. Acesso em 20 de 06 de 2012.

UFPE. **Cursos de Graduação - Sistemas de Informação**. Disponível em: <[http://www.ufpe.br/proacad/index.php?option=com\\_content&view=article&id=200%3Asistemas-de-informacao&catid=1&Itemid=138](http://www.ufpe.br/proacad/index.php?option=com_content&view=article&id=200%3Asistemas-de-informacao&catid=1&Itemid=138)>. Acesso em 13 nov. 2012.

UNISINOS. **Sistemas de Informação - Disciplinas**. Disponível em: <<http://www.unisinos.br/graduacao/sistemas-de-informacao/disciplinas>>. Acesso em 15 nov. 2012.

WAKE, W. **Scrum from Hell**. 2004. Disponível em: <<http://xp123.com/articles/scrum-from-hell/>>. Acesso em 05 jul. 2013.

WAKE, W., & COHN, M. **The Scrum Game: a Fun Interactive Tool for Learning**. 2007. Disponível em: <<http://www.mountangoatsoftware.com/pages/28-the-scrum-game-a-fun-interactive-tool-for-learning-scrum-amp-agile>>. Acesso em 07 jul. 2013.

WASKO, M. M.; FARAJ, S. "It is one does": why people participate and help others in electronic communities of practice. *Journal of Strategic Information Systems* 9, 155-173, 2000.

WENGER, E. **Communities of Practice and Social Learning Systems**. Organization articles, 225-246, 2000.

WENGER, E. **Conceptual Tools for CoPs as Social Learning Systems: Boundaries, Identity, Trajectories and Participation**. Em C. Blackmore, *Social Learning Systems and Communities of Practice* (pp. 125-143). United Kingdom: Springer, 2010.

WENGER, E.; TRAYNER, B. **Wenger-Trayner**. 2011. Disponível em: <<http://wenger-trayner.com/resources/slide-forms-of-participation/>>. Acesso em 10. jul. 2013.

WENGER, E.; MCDERMOTT, R.; SNYDER, W. M. **Cultivating Communities of Practice: a**

guide to managing knowledge. Boston: Harvard Business School, 2002.

WILEY, D. A. **Learning Object Design and Sequencing Theory**. Provo: Brigham Young University, 2000.

# Apêndice A

## Avaliação do Scrum Games Repository - SGR

Esta avaliação faz parte de uma pesquisa para desenvolver um repositório de jogos para o ensino do Scrum, realizada por João Paulo Girardi Piccinini sob a coordenação da Prof.ª. rer. nat. Christiane Gresse von Wangenheim, PMP, do laboratório do GQS / INCOD - INE - UFSC (<http://www.gqs.ufsc.br/>).

Gostaríamos de pedir a sua opinião sobre a usabilidade do repositório SGR, disponível em <http://java.inf.ufsc.br:8080/scrum-games/>, executando as tarefas descritas na próxima página e respondendo as perguntas do questionário, para que seja possível avaliar e melhorar o sistema desenvolvido.

As suas respostas, assim como os seus dados pessoais, serão mantidas em sigilo e farão parte do conjunto de dados que será utilizado para análises estatísticas.

Lembramos que a participação é voluntária e que você pode desistir do processo de avaliação a qualquer momento, sem necessidade de aviso prévio.

Se você tiver alguma dúvida pode nos procurar a qualquer momento, através do e-mail: [joapiccinini@gmail.com](mailto:joapiccinini@gmail.com)

## Tarefas para executar no SGR

Para que seja possível avaliar o SGR, pedimos que você tente executar as tarefas descritas a seguir:

1. Crie uma conta de usuário, informando seus dados pessoais;
2. Cadastre um jogo que você mesmo tenha criado ou que tenha obtido de terceiros (no site <<http://tastycupcakes.org/pt/category/games/>>, por exemplo). Caso você não se sinta confortável com esta tarefa, pode ignorá-la, pulando para a Tarefa 5.
3. Altere as informações de um jogo que você tenha cadastrado;
4. Remova um jogo que você tenha cadastrado;
5. Pesquise por jogos utilizando os filtros disponíveis;
6. Avalie um jogo cadastrado.

Obs.: algumas perguntas do questionário podem ser ignoradas, mas pedimos que você as responda caso consiga realizar a tarefa correspondente.

\*Obrigatório

## Experiência no ensino e na aplicação do Scrum

Gostaríamos de saber mais sobre a sua experiência na utilização do Scrum para o gerenciamento de projetos, e na aplicação de jogos para o ensino do Scrum em aulas ou treinamentos.

**Você já participou de um jogo sobre Scrum? Quantas vezes? \***

- Nunca.
- 1 a 2 vezes.
- 3 a 5 vezes.
- 5 a 10 vezes.
- mais de 10 vezes.

**Você já participou do desenvolvimento de um jogo para o ensino de Scrum ou de gerenciamento ágil de projetos? \***

- Sim
- Não

**Você utiliza o Scrum a quanto tempo? \***

- menos de 2 anos.
- entre 2 e 5 anos.
- entre 5 e 10 anos.
- mais de 10 anos.

\*Obrigatório

## Questionário

**Eu consegui criar a minha conta de usuário no SGR. \***

1 2 3 4 5

Discordo totalmente      Concordo totalmente

**Eu consegui realizar a busca por jogos no SGR. \***

1 2 3 4 5

Discordo totalmente      Concordo totalmente

**Eu tive dificuldades para buscar jogos no SGR. \***

1 2 3 4 5

Discordo totalmente      Concordo totalmente

**Eu consegui visualizar os resultados da busca no SGR em menos de 5 segundos. \***

1 2 3 4 5

Discordo totalmente      Concordo totalmente

**A busca por jogos no SGR atende as minhas necessidades. \***

1 2 3 4 5

Discordo totalmente      Concordo totalmente

**Eu não consegui cadastrar um jogo no SGR**

1 2 3 4 5

Discordo totalmente      Concordo totalmente

**Foi fácil cadastrar jogos no SGR.**

1 2 3 4 5

Discordo totalmente      Concordo totalmente

**Após cadastrar o jogo no SGR, eu consegui visualizar a página com as informações em menos de 5 segundos.**

1 2 3 4 5

Discordo totalmente      Concordo totalmente

**Foram exibidas informações incorretas após o cadastro do jogo no SGR.**

1 2 3 4 5

Discordo totalmente      Concordo totalmente

**As informações do jogo exibidas no SGR não atendem as minhas necessidades. \***

1 2 3 4 5

Discordo totalmente      Concordo totalmente

**Eu consegui alterar as informações de um jogo que cadastrei no SGR.**

1 2 3 4 5

Discordo totalmente      Concordo totalmente

**Eu consegui excluir um jogo que cadastrei no SGR.**

1 2 3 4 5

Discordo totalmente      Concordo totalmente

**O design gráfico do SGR é atraente. \***

1 2 3 4 5

Discordo totalmente      Concordo totalmente

**Eu aprendi a utilizar o SGR rapidamente. \***

1 2 3 4 5

Discordo totalmente      Concordo totalmente

**Eu recomendaria o SGR para meus colegas. \***

1 2 3 4 5

Discordo totalmente      Concordo totalmente

**Eu usaria o SGR para buscar jogos para o ensino do Scrum. \***

1 2 3 4 5

Discordo totalmente      Concordo totalmente

**Estou satisfeito com o resultado da busca de jogos no SGR. \***

1 2 3 4 5

Discordo totalmente      Concordo totalmente

## Avaliação geral

O que você mais gostou no SGR?

Teve algo que você não gostou e que deveria ser retirado do SGR? O que?

O que pode ser incluído ou melhorado no SGR?

A sua avaliação foi registrada com sucesso.

O laboratório do GQS-INCOD / INE / UFSC e o aluno João Paulo Girardi Piccinini agradecem a sua contribuição.

Caso deseje receber uma notificação sobre o lançamento do SGR ou sobre os resultados deste processo de avaliação, envie um e-mail para este endereço: [joaopiccinini@gmail.com](mailto:joaopiccinini@gmail.com)

## Apêndice B

### Plano de execução da avaliação

O processo de avaliação é baseado em um plano que contém a descrição da experiência dos participantes; os requisitos técnicos computacionais necessários para a leitura do guia de avaliação e para execução da aplicação; e a descrição dos passos de execução.

A aplicação da pesquisa é realizada por um painel de especialistas, composto por quatro participantes com experiência no ensino de Scrum e com participação na aplicação de jogos educativos, quatro participantes que criaram jogos educativo para o ensino de gerenciamento de projetos, dois participantes que colaboraram na criação de um repositório de jogos para o ensino de gerenciamento ágil de projetos, e três participantes com experiência prática na aplicação do Scrum e com participação na aplicação de jogos educativos.

Para que os especialistas mencionados possam avaliar o sistema, é necessário que possuam um computador com configurações mínimas a seguir:

- monitor (com resolução de 1024x768 pixels), teclado e mouse;
- acesso à *Internet*, com velocidade de download de, no mínimo, 512Kbits/seg. e upload de, no mínimo, 128Kbits/seg.;
- navegador de *Internet* compatível com a tecnologia HTML 5 (ex.: Microsoft Internet Explorer 9, Opera 10, Mozilla Firefox 3.5, Safari 4.0);
- acesso ao formulário na URL: “<https://docs.google.com>”; e
- acesso ao repositório na URL: “<http://java.inf.ufsc.br/scrum-games>”.

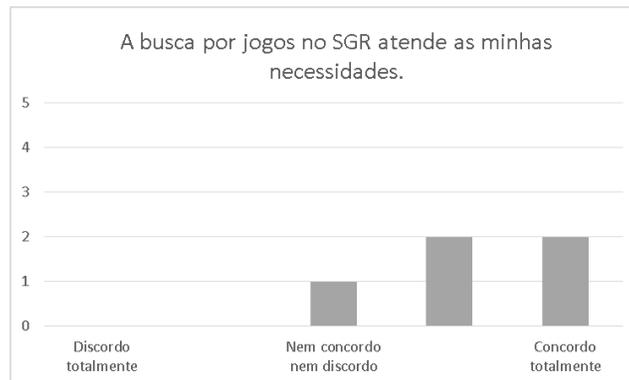
Através de um computador com as configurações mencionadas, o participante deve acessar o formulário de avaliação distribuído através da ferramenta Google Drive (pode ser encontrado no Apêndice A), ler e executar as instruções no SGR e, posteriormente, avaliar a uso do sistema através do questionário.

## Apêndice C

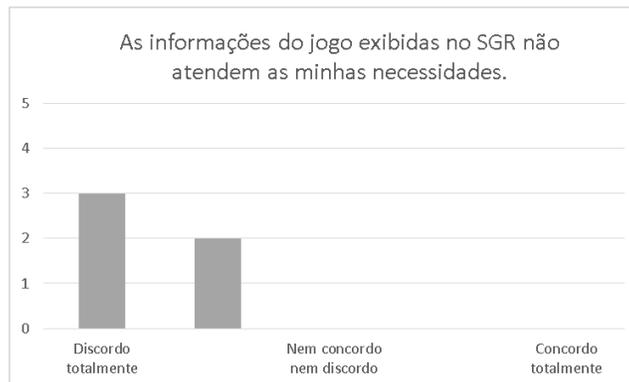
Resumo dos dados coletados com a aplicação do questionário.

Nesta seção são apresentados os gráficos referentes aos dados coletados, vinculados às perguntas do modelo GQM das quais foram extraídas as métricas (vinculadas à questão aplicada ao avaliador).

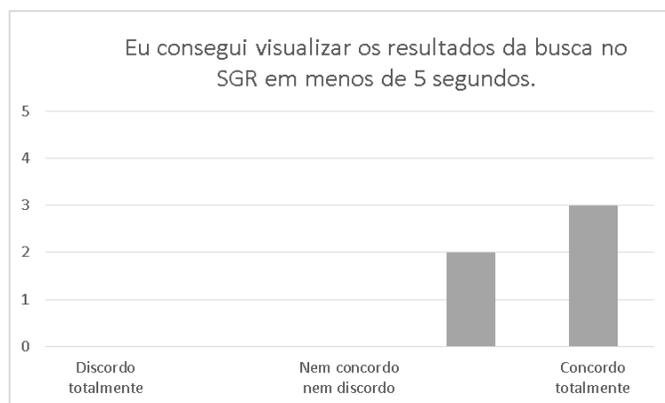
Q1. A busca por jogos atende as necessidades do usuário?



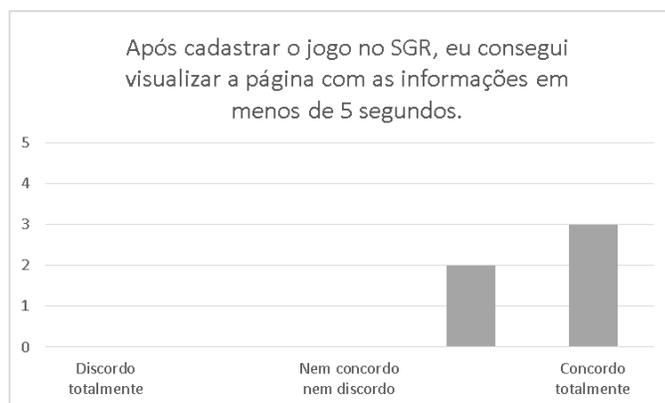
Q2. As informações exibidas na página do jogo atendem as necessidades do usuário?



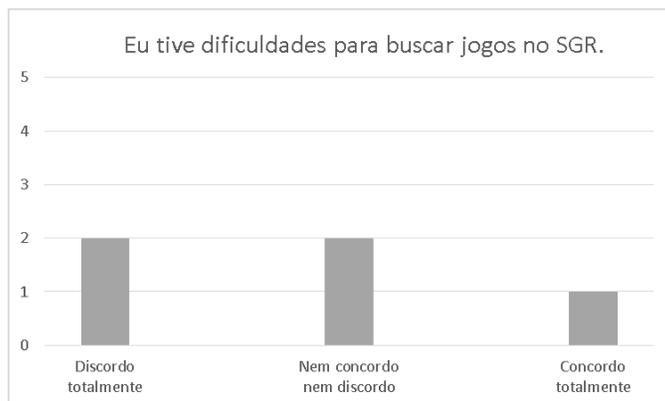
Q3. O resultado da busca foi apresentado em menos de cinco segundos?

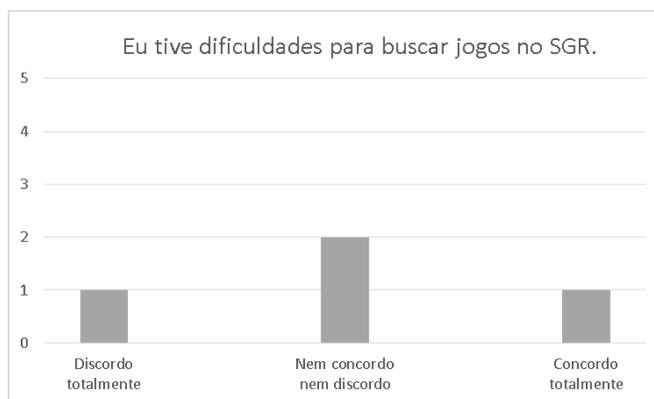


Q4. A exibição do jogo após seu cadastro foi apresentada em menos de cinco segundos?

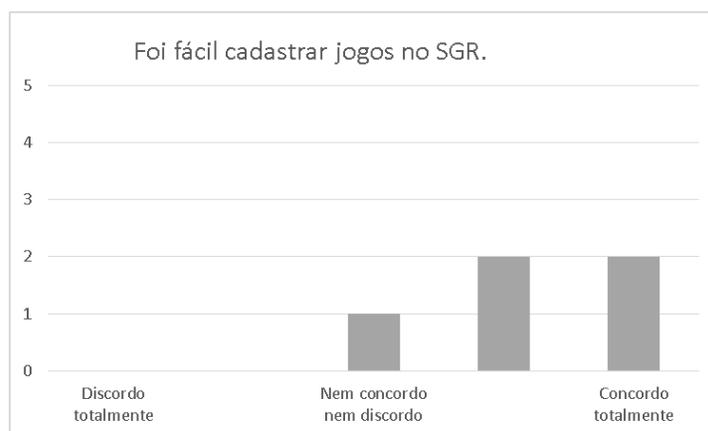


Q5. É fácil realizar a busca por jogos?

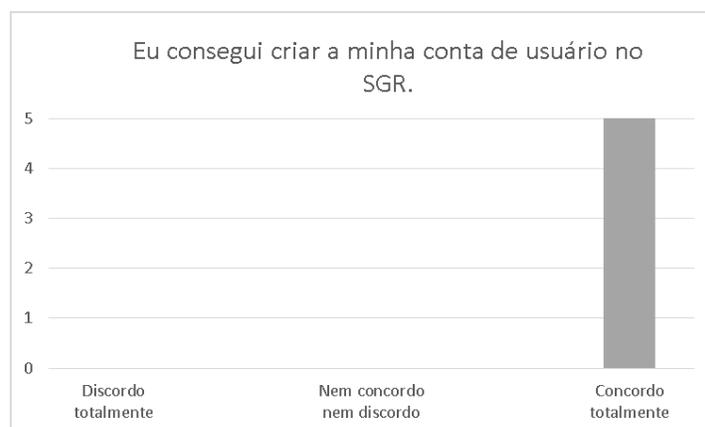




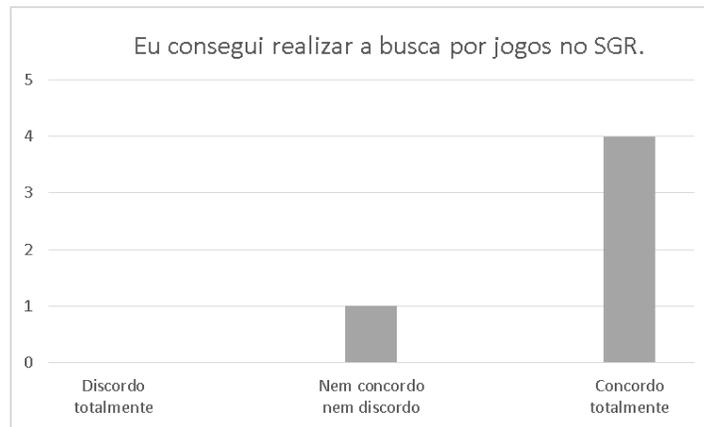
Q6. É fácil realizar o cadastro de jogos?



Q7. É possível cadastrar um usuário?



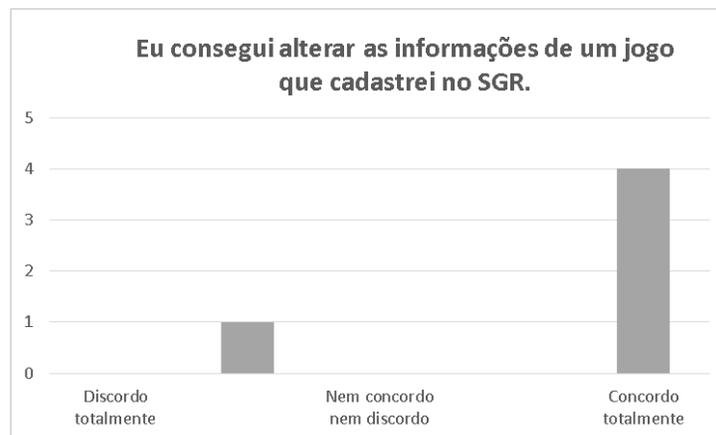
Q8. É possível realizar a busca por jogos?



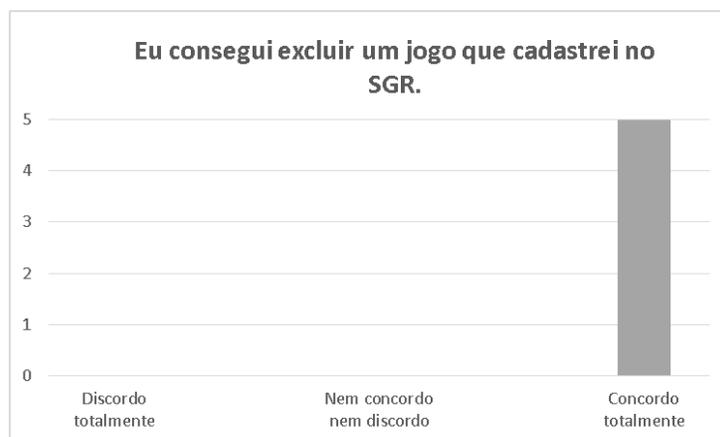
Q9. É possível cadastrar um jogo?



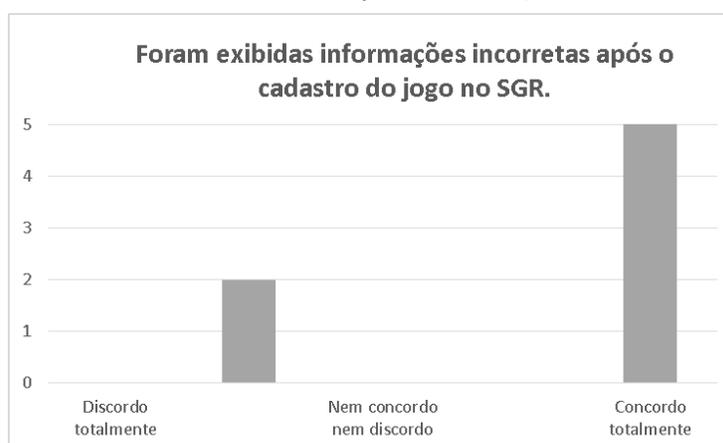
Q10. É possível alterar as informações de um jogo cadastrado?



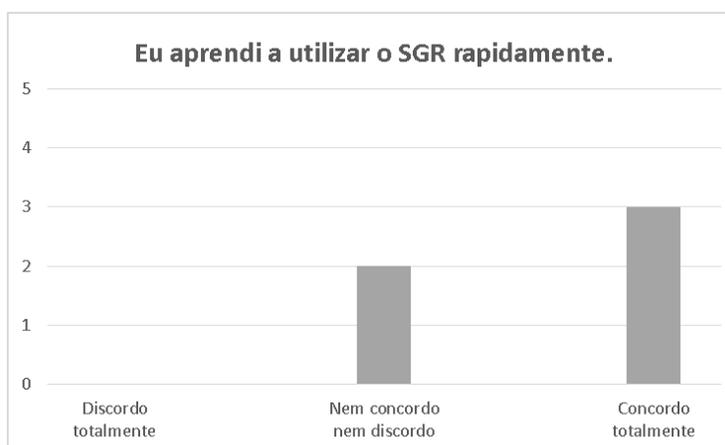
Q11. É possível excluir um jogo cadastrado?



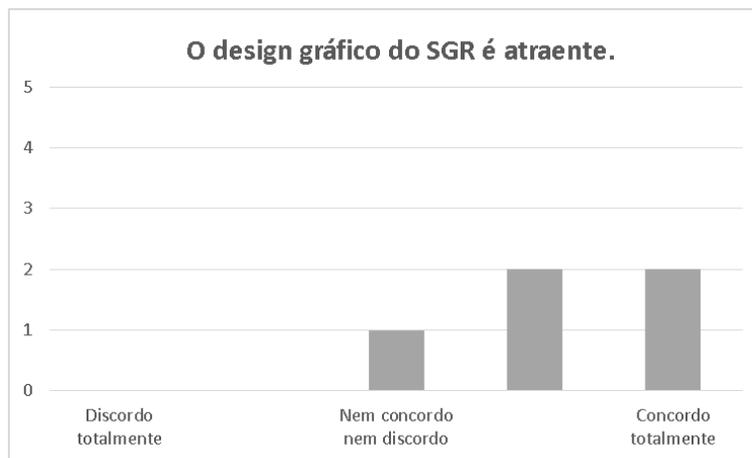
Q12. São encontrados erros nas informações exibidas após o cadastro de um jogo?



Q13. A aprendizagem na utilização do sistema é rápida?



Q14. O design gráfico do SGR é atraente?



Q15. O uso das operações disponibilizadas pelo SGR é satisfatório?

