

Abreviaturas e Símbolos

IEE	Instituto de Engenheiros Eletricistas e Eletrônicos
KA	Áreas de Conhecimento
GBL	Aprendizagem baseada em jogos
DGBL	Aprendizagem baseada em jogos digitais
SE	Engenharia de Software
CSE	Conferência Internacional de Ciência e Engenharia de Computação

Capítulo 1

Introdução

Este capítulo apresenta o contexto, e o problema e motivação deste trabalho de investigação, descrevendo os objetivos e resultados esperados do trabalho realizado.

1.1 Contexto e Motivação

A Engenharia de Software é uma área com elevada densidade de conteúdos, tendo sido por isso organizada numa base de conhecimento pelo IEEE, de forma a existir um entendimento consistente e comum a todo mundo sobre esta e esclarecer aspetos desta área como o seu âmbito ou todas as diversas áreas a que esta se encontra emparelhada. Tendo em conta tamanha diversidade de conteúdos, é lógico que, para cada um desses conteúdos, existam técnicas de ensino de Engenharia de Software que se adequam melhor, tornando mais difícil o seu ensino.

Uma das formas de melhorar a captura de conhecimentos é através de jogos, neste caso jogos sérios. Estes têm como objetivo principal a captura e ganho de conhecimento por parte do jogador, através da motivação e atenção deste aos aspetos do jogo que representam o conhecimento que se pretende adquirir. Estes métodos têm-se revelado vantajosos em relação aos métodos mais tradicionais. Visto isto, a motivação deste trabalho centra-se em identificar os padrões e técnicas de design desses jogos que servem melhor o ensino dos conteúdos da área de Gestão de Engenharia de Software, tendo em conta as diferentes técnicas e objetivos de ensino referentes a cada um desses conteúdos.

1.2 Problema

Apesar do potencial reconhecido dos jogos no ensino, duas grandes barreiras na sua implementação impedem o seu uso mais regular e em grande escala.

- Primeiro, o design destes jogos tende a ser complexo. Schell em [Sch08], refere que o design de jogos de sucesso necessita de ter em conta várias perspectivas diferentes.
- Segundo, os requisitos para um bom jogo não coincidem necessariamente com os requisitos para uma aprendizagem eficiente, especificamente neste caso de Gestão de Engenharia de Software.

Para superar estas duas barreiras pretende-se relacionar os requisitos pedagógicos da área de Gestão de Engenharia de Software com padrões de design de jogos, como forma de ajudar a desenvolver esses jogos educativos e, simultaneamente, aumentar a sua eficiência no ensino de Gestão de Engenharia de Software.

1.3 Objectivos e Resultados Esperados

O principal objetivo deste trabalho de investigação passa por produzir um mapa semântico dos objetivos educacionais e pedagógicos de Gestão de Engenharia de Software para padrões de design de jogos. Pretende-se propor uma metodologia de design de jogos sérios, usando padrões de desenho de jogos e relacionando-os com os objetivos de aprendizagem de Gestão de Engenharia de Software, reduzindo o esforço no design e desenvolvimento destes.

Como resultado final deste trabalho pretende-se que, no futuro, este trabalho sirva de base no design de jogos sérios para Gestão de Engenharia de Software, oferecendo um lista validada de padrões de design para cada um dos objetivos pedagógicos que os desenvolvedores pretendam implementar no seu jogo.

1.4 Estrutura do Documento

O presente documento encontra-se estruturado em 4 capítulos principais. O primeiro e presente capítulo serve de introdução ao documento, apresentando o contexto, motivação e problema para esta tese, assim como os seus objetivos e resultados esperados.

No segundo capítulo é apresentado o estado da arte em que se podem encontrar os tópicos de Engenharia de Software e seu Ensino, Jogos Sérios, Padrões Pedagógicos e Padrões de Design de Jogos Educativos.

No terceiro capítulo é apresentada a abordagem usada no mapeamento dos padrões de design de jogos para a área de gestão de Engenharia de Software, assim como os métodos através dos quais esse mapeamento é validado.

No quarto capítulo é apresentada a experiência empírica com estudantes usada para a validação do trabalho realizado.

Por último, no quinto capítulo são apresentadas as conclusões deste trabalho, o trabalho a realizar no futuro, e as referências utilizadas neste trabalho.

Capítulo 2

Estado da Arte

Neste capítulo faz-se um levantamento do estado da arte relativo ao ensino de Engenharia de Software e das suas áreas de conhecimento, ao uso de jogos sérios no ensino, aos padrões pedagógicos existentes, e aos padrões de design de jogos educativos. O capítulo divide-se em 5 secções:

- 2.1 Engenharia de Software e o seu Ensino – apresenta as áreas de conhecimento de Engenharia de Software sobre as quais se pretende realizar o mapeamento para padrões de design de jogos;
- 2.2 Jogos Sérios – apresenta um estudo sobre o impacto dos jogos sérios no educação em geral, e sobre os jogos existentes para o ensino de Engenharia de Software;
- 2.3 Padrões Pedagógicos – faz uma revisão dos padrões pedagógicos existentes que possam ser usados na área de Gestão de Engenharia de Software;
- 2.4 Padrões de Design de Jogos Educativos – faz uma revisão dos padrões de design de jogos educativos existentes que possam ser utilizados;
- 2.5 Resumo e Conclusões – sintetiza as conclusões retiradas do estudo feito.

2.1 Engenharia de Software e o seu Ensino

A IEEE define engenharia de software a aplicação ou estudo de uma abordagem sistemática, disciplinada e quantificável ao desenvolvimento, a operação, e a manutenção de software.

No Swebok [Swe14] são identificadas 7 áreas às quais engenharia de software se relaciona:

- Engenharia de Computadores
- Ciência da computação

- Gestão (geral)
- Matemática
- Gestão de Projetos
- Gestão de Qualidade
- Engenharia de Sistemas

Também no Swebok [Swe14] são apresentadas as 15 áreas de conhecimento (KA's) da engenharia de software, apresentadas na Tabela 1.

15 Swebok áreas de conhecimento (KA)

Requisitos de Software
 Design de Software
 Construção de Software
 Teste de Software
 Manutenção de Software
 Gestão de Configuração de Software
 Gestão de Engenharia de Software
 Processo de Engenharia de Software
 Modelos e Métodos de Engenharia de Software
 Qualidade de Software
 Prática Profissional de Engenharia de Software
 Economia de Engenharia de Software
 Fundações da Computação
 Fundações da Matemática
 Fundações da Engenharia

Tabela 1 – 15 áreas de conhecimento (KA) de Swebok [Swe14]

Destas 15 KA's, a área de Gestão de Engenharia de Software é a área sobre a qual o trabalho de investigação incide. No Swebok são identificados os seguintes tópicos como constituintes desta área:

- Iniciação e definição do âmbito: onde se encontram atividades como a determinação e negociação dos requisitos, a análise de viabilidade e a revisão dos requisitos;
- Planeamento do projeto de software: onde se encontram atividades como o planeamento do processo, a alocação de recursos, a gestão de risco, entre outros;
- Concretização do projeto de Software: onde se encontram atividades como a implementação de planos, a monitorização de processos, o controlo de processos, entre outros;
- Revisão e avaliação: onde se encontram atividades como a determinação do grau de satisfação dos requisitos e a revisão e avaliação do desempenho;

- Fecho: onde se encontram atividades como a determinação do fecho do projeto e suas atividades;
- Análise do projeto de engenharia de software: onde se encontram atividades como o planejamento do processo de medição de desempenho do projeto, a medição em si, e a avaliação dessa medição;
- Ferramentas de gestão de engenharia de software.

Os desenvolvedores de software muitas vezes não são educados através dos métodos adequados, apresentado por Shaw em [Sha00], sendo a sua educação confundida com educação para programadores e outros não engenheiros. Um dos passos que Shaw refere é identificar os papéis distintos no desenvolvimento de software e aplicar métodos educativos apropriados a cada um desses papéis. Algumas das medidas propostas são: distinção entre os diversos papéis envolvidos no desenvolvimento de software; apostar a longo prazo no ensino de áreas relacionados com o desenvolvimento de software no ensino não superior; e fornecer especialização através de treino e educação superior.

2.2 Jogos Sérios

Jogos digitais são uma forma de estimular curiosidade e motivação necessárias a um bom processo de aprendizagem. Prensky [Pre01] apresenta doze razões segundo as quais jogos digitais têm o maior potencial atrativo de todos os passatempos na história da humanidade:

1. Jogos digitais são uma forma de divertimento. Isso desperta alegria e prazer no sujeito.
2. Jogos digitais são uma forma de jogo. Isso desperta um envolvimento intenso do sujeito.
3. Jogos digitais têm regras. Isso transmite uma estrutura bem definida ao sujeito.
4. Jogos digitais têm objetivos. Isso desperta motivação no sujeito.
5. Jogos digitais são interativos. Isso desperta o sentimento de esforço no sujeito.
6. Jogos digitais são adaptativos. Isso transmite uma sensação de fluxo ao sujeito.
7. Jogos digitais têm resultados e feedback. Isso desperta aprendizagem no sujeito.
8. Jogos digitais têm estados de vitória. Isso desperta gratificação no sujeito.
9. Jogos digitais têm conflito / competição/ desafios / opositores. Isso transmite adrenalina ao sujeito.
10. Jogos digitais apresentam problemas para resolver. Isso desperta criatividade no sujeito.
11. Jogos digitais têm interação social. Isso desperta percepção social no sujeito.
12. Jogos digitais têm uma história e enredo. Isso transmite emoção ao sujeito.

Prensky refere que, apesar de existirem outros meios onde algumas destas características podem ser encontradas como filmes ou livros, jogos digitais é o único meio onde estas características estão todas presentes em simultâneo. Refere ainda que todos os jogos devem ter como objetivo principal entreter e só de seguida encorajar a aprendizagem. Por outro lado, Michael e Chen [MC06] argumentam que, em relação aos jogos sérios, que o principal objetivo é a aprendizagem de algum conhecimento por parte do sujeito e, após isso e apenas se possível, entreter o sujeito durante esse processo.

Game-based learning (aprendizagem baseada em jogos), GBL, segundo Garris et al. [GAD02], existe há algum tempo sob a forma de simuladores e motivadores, tendo estas em comum o facto de captar a atenção do sujeito numa série de processos cognitivos, provados benéficos para melhorar a atenção e, conseqüentemente, a aprendizagem em si. Os jogos educativos sob a forma de simuladores, também conhecidos como jogos sérios [SJB07] centram-se no conceito de treinar o sujeito para a aplicação do conhecimento num contexto real, sem estar diretamente exposto aos riscos associados a essa aplicação no mundo real (como o treino de pilotos de aviação através de simuladores de voo).

Os jogos educativos sob a forma de motivadores, mais usados no ensino superior e vocacional, centram-se mais num tipo de aprendizagem mais autónoma, tentando ultrapassar obstáculos como a falta de atenção, frustração, desmotivação ou falta de acompanhamento através de uma experiência de aprendizagem positiva.

Susi [SJB07] sugere ainda uma outra classificação para jogos educativos, em que os diferencia entre jogos militares, jogos educacionais, jogos corporativos e jogos de serviço de saúde, sendo que em referências mais recentes sobre jogos sérios [RCV09] jogos educacionais sejam vistos como uma subclasse de jogos sérios.

Corti [Cor06] considera GBL, mais ou menos o mesmo que jogos sérios. Segundo ele, GBL tem a capacidade de melhorar as atividades e iniciativas de treino através do empenho, motivação, dramatização e repetibilidade que lhe são reconhecidos. GBL digital (DGBL) é, segundo Prensky [Pre01], a nova “moda” do e-learning, sendo este baseado em duas premissas: primeiro, que os estudantes atuais são “nativos” no que toca a linguagem digital, alterando os seus padrões de raciocínio; segundo, estes estudantes têm uma nova experiência de interação com jogos digitais, alterando as suas preferências e habilidades, oferecendo um grande potencial de aprendizagem tanto para adultos como para crianças.

Em termos de diferença entre jogos sérios e jogos de entretenimento, Michael e Chen [MC06] analisam a situação segundo duas perspetivas: de design e de desenvolvimento desses jogos. O hardware usado na maior parte dos mercados de jogos sérios é muito anterior ao usado em jogos de entretenimento; o mercado de jogos sérios é também mais provável de possuir uma maior variedade de hardware e sistemas operativos; e o mercado de jogos sérios inclui, não só jogadores experientes, mas também novos jogadores, aumentando a necessidade deste ser muito acessível.

Enquanto jogadores experientes procuram uma experiência o mais rica e completa possível, o mercado de jogos sérios procura que o modelo ou simulação apresentada no jogo possa servir para resolver problemas reais. Desta forma, nos jogos sérios é essencial que o

foco principal seja nos elementos de aprendizagem e que os pressupostos da simulação estejam corretos, para que o sujeito não esteja a adquirir capacidades e conhecimentos errados ou inúteis. Nos jogos de entretenimento, os elementos de entretenimento já podem ser o foco principal e podem ser usadas técnicas para a simplificação dos processos de simulação, técnicas estas que, segundo Michael e Chen [MC06], devem ser repensadas para jogos sérios pois estes procuram o treino de situações reais em que a aleatoriedade pode não ser apropriada.

Também a comunicação, normalmente perfeita nos jogos de entretenimento, é diferente para os jogos sérios, pois estes devem refletir e treinar situações tendo em conta os problemas comunicativos inerentes à realidade. A tabela 2 é apresentada de seguida com as diferenças entre jogos sérios e jogos de entretenimento, sumarizadas por [SJB07].

	Jogos Sérios	Jogos de Entretenimento
Tarefa vs. Experiência Rica	Foco na resolução de problemas	Preferência por experiências mais ricas
Foco	Elementos importantes de aprendizagem	Entretenimento
Simulação	Pressupostos necessários para a aplicação da simulação	Processos de simulação simplificados
Comunicação	Deve refletir comunicação real (não perfeita)	Comunicação frequentemente perfeita

Tabela 2 – Diferenças entre jogos sérios e jogos de entretenimento [SJB07]

No que toca aos impactos positivos dos jogos digitais, Mitchell e Savill-Smith [MS04] referem que estes podem auxiliar o desenvolvimento de vários tipos de capacidades: capacidades analíticas e espaciais, capacidades estratégicas e de reconhecimento, capacidades de captação de conhecimento e aprendizagem, capacidades psicomotoras, seletividade visual, atenção, etc. Referem ainda que até jogos violentos podem ser benéficos, como forma de alívio de frustração e stress por parte do sujeito que os joga.

Enochsson et al. [Eno04] reportou impactos positivos mais específicos dos jogos digitais, tendo encontrado uma correlação positiva entre experiência em jogos digitais e performance em simulações médicas digitais efetuadas por alunos de medicina, devido à maior percepção tridimensional obtida através dos jogos digitais. Similarmente, jogos digitais podem ser usados na arquitetura e design para desenvolver nos alunos confiança, capacidades em modelagem espacial, design e criação de formas [Coy03] [Rad00]. Navarro et al. [Nav03] mostrou através de experiências que software especialmente desenhado para treino de

2.3 Padrões Pedagógicos

Um padrão é, na definição de Riehle e Zullighoven [RZ96], uma abstração de algo concreto que se repete recorrentemente num contexto não arbitrário. São uma estrutura que liga o sujeito à coleção e ao domínio em causa. Fincher e Utting, em [FU01] descreveram os requisitos funcionais e não funcionais comuns a todos os padrões, apresentados na tabela 3 em que o requisito funcional à esquerda faz par com o requisito não funcional à sua direita.

<i>Requisitos Funcionais</i>	<i>Requisitos Não Funcionais</i>
Captura de Práticas	Não Óbvio (não pode ser algo que seja obrigatório e/ou comum)
Abstração	Compreensão
Sistema de Valor	-
Princípio Estrutural	Gerador
Forma de Apresentação	Poder Comunicativo

Tabela 3 – Requisitos funcionais e não funcionais de padrões [FU01]

A utilização destes padrões na educação é muito importante e comum. Foram capturados, através do Pedagogical Patterns Project [PPP15], uma grande amostra de padrões essenciais à aplicação de certas estratégias pedagógicas. Para o bom funcionamento destes padrões, foi identificado o seguinte pré requisito: sendo o sujeito o foco central do ensino, a motivação do sujeito é necessária a uma aprendizagem eficaz.

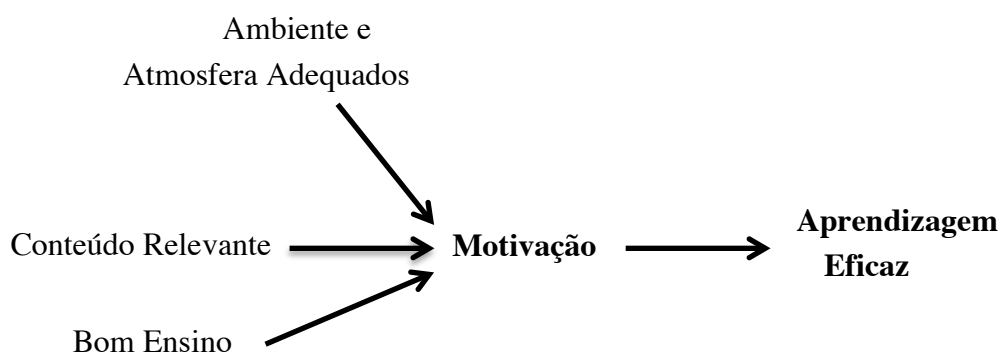


Figura 2 - Pré requisitos para aprendizagem eficaz, traduzido de [PPP15]

Alguns dos muitos exemplos de padrões referidos no Pedagogical Patterns Project são: “Built-In Failure”, que descreve o método de criar um ambiente de ensino em que as falhas são esperadas, removendo assim o medo de falhar como barreira à aprendizagem tornando-o parte do processo; “Mission Impossible”, que descreve o método de apresentar ao aluno um problema aparentemente simples de resolver mas em que a sua solução completa requer uma compreensão mais profunda da matéria; “Solution Before Abstraction”, que descreve o método apresentar ao aluno um exemplo do problema num contexto em que ele se sinta confortável; entre muitos outros. Todos estes padrões têm em comum serem diretamente aplicáveis a uma situação real de ensino.

Grösser [Gro07], com base no trabalho de Shuell e Moran [SM94], criou uma lista de 22 funções de aprendizagem e ensino que supostamente englobam toda a área pedagógica, funções estas que referem atividades cognitivas e metacognitivas que visam melhorar a eficácia da aprendizagem. Grösser reagrupou as funções de acordo com os seus diferentes tipos: preparação, manipulação de conhecimento, relações de ordem superior, regulação do aluno e ações produtivas. A lista das 22 funções de aprendizagem e ensino é apresentada na tabela 4.

<i>Funções de Aprendizagem</i>	<i>Funções de Ensino</i>
<i>Preparação</i>	
Ativação de conhecimento	Relembrar alunos de informações que são pré requisitos ou perguntar-se o que já é conhecido sobre o tópico a lecionar
Motivação	Persistência e contribuição do aluno deve ser nutrida
Expectativas	Alunos devem ter uma ideia geral dos objetivos da aprendizagem em questão
Atenção	Focar os alunos na informação relevante, em vez de na não relevante
<i>Manipulação de conhecimento</i>	
Codificação	Ajudar alunos a atribuir significados pessoais à nova informação
Comparação	Fazer comparações, procurando similaridades e diferenças que permitam a formação de relações de ordem superior de compreensão
Repetição	Induzir múltiplas perspetivas e fazer revisões sistemáticas
Interpretação	Ajudar alunos a converter informação de uma forma de representação para outra
Exemplificação	Motivar alunos a esclarecer os conhecimentos, usando novos exemplos
<i>Relações de ordem superior</i>	
Combinação, Integração,	Guiar os alunos a combinar os vários fragmentos de informação de forma a permitir a sua integração e síntese. O

Síntese	desenvolvimento de esquemas e tabelas são exemplos de como esta função pode ser ensinada.
Classificação	Habilitar alunos a determinar categorias de conceitos
Sumarização	Guiar alunos a escrever resumos que representem a informação
Análise	Guiar alunos a dividir o conhecimento em partes e a determinar como estas se relacionam
<i>Regulação do aluno</i>	
Feedback	Alunos têm de interpretar o feedback sobre adequação e exatidão da sua compreensão
Avaliação	Dar aos alunos a oportunidade de interpretar e avaliar o feedback, assim como a oportunidade de avaliar o seu próprio trabalho relativamente a certos critérios
Monitorização	Dar aos alunos a oportunidade de monitorizar a sua própria aprendizagem, de forma a determinar se existiu um progresso significativo
Planeamento	Ajudar alunos em métodos organizacionais para cumprimento de tarefas
<i>Ações produtivas</i>	
Geração de Hipóteses	Encorajar alunos a experimentar diferentes métodos de resolução
Dedução	Ajudar alunos a criar as suas conclusões sobre a informação dada
Explicação	Guiar alunos na construção mental e uso de modelos de causa-efeito
Aplicação	Ensinar alunos a usar métodos para resolução de exercícios e problemas
Produção e Construção	Guiar alunos a inventar um produto

Tabela 4 – Funções de aprendizagem e ensino segundo Shuell e Moran [SM94]

2.4 Padrões de Design de Jogos Educativos

Tendo já identificada uma base de padrões pedagógicos, esta secção descreve os vários padrões de design de jogos encontrados que podem satisfazer as requisitos desses padrões pedagógicos.

2.4.1 Padrões de Design de Jogos

Gamma [Gam95] refere que um padrão de design dá nome, explica a motivação, e explica um design geral que responde a um problema de design recorrente. Descreve o

problema, a solução, dicas de como implementar, em que situações aplicar a solução, e as suas consequências. Estes padrões estão descritos de forma predominantemente textual (Alexander [Ale78]), ao contrário de padrões de design de software que se focam já na implementação e código, o que lhes confere 3 grandes vantagens segundo Agerbo e Cornils [AC98]: o encapsulamento da experiência, o vocabulário comum e o melhoramento da documentação.

Björk e Holopainen [BH04] desenvolveram um grande inventário de padrões de design relevantes para jogos, cerca de 296, descrevendo blocos de desenvolvimento de um jogo. Estes padrões estão divididos em 11 categorias, segundo 4 pontos de vista sobre jogos: holística (descrição das atividades), fronteiras (descrição dos limites das atividades), tempo (descrição da ordem temporal do jogo) e visão estrutural (descrição das funcionalidades do jogo). Estas 11 categorias são apresentadas e explicadas de forma abreviada na tabela 5.

<i>Categoria do Padrão</i>	<i>Descrição</i>
Padrões para elementos de jogo	Estes padrões descrevem objetos de jogo que definem a área da realidade de jogo ou que os jogadores podem manipular (48 padrões)
Padrões para recursos e sua gestão	Estes padrões descrevem diferentes tipos de recursos que podem ser controlados pelos jogadores e pelo sistema (20 padrões)
Padrões para informação, comunicação e apresentação	Estes padrões descrevem como a informação sobre o estado de jogo é tratada, como por exemplo a ocultação de certa informação para avaliação do jogador (20 padrões)
Padrões para ações e eventos	Estes padrões controlam que tipo de ações estão disponíveis para os jogadores, como estas se relacionam com as alterações do estado do jogo, e como estas se relacionam com os objetivos dos jogadores (44 padrões)
Padrões para estruturas narrativas, previsão e imersão	Estes padrões lidam com o enredo, imersão e empenho sobre o jogo pelos jogadores (31 padrões)
Padrões para interação social	Estes padrões cobrem como os jogos suportam a interação social entre os jogadores (30 padrões)
Padrões para objetivos	Padrões que lidam com os objetivos que os jogadores pretendem atingir (26 padrões)
Padrões para estruturas objetivo	Estes padrões descrevem como a jogabilidade afeta os objetivos (20 padrões)
Padrões para sessões de jogo	Estes padrões lidam com as características das instâncias e sessões de jogo e as suas limitações, possibilidades e atributos da participação do jogador no jogo (20 padrões)
Padrões para mestria do jogo e equilíbrio	Estes padrões descrevem como os jogadores podem usar as suas habilidades e experiência

Padrões para repetibilidade de meta jogos e curvas de aprendizagem	de jogo e como é possível balancear a jogabilidade para jogadores com diferentes habilidades (27 padrões) Estes padrões lidam com os problemas com vêm de fora da jogabilidade de um só jogador (10 padrões)
--------------------------------------------------------------------	---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

Tabela 5 – Categorias de padrões de Björk e Holopainen [BH04]

Para cada um destes padrões, Björk e Holopainen apresentam os seguintes componentes: uma descrição geral, informação sobre como usar o padrão, exemplos, uma descrição das consequência da sua aplicação e informação sobre conflitos deste padrão com outros. Este último pormenor é muito importante pois, para formar um jogo, os padrões precisam de ser combinados com outros padrões, nunca aparecendo sozinhos numa aplicação.

Tendo em conta os padrões anteriores, estes possuem 3 restrições para a sua interligação: instanciação e modelação, no que toca à forma de ligação entre eles; conflitos, no que toca aos padrões com que entra em conflito lógico. Na descrição de cada padrão são apresentados todos os padrões a que este pode ser ligado, assim como todos os padrões com que entra em conflito, podendo-lhe ser atribuídos graus de conectividade e de conflito (número de padrões com os quais se liga e que entra em conflito respetivamente).

2.4.2 Mapeamento de Padrões de Design de Jogos na Pedagogia

Kelle et al em [KKS11] estabeleceram um mapeamento entre as funções de aprendizagem e ensino de Shuell e Moran referidas anteriormente (Tabela 4) e os padrões de design de jogos de Björk e Holopainen (Tabela 5). Tendo em conta que existe uma dependência sobre variáveis como o contexto, o domínio, o cenário e o tipo de aprendizagem, foi analisado apenas um cenário específico, de forma a conseguir identificar quais os padrões de design de jogos com mais relevância para uma aplicação universal, utilizando apenas os padrões com um maior nível de abstração e grau de conectividade (para consequente instanciação de padrões mais detalhados).

Com base nas considerações anteriores, foi criado o processo de mapeamento apresentado de seguida, tendo também em conta as seguintes taxonomias pedagógicas: design pedagógico de Heinich et al. [Hei01], eventos de instrução de Gagné [Gag65], Objetivos pedagógicos de Long e Robinson (1998), atividades de aprendizagem de Kolb [Kol84], e modelo para atenção, relevância, confiança e satisfação (ACRS) de Keller [Kel83].

1. Começar das funções de aprendizagem e ensino de Shuell e Moran.
2. Ligar as funções de aprendizagem e ensino às taxonomias pedagógicas referidas, através de sobreposição semântica.
3. Utilizando as ligações do ponto 2, identificar os requisitos pedagógicos de cada função de aprendizagem e ensino.

4. Escolha dos padrões de design de jogos que melhor suportam os conceitos pedagógicos da função de aprendizagem e ensino, usando o inventário de Björk e Holopainen.

Para validar este procedimento de mapeamento, Kelle et al. (2011) testaram uma amostra de 11 padrões de design de jogos, um de cada uma das 11 categorias, pedindo a 10 especialistas na área pedagógica para avaliar, seguindo os passos apresentados, o suporte de cada um desses padrões a cada função de aprendizagem de Shuell e Moran, de 1 (menor suporte) a 5 (maior suporte). Os 11 padrões escolhidos, escolhidos devido a parecerem mais promissores no que toca a atividades de aprendizagem, foram: “score”, “resources”, “asymmetric information”, “surprises”, “roleplay”, “gain information”, “randomness”, “levels”, “clues”, “time limits” e “replayability”. Depois de analisar os resultados tendo em conta o nível de usabilidade de cada padrão para cada cenário pedagógico (utilizando médias de classificação e ANOVA) e o grau de concordância dos especialistas (utilizando coeficiente de correlação), o método de mapeamento foi provado viável.

Kelle et al aplicaram de seguida o método de planeamento ao inventário completo de padrões. Os resultados são as 5 tabelas apresentadas de seguida, em que a primeira coluna representa as funções de aprendizagem e ensino em questão, a segunda coluna identifica os conceitos pedagógicos subjacentes, e a terceira coluna o mapeamento para as classes de padrões de design de jogos associadas e exemplos de padrões. Estes padrões são identificados pelo seu nome e um triplo que indica a sua categoria (segundo a Tabela 5), o seu grau de conectividade e grau de conflito.

<i>Função de Aprendizagem</i>	<i>Taxonomia Subjacente</i>	<i>Classe de Padrões de Design</i>
Ativação de conhecimento	“Retrieval” de Gagné	Padrões para objetivos, exemplo “reconnaissance” (7,18,1)
Motivação	ACRS de Keller	Padrões relacionados com pontuação, exemplo “rewards” (4, 54, 1)
Expectativas	“Expectancy” de Gagné	Padrões relacionados com objetivos, exemplo “predefined goals” (8, 10, 2), e padrões de narrativa, exemplo “anticipation” (5, 22, 2)
Atenção	Atenção listada por Keller e Gagné	Padrões de elementos de jogo, exemplo “clues” (1, 19, 3)

Tabela 6 – Mapeamento das funções de aprendizagem de “Preparação” [KKS11]

<i>Função de Aprendizagem</i>	<i>Taxonomia Subjacente</i>	<i>Classe de Padrões de Design</i>
Codificação	Conceito de conceptualização abstrata de Kolb	Padrões para informação
Comparação	Conceito de observação refletiva de Kolb	Padrões para informação
Repetição	Conceito de confiança de Kolb, design de exercício e prática de Heinich	Padrões para meta jogos, exemplo “replayability” (11, 23, 8)
Interpretação	Objetivo pedagógico de encorajamento de múltiplas perspectivas de Robinson	Padrões para objetivos, exemplo “gain information” (7, 21, 1)
Exemplificação	Objetivo pedagógico de encorajamento de múltiplas perspectivas de Robinson	Padrões para elementos de jogo, exemplo “levels” (1, 24, 0)

Tabela 7 – Mapeamento das funções de aprendizagem de “Manipulação de conhecimento” [KKS11]

<i>Função de Aprendizagem</i>	<i>Taxonomia Subjacente</i>	<i>Classe de Padrões de Design</i>
Combinação, Síntese	Integração, Objetivo pedagógico de ganho de múltiplas perspectivas de Robinson	Padrões para objetivos, exemplo “gain information” (7, 21, 1)
Classificação	Conceito de relevância de Keller, conceito de conceptualização de Kolb	Padrões informação e comunicação, exemplo “perfectinformation”(3,16, 8)
Sumarização	Evento para obter desempenho de Gagné, design de apresentação de Heinich	Padrões para informação e comunicação, exemplo “communication channels” (3, 10, 0); padrões para sessões de jogo, exemplo “time limits” (9, 39, 4)
Análise	Design de resolução de problemas de Heinich	Padrões para mestria de jogo e equilíbrio, exemplo “strategic knowledge” (10, 48, 1)

Tabela 8 – Mapeamento das funções de aprendizagem de “Relações de ordem superior” [KKS11]

<i>Função de Aprendizagem</i>	<i>Taxonomia Subjacente</i>	<i>Classe de Padrões de Design</i>
Feedback	Evento de fornecer feedback de Gagné	Padrões relacionados com pontuação, exemplo “score” (4, 18, 0); padrões para mestria de jogo e equilíbrio, exemplo “near miss indicators” (10, 30, 5); padrões para informação, exemplo “progress indicators” (3, 21, 2)
Avaliação	Evento de fornecer feedback de Gagné, objetivo de auto percepção da construção de conhecimento de Robinson	Padrões para informação, exemplo “status indicators” (3, 14, 2); Padrões relacionados com pontuação, exemplo “rewards” (4, 54, 1)
Monitorização	Evento de fornecer guia a aprendizagem de Gagné	Padrões para informação, exemplo “status indicators” (3, 14, 2); Padrões relacionados com pontuação, exemplo “rewards” (4, 54, 1) e “penalties” (4, 51, 3)
Planeamento	Conceito de exploração concreta de Kolb	Padrões para mestria de jogo e equilíbrio, exemplo “Stimulated planning” (10, 51, 0)

Tabela 9 – Mapeamento das funções de aprendizagem de “Regulação do aluno” [KKS11]

<i>Função de Aprendizagem</i>	<i>Taxonomia Subjacente</i>	<i>Classe de Padrões de Design</i>
Geração de Hipóteses	Design de descoberta e resolução de problemas de Heinich	Padrões para interação, exemplo “exploration” (6, 33, 1)
Dedução	Design de descoberta e resolução de problemas de Heinich	Padrões para estruturas objetivo, exemplo “player defined goals” (8, 27, 2)
Explicação	Design de apresentação, demonstração e tutoriais de Heinich, evento de fornecer guia a aprendizagem de Gagné	Padrões para informação, exemplo “direct information” (3, 15, 5); padrões para elementos de jogo, exemplo “clues” (1, 19, 3) e “helpers” (1, 10, 1)
Aplicação	Conceito de exploração e experimentação de Kolb	Padrões de elementos de jogo, exemplo “clues” (1, 19, 3)

Produção e Construção	Posse de conhecimento e conceito de experimentação de Kolb	Padrões de imersão, exemplo “creative control” (5, 27, 0)
-----------------------	------------------------------------------------------------	-----------------------------------------------------------

Tabela 10 – Mapeamento das funções de aprendizagem de “Ações produtivas” [KKS11]

Kelle et al. (2011) conclui que as consequências deste trabalho de mapeamento são a criação de 3 diferentes perspectivas na aplicação de padrões de design de jogos na educação: primeiro, o enriquecimento pedagógico de jogos comerciais já existentes de forma a criar uma abordagem coerente de aprendizagem (Van Eck, 2006); segundo, o enriquecimento das técnicas educativas em prática com os padrões de design de jogos; terceiro, o desenvolvimento de métodos e ferramentas para identificar padrões de design de jogos ocultos nas práticas educacionais existentes.

2.4.3 Padrões de Design de Conteúdos de Jogos

Maciuszek et al. [MLM11] elaboraram um estudo sobre padrões de design de conteúdo de jogos. Tendo já demonstrado, em estudos anteriores [HM07], um padrão que pode ser transferido para jogos educacionais do tipo RPG, o estudo foi desenvolvido com base numa arquitetura, proposta por Maciuszek e Martens em [MM10] para o desenvolvimento de jogo educativos, apresentada na figura 3.

A análise incidiu sobre padrões de design para 4 tipos de conteúdos: “quest” (missão), “character” (personagem), “environment” (ambiente) e “item”, apresentados na Tabela 11.

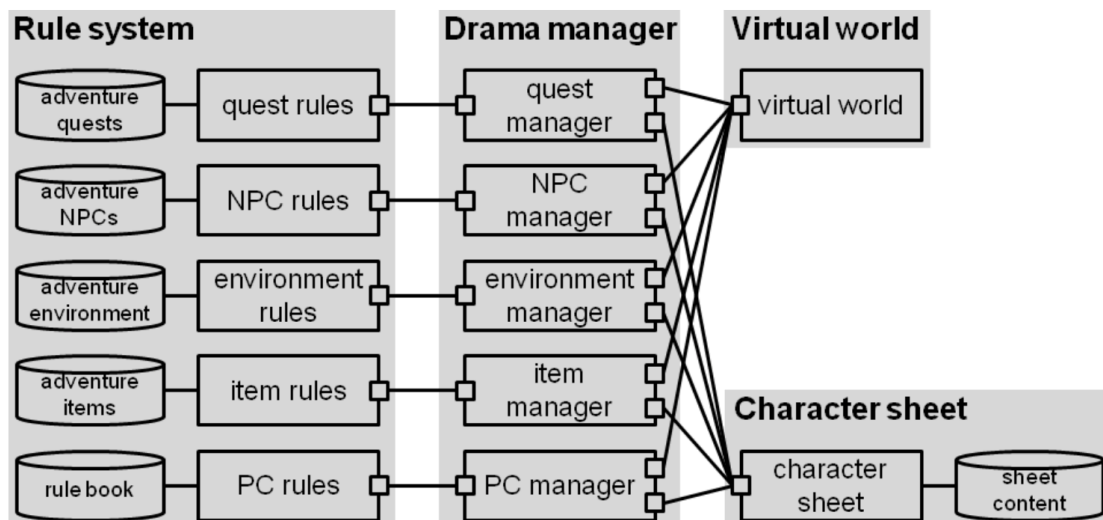


Figura 3 – Arquitetura de RPG educacional, por Maciuszek e Martens [MM10]

<i>Padrão de Design Conteúdo</i>	<i>de de</i>	<i>Importância</i>	<i>Resultado</i>	<i>Discussão / Aplicação</i>
“Quest” (missão)		Conhecimento humano representado por histórias, Schank [SC95]; história de nível macro e micro (micro importante nas unidades de aprendizagem), Hallford e Hallford [HH01]	13 padrões de design, exemplos: “Coming out of the Closet”, “Declaration of Love”, “Interrupted Wedding / Runaway Bride”, etc	Integração com conteúdo educacional fácil e não requer estruturas separadas, podendo qualquer dos padrões atuar com caminhos de aprendizagem
“Character” (personagem)		Boas personagens necessárias para estabelecer ligação forte, Crawford [Cra05]; perspectiva sobre o gênero das personagens	Padrões de design de heróis de ação (comum masculino); padrões de design de soldado, “sexy” e frágil (comum feminino); entre outros	Mais pesquisas são necessárias nas aplicações destes designs aos jogos
“Environment” (ambiente)		Taxonomia proposta por Holmer e Jödick [HJ04] sobre propósitos de ferramentas colaborativas: “awareness”, “steer”, “repository”, “editor” e “complement”	6 padrões de design: “Auction House”, “Battleground”, “Guilt Bank”, “Guilt Castle”, “Housing” e “Mailbox”	Resultados interessantes, catálogo de padrões de design arquiteturais por Alexander et al. (1977) pode ser útil
“Item”		Para garantir a interatividade, facilidade de experimentação e variedade de estratégias, os jogos necessitam de itens (objetos usáveis de pequeno tamanho), Maciuszek e Martens [MM09]	5 padrões de design de itens: “tool”, “reference material”, “map”, “meter” e “souvenir”	Padrões de design de “item” diretamente relacionados com os de “environment”; padrão de design “souvenir” descoberta interessante, devido à pretensão dos jogadores em guardar troféus e experiências (importante para jogos educativos)

Tabela 11 – Padrões de design de Conteúdos [MLM11]

Utilizando este catálogo de padrões, é possível criar um grande leque de histórias interativas ligadas a conteúdo educacional.

2.5 Resumo e Conclusões

Neste capítulo encontra-se documentado todo o estudo realizado sobre o Estado da Arte referente a este trabalho de investigação. O estudo focou-se no estudo do ensino de Engenharia de Software e das suas áreas de conhecimento, no uso de jogos sérios no ensino, nos padrões pedagógicos existentes, e nos padrões de design de jogos educativos.

Das áreas de conhecimento de Engenharia de Software, a área de Gestão de Engenharia de Software foi analisada a um nível mais detalhado, tendo sido identificados os seus diversos tópicos para uso no trabalho de investigação.

De seguida, foram analisados os jogos sérios, descrevendo as características e componentes que os distinguem dos restantes jogos. Foram também analisados o seu impacto em processos pedagógicos. Por fim, foi analisado um jogo sério desenvolvido para o ensino de Engenharia de Software, o SimSE, tendo sido identificado como possível ferramenta de validação do mapeamento dos padrões de design de jogos para a área de Gestão de Engenharia de Software (área na qual o jogo baseia o seu objetivo pedagógico).

Por fim, foram apresentados os estudos sobre os padrões pedagógicos e os padrões de design de jogos existentes. Primeiro, foram estudados os padrões pedagógicos, tendo sido identificadas as funções de ensino e aprendizagem de Shuell e Moran. Em segundo lugar, foram estudados os padrões de design de jogos, onde foi encontrado um mapeamento já validado, realizado por Kelle et al, das funções de ensino e aprendizagem de Shuell e Moran (referidas anteriormente) para a coleção de padrões de design de jogos de Björk e Holopainen.

Em resumo, foi encontrado um mapeamento já validado de funções pedagógicas para padrões de design de jogos, sendo agora necessário para este trabalho de investigação a sua validação para a área de Gestão de Engenharia de Software. Como possível ferramenta para esta validação foi identificado o jogo sério SimSE, desenvolvido para o ensino de Engenharia de Software.

Capítulo 3

Mapeamento dos Padrões

Este capítulo apresenta o processo de mapeamento dos padrões de design de jogos para os objetivos pedagógicos da área de Gestão de Engenharia de Software. É detalhada a abordagem escolhida para esse mapeamento, assim como os processos usados para estabelecer e validar esse mapeamento. Assim, o capítulo divide-se em 4 seções:

- 3.1 Abordagem do Mapeamento;
- 3.2 Relação entre Gestão de Engenharia de Software e Funções de Ensino e Aprendizagem;
- 3.3 Relação entre Gestão de Engenharia de Software e Padrões de Design de Jogos;
- 3.4 Resumos e Conclusões.

3.1 Abordagem do Mapeamento

A abordagem seguida para estabelecer a relação entre os objetivos pedagógicos da área de Gestão de Engenharia de Software e os padrões de design de jogos consiste em 3 passos, representados na figura 4.

O primeiro passo foi a identificação de padrões de design de jogos apropriados para determinadas funções de ensino e aprendizagem (Aresta A da figura 4). Esta relação foi conseguida através do artigo de Kelle et al. [KKS11], apresentado no capítulo anterior, que para cada uma das 22 funções de ensino e aprendizagem de Shuell e Moran [SM94] apresenta (e valida) a classe de padrões de design de jogos do inventário de Björk e Holopainen apropriada para a sua implementação.

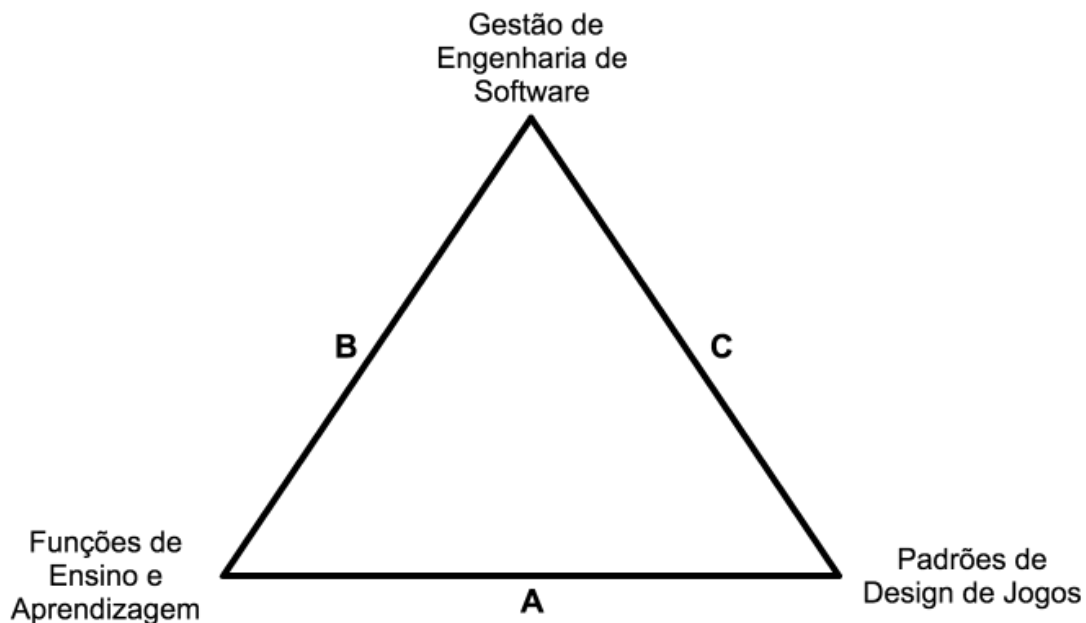


Figura 4 – Abordagem ao mapeamento.

Assim, de forma a utilizar esse mapeamento feito por Kelle na área de conhecimento de Gestão de Engenharia de Software, foi necessário relacionar esta área às funções de ensino e aprendizagem (Aresta B da figura 4) e aos padrões de design de jogos utilizados por Kelle (Aresta C da figura 4).

3.2 Relação entre Gestão de Engenharia de Software e Funções de Ensino e Aprendizagem

Devido à ambiguidade das funções de ensino e aprendizagem de Shuell e Moran, para estabelecer a relação dessas funções com a área de Gestão de Engenharia de Software (aresta B da figura 4) foi necessário recorrer a especialistas no ensino da área.

Desta forma, foi enviada uma matriz (apresentada no Anexo A) a professores de Engenharia de Software de várias instituições de ensino nacionais e internacionais. No preenchimento dessa matriz, pediu-se aos professores que selecionassem, para cada um dos tópicos de Gestão de Engenharia de Software, as funções de aprendizagem que considerassem ter maior importância no ensino desse tópico.

Apesar de terem sido obtidas apenas 5 respostas, estas provieram de diferentes instituições de ensino, a nível nacional e internacional. Desta forma, foram só consideradas relevantes e válidas as funções de aprendizagem que obtiveram 100% de concordância entre

os 5 especialistas. Os resultados são apresentados na tabela 12 (em inglês pois a matriz enviada de onde foi retirada está também em inglês).

Learning Functions	Software Engineering Management Knowledge Areas						
	Initiation and Scope Definition	Software Project Planning	Software Project Enactment	Review and Evaluation	Closure	Software Engineering Measurement	Software Engineering Management Tools
Prior knowledge activation	0%	0%	60%	0%	20%	40%	40%
Motivation	0%	40%	40%	0%	0%	0%	20%
Expectations	60%	60%	40%	40%	40%	40%	40%
Attention	20%	80%	60%	100%	20%	100%	20%
Encoding	20%	40%	40%	20%	40%	0%	0%
Comparison	40%	60%	40%	40%	0%	40%	0%
Repetition	0%	40%	80%	20%	0%	0%	60%
Interpreting	0%	40%	0%	100%	60%	60%	0%
Exemplifying	60%	40%	20%	40%	0%	40%	60%
Combination, integration, synthesis	0%	20%	0%	60%	60%	60%	0%
Classifying	60%	60%	20%	40%	40%	60%	20%
Summarising	40%	60%	0%	60%	20%	40%	0%
Analysing	40%	40%	20%	100%	0%	100%	0%
Feedback	100%	100%	80%	40%	0%	40%	0%
Evaluation	40%	40%	40%	100%	60%	40%	0%
Monitoring	40%	80%	100%	40%	20%	100%	20%
Planning	0%	80%	40%	40%	0%	0%	20%
Hypothesis generation	60%	40%	40%	0%	0%	0%	0%
Inferring	20%	20%	40%	60%	20%	40%	0%
Explaining	0%	40%	60%	100%	20%	40%	0%
Applying	40%	40%	60%	40%	100%	60%	100%
Producing and constructing	0%	0%	0%	0%	0%	0%	0%

Tabela 12 – Resultados da matriz preenchida pelos especialistas.

Como se pode observar na tabela 12, para cada tópico da área de Gestão de Engenharia de Software, existe **pelo menos uma função de aprendizagem** que reúne 100% de concordância por parte dos especialistas no que toca à sua importância no ensino desse tópico.

A tabela 13 apresenta as funções validadas pelos especialistas para cada um dos tópicos da área de Gestão de Engenharia de Software, assim como a classe de padrões de design de jogos apropriada para a sua implementação segundo Kelle. Esta tabela representa o resultado do mapeamento dos objetivos pedagógicos da área de Gestão de Engenharia de Software para os padrões de design de jogos encontrados.

<i>Tópico Gest. Eng. Software</i>	<i>Função de Aprendizagem</i>	<i>Padrão de Design de Jogos</i>
Iniciação e definição do âmbito	Feedback	Padrões para ações e eventos (ex: <i>score pattern</i>) e padrões para informação (ex: <i>progress indicator pattern</i>)
Planeamento do projeto de software	Feedback	Padrões para ações e eventos (ex: <i>score pattern</i>) e padrões para informação (ex: <i>progress indicator pattern</i>)
Concretização do projeto de Software	Monitorização	padrões para informação (ex: <i>status indicator pattern</i>)
Revisão e avaliação	Atenção, Interpretação, Análise, Avaliação e Explicação	Padrões para estruturas narrativas (ex: <i>surprises pattern</i>), padrões para objetivos (ex: <i>predefined goals pattern</i>), padrões para mestria do jogo e equilíbrio (ex: <i>randomness pattern</i>), padrões para informação (ex: <i>status indicator</i>)
Fecho	Aplicação	Padrões para elementos de jogo
Análise do projeto de engenharia de software	Atenção, Análise e Monitorização	Padrões para estruturas narrativas (ex: <i>surprises pattern</i>), padrões para mestria do jogo e equilíbrio (ex: <i>randomness pattern</i>) e padrões para informação (ex: <i>status indicator pattern</i>).

Ferramentas de gestão de engenharia de software	Aplicação	Padrões para elementos de jogo
-------------------------------------------------	-----------	--------------------------------

Tabela 13 – Mapeamento dos objetivos pedagógicos de Gestão de Engenharia de Software para os padrões de design de jogos.

3.3 Relação entre Gestão de Engenharia de Software e Padrões de Design de Jogos

De forma a validar a utilização das classes de padrões de design de jogos referidos no mapeamento apresentado na tabela 13, foi necessário estabelecer a relação direta entre esses padrões e a área de Gestão de Engenharia de Software (aresta C da figura 4). Para tal, foi encontrado o jogo SimSE, desenvolvido com o principal objetivo do ensino de Engenharia de Software, e em especial da área de gestão de projetos de software.

O jogo SimSE foi analisado, usando a biografia de Navarro [Nav06], de forma a concluir se as classes padrões de design referidos podiam ser nele encontrados.

- **Padrões para ações e eventos:** “*score pattern*” pode facilmente ser identificado na figura 5 na apresentação da pontuação final após entregar o projeto ao cliente no jogo;

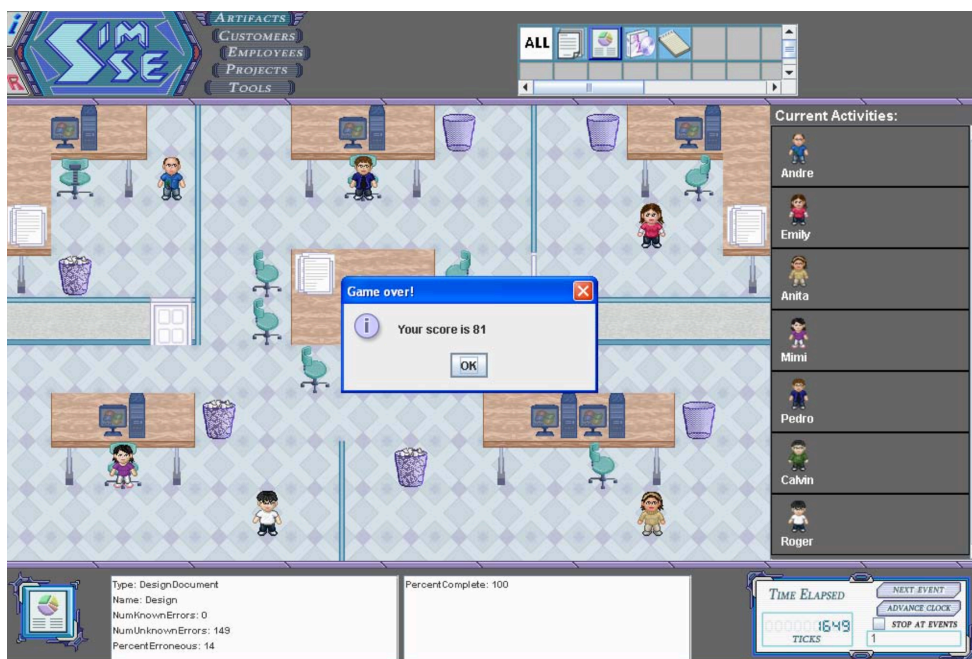


Figura 5 – Exemplo do “*score pattern*” no SimSE

- **Padrões para informação:** “*progress indicator pattern*” e “*status indicator pattern*” diretamente identificados na implementação dos indicadores do estado da equipa (figura 6) e dos documentos de entrega e tempo gasto (figura 7);

SoftwareEngineers:							
Name	Energy	Mood	Requirement...	DesignExperi...	CodingExperi...	TestingExperi...	PayRate
Andre	1.00	1.00	10 years	11 years, con...	7 years, fast ...	9 years	35.00
Emily	0.70	0.80	3 years	5 years	6 years	1.5 years	30.00
Anita	0.58	0.54	8 years	5 years	2 years, hate...	6 months	33.00
Mimi	0.88	0.74	3 months, be...	5 months, be...	3 months, be...	8 years, testi...	20.00
Pedro	0.38	0.34	7 years	2 years, but h...	8 years	15 years	28.50
Calvin	0.99	0.99	9 years, con...	8 months	8 years	2 weeks	32.00
Roger	0.24	0.77	Beginner	Beginner	Beginner	Beginner	10.00

Figura 6 – Indicadores do estado da equipa.

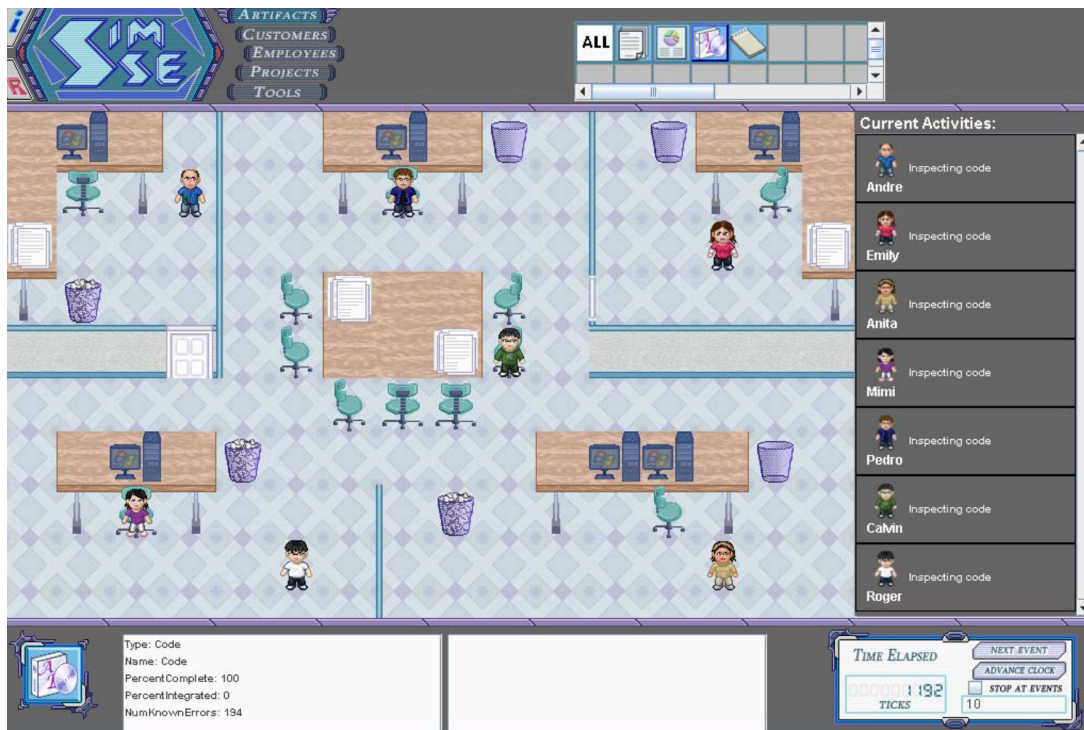


Figura 7 – Indicadores do estado dos documentos de entrega e tempo gasto (representados na parte debaixo da figura).

- **Padrões para estruturas narrativas:** “*surprises pattern*” diretamente identificado pela apresentação de novos requisitos aleatórios e inesperados por parte do cliente (figura 8);
- **Padrões para mestria do jogo e equilíbrio:** “*randomness pattern*” diretamente identificado pela apresentação de novos requisitos aleatórios e inesperados por parte do cliente (figura 8);

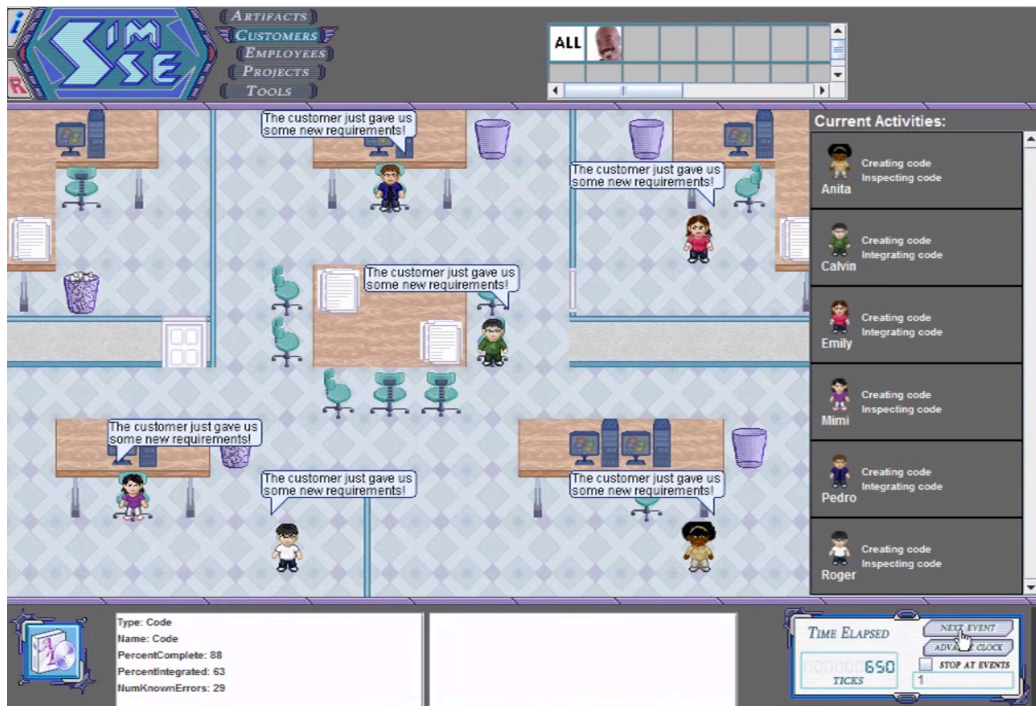


Figura 8 – Apresentação inesperada e aleatória de novos requisitos por parte do cliente.

- **Padrões para objetivos:** “*predefined goals pattern*” diretamente presente no jogo na forma com este define inicialmente qual o tempo e orçamento a cumprir para o projeto apresentado;
- **Padrões para elementos de jogo:** padrões presentes em todos os elementos de jogo manipuláveis pelo jogador, como por exemplo os membros da equipa (manipulações possíveis apresentadas na figura 9).

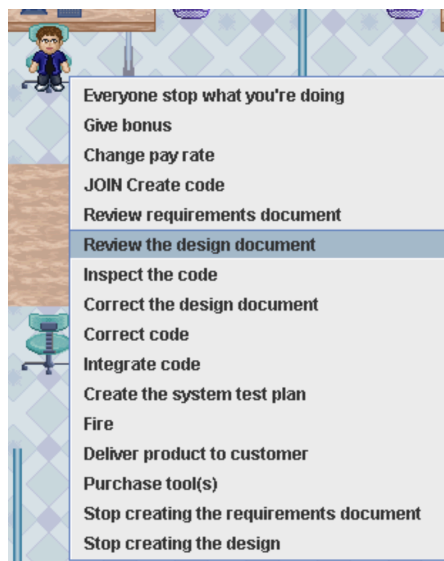


Figura 9 – Ações possíveis aplicáveis a um membro da equipa de desenvolvimento.

Confirmada a presença dos padrões de design no jogo SimSE, foi realizada uma experiência empírica com alunos, apresentada no capítulo seguinte, de forma a validar na prática o uso dos padrões de design mapeados num jogo de ensino de Engenharia de Software.

3.4 Resumo e Conclusões

Este capítulo descreve o processo de mapeamento dos padrões de design de jogos para os objetivos pedagógicos da área de Gestão de Engenharia de Software. Foi detalhada a abordagem escolhida para esse mapeamento, desde a escolha do artigo de Kelle como base do mapeamento, à matriz resultante do inquérito a professores de Engenharia de Software que permitiu o mapeamento, e finalmente à análise do jogo SimSE como ferramenta de validação do uso desses padrões para um jogo de Engenharia de Software.

Conclui-se que se o jogo SimSE se revelar útil e eficaz no ensino da área de Gestão de Engenharia de Software, o mapeamento apresentado na tabela 13 poderá ser considerado válido, assim como o possível uso de outros padrões mapeados por Kelle no artigo referido.

Capítulo 4

Validação

De forma a validar a utilização dos padrões de design de jogos identificados, no ensino de Gestão de Engenharia de Software (Aresta C na figura X), desenhou-se uma experiência empírica com alunos utilizando o jogo SimSE para teste da uso dos referidos padrões num jogo de ensino de Engenharia de Software, apresentada neste capítulo. Assim, o capítulo divide-se em 5 secções:

- 4.1 Sujeitos – descreve os sujeitos da experiência;
- 4.2 Ambiente da Experiência – descreve o ambiente e sua monitorização;
- 4.3 Protocolo – descreve o protocolo seguido para cada grupo;
- 4.4 Análise de Resultados – apresenta os resultados da experiência;
- 4.5 Resumo e Conclusões – apresenta um resumo do capítulo e as conclusões da experiência.

4.1 Sujeitos

Os sujeitos desta experiência são alunos do Mestrado Integrado em Engenharia Informática e Computação, lecionado na Faculdade de Engenharia da Universidade do Porto, estando todos a frequentar o 2º ano do curso.

Os sujeitos foram divididos em 2 grupos:

- Grupo A – este grupo respondeu ao questionário de avaliação de conhecimentos de Engenharia de Software após jogar o SimSE;
- Grupo B – este grupo estabeleceu o patamar de controlo para a experiência, ao responder ao questionário de avaliação de conhecimentos de Engenharia de Software antes de jogar o SimSE. Após isso, este grupo também jogou o SimSE, tendo volta a responder ao questionário no final.

De forma a minimizar possíveis ameaças à validação dos resultados, as médias de classificação escolar dos dois grupos foram analisadas de forma a garantir a semelhança dos indivíduos de ambos os grupos em termos de conhecimentos e capacidades.

	Média	Desvio Padrão
Grupo B	13,40	1,77
Grupo A	13,12	1,54

Tabela 14 – Médias de classificação escolar dos alunos do Grupo A e B.

2-tail t-test da igualdade de médias (A vs. B)	
Assumindo Igualdade Var.	0,64
Não Assumindo Igualdade Var.	0,65

Tabela 15 – 2-tail t-test entre as médias de classificação escolar dos grupos A e B.

As médias de classificação escolar dos alunos de ambos os grupos são apresentadas no Anexo B, Tabela B.1. De forma a comparar as médias de ambos os grupos (apresentadas na tabela 14), foi realizado um t-test de amostras independentes.

Como se pode verificar na tabela 15, dentro de um intervalo de confiança de 95%, o valor de 0,64 (assumindo igualdade de variâncias) permite concluir que **não existe uma diferença estatisticamente relevante** entre as médias dos dois grupos.

Conhecimento e Experiência Prévia à Experiência

Para o sucesso esta experiência foi importante que todos os sujeitos tivessem conhecimentos básicos do tema de Engenharia de Software e de Desenvolvimento de Software, mas sem que tivessem ainda conhecimentos aprofundados em Gestão de Engenharia de Software, de forma a que conhecimentos prévios não adquiridos em jogo ou sujeitos sem conhecimentos base suficientes não fossem ameaças à validade dos resultados.

Desta forma, foi identificado que os alunos a frequentar o 2º semestre do 2º ano do curso já frequentaram unidades curriculares consideradas relevantes para obtenção dos conhecimentos básicos falados tendo em conta o seu conteúdo e trabalhos nestas realizados, sendo elas: (i) Fundamentos da Programação, (ii) Programação, (iii) Algoritmos e Estruturas de Dados, (iv) Concepção e Análise de Algoritmos e (v) Laboratório de Programação Orientada a Objetos, entre outras com menor importância.

4.2 Ambiente da Experiência

A figura 10 apresenta o todo o protocolo e diferentes fases da experiência empírica conduzida.

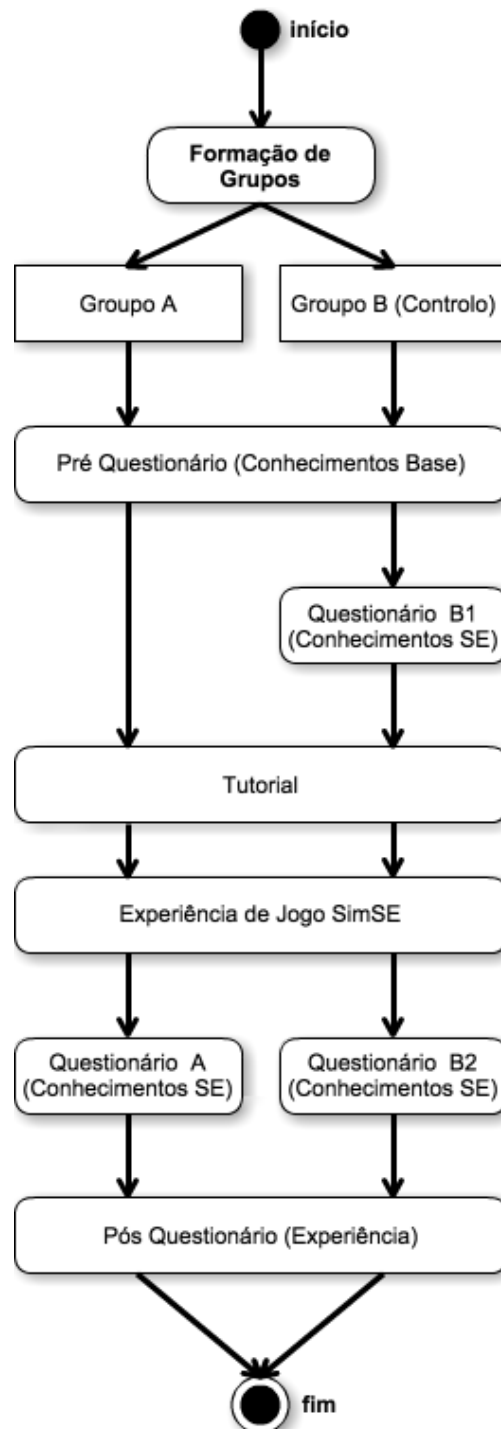


Figura 10: Protocolo e Fases da Experiência

Montagem da Experiência

A experiência foi conduzida numa sala de computadores da Faculdade de Engenharia da Universidade do Porto, ambiente familiar aos alunos visto terem aulas nestas salas, de forma a minimizar os fatores ambientais externos que possam ameaçar a validação dos resultados. A experiência foi realizada durante uma aula da unidade curricular de Laboratório de Programação Orientada a Objetos.

Como [CJMS10] refere, deve-se forçar o trabalho colaborativo e em grupo em estudos empíricos com estudantes, logo foi atribuída uma estação de trabalho a cada par de alunos. Neste emparelhamento de alunos foi tido em conta a experiência anterior de cada aluno a trabalhar com o seu par, tendo sido privilegiados os pares de trabalho já existentes na unidade curricular na qual a experiência foi conduzida (pares com experiência de trabalho conjunto).

A cada par de alunos foi atribuído: uma estação de trabalho (computador), um guia do jogo SimSE, e o link para download do jogo SimSE para o modelo Waterfall.

Monitorização da Experiência

Com foi referido, de forma a monitorizar e controlar a experiência, foi fornecido um guião (Anexo F) a cada par de alunos, de forma a mantê-los focados nos objetivos do jogo, evitando experiências de jogo pouco focadas nos objetivos da experiência em questão.

Foi também apresentado um tutorial, na duração aproximada de 15 minutos, em que foram explicadas todas as ações e objetivos do jogo, visto que nenhum dos alunos está familiarizado com o jogo (situação coberta no pré questionário).

Durante a fase de jogo, o uso dos computadores e telemóveis foi controlado, e qualquer dúvida que surgisse era devidamente esclarecida.

4.3 Protocolo

Nesta experiência, o tratamento ao qual ambos os grupos foram submetidos foi semelhante, diferindo apenas na fase em que cada grupo foi submetido ao questionário de avaliação de conhecimentos de Engenharia de Software pela primeira vez. As fases destes tratamentos são apresentadas de seguida, juntamente com as sua respetivas durações.

Grupo A

1. Pré Questionário (Conhecimentos Base) – 5 minutos;
2. Tempo Livre – 20 minutos;
3. Apresentação do jogo e Tutorial – 15 minutos;
4. Experiência com o jogo – 45 minutos;

5. Questionário de avaliação de Conhecimentos de Engenharia de Software (A) – 20 minutos;
6. Questionário sobre a experiência de jogo – 10 minutos.

Grupo B

1. Pré Questionário (Conhecimentos Base) – 5 minutos;
2. Questionário de avaliação de Conhecimentos de Engenharia de Software (B1) – 20 minutos;
3. Apresentação do jogo e Tutorial – 15 minutos;
4. Experiência com o jogo – 45 minutos;
5. Repetição do Questionário de avaliação de Conhecimentos de Engenharia de Software (B2) – 20 minutos;
6. Questionário sobre a experiência de jogo – 10 minutos.

A duração dos questionários nem sempre foi cumprida, visto que por vezes todos sujeitos terminavam os questionários antes do final do tempo apresentado.

Tarefas da Experiência

A experiência em si resume-se a uma única tarefa, jogar o SimSE. A versão do SimSE a utilizada na experiência é com o modelo de Waterfall.

O desempenho no jogo por parte dos alunos não é considerado relevante, sendo apenas tido em conta a sua análise da capacidade do jogo em desenvolver as funções de aprendizagem relativas aos padrões de design presentes no jogo.

4.4 Análise de Resultados

De forma a determinar provas de validade das suposições apresentadas nesta dissertação, são analisadas as respostas dos questionários aos quais os alunos foram submetidos. Nesta análise são devidamente tidos em conta todos os fatores que possam afetar a validação dos dados propostos, como: (i) Pré Questionário (Conhecimentos Prévios), (ii) Fatores Externos, (iii) Satisfação com a Experiência, e (iv) Aprendizagem com a Experiência. Nos primeiros 3 foi utilizada a escala de Likert [Lik32], que consiste medição de uma resposta (negativa ou positiva) a uma qualquer afirmação, num formato de 5 pontos: (1) Discordo Totalmente, (2) Discordo Parcialmente, (3) Não Concordo nem Discordo, (4) Concordo Parcialmente, e (5) Concordo Totalmente.

Pré Questionário (Conhecimentos Prévios)

A primeira fase da experiência foi a respostas dos alunos de ambos os grupos a um pré questionário (pré questionário no Anexo C, resultados do pré questionário nas Tabelas C.1, C.2), de forma a confirmar que os alunos sujeitos à experiência tinham o nível de conhecimentos base pretendido. As questões e respetivos resultados são apresentados de seguida (tabelas 16 e 17).

		Média	Desvio Padrão
Questões	1	2,56	0,98
	2	2,22	0,81
	3	1,94	0,73
	4	1,00	0,00
	5	1,00	0,00

		Média	Desvio Padrão
Questões	1	2,85	0,55
	2	2,46	0,66
	3	2,46	0,88
	4	1,08	0,28
	5	1,08	0,28

Tabelas 16 e 17 – Resultados do pré questionário dos grupos A e B respetivamente.

1 – Tenho conhecimentos vastos na área de Engenharia de Software: Tanto no Grupo A ($\bar{x} = 2,56$, $\sigma_X = 0,98$) como no Grupo B ($\bar{x} = 2,85$, $\sigma_X = 0,55$) os alunos não demonstram conhecimentos vastos na área de Engenharia de Software, tal como pretendido para esta experiência (valores nos resultados menores ou iguais a 3).

2 – Tenho experiência em gestão de projetos de software: Tanto no Grupo A ($\bar{x} = 2,22$, $\sigma_X = 0,81$) como no Grupo B ($\bar{x} = 2,46$, $\sigma_X = 0,66$) os alunos não demonstram experiência em gestão de projetos de software, tal como pretendido para esta experiência (valores nos resultados menores ou iguais a 3) visto o objetivo do jogo ser educá-los nessa matéria.

3 – Tenho experiência de trabalho em equipas de desenvolvimento de projetos de média/grande dimensão: Tanto no Grupo A ($\bar{x} = 1,94$, $\sigma_X = 0,73$) como no Grupo B ($\bar{x} = 2,46$, $\sigma_X = 0,88$) os alunos não demonstram experiência de trabalho em equipas de desenvolvimento de projetos de média/grande dimensão, tal como pretendido para esta

experiência (valores nos resultados menores ou iguais a 3) visto o objetivo do jogo ser educá-los nessa matéria.

4 – Estou familiarizado com o jogo SimSE: Tanto no Grupo A ($\bar{x}= 1,00$, $\sigma_x= 0,00$) como no Grupo B ($\bar{x}= 1,08$, $\sigma_x= 0,28$) os alunos não se demonstram familiarizados com o jogo SimSE, tal como pretendido para esta experiência (valores nos resultados menores ou iguais a 3), visto que conhecimento anterior do jogo poderia ameaçar a validação dos resultados.

5 – Tenho experiência de jogo do SimSE: Tanto no Grupo A ($\bar{x}= 1,00$, $\sigma_x= 0,00$) como no Grupo B ($\bar{x}= 1,08$, $\sigma_x= 0,28$) os alunos não demonstram experiência no jogo SimSE, tal como pretendido para esta experiência (valores nos resultados menores ou iguais a 3), visto que experiência anterior com o jogo poderia ameaçar a validação dos resultados.

Desta forma, pode-se concluir que o conhecimento dos alunos prévio à experiência era o apropriado para que este não ameace a validação dos resultados.

Fatores Externos

Apesar dos cuidados na concepção da experiência, existem sempre alguns aspetos no ambiente de experiência fora do controlo (barulho, interação com o grupo, gostos pessoais) que podem afetar a validade dos dados adquiridos. Esses aspetos e respetivos resultados são descritos de seguida (tabelas 18 e 19). O questionário referente a estes aspectos encontra-se no Anexo D e os resultados na Tabela D.1 e D.2.

Fatores Externos A			
		Média	Desvio Padrão
Questões	1	4,33	0,59
	2	4,39	0,92
	3	4,28	0,89
	4	4,56	0,86
	5	2,11	1,08
	6	1,17	0,51

Fatores Externos B			
		Média	Desvio Padrão
Questões	1	4,31	0,63
	2	4,38	0,51
	3	4,31	0,75
	4	4,46	0,66
	5	2,46	1,13
	6	1,08	0,28

Tabelas 18 e 19 – Resultados das questões referentes aos fatores externos à experiência dos grupos A e B respetivamente.

1 – Achei o ambiente da experiência apropriado: Tanto no Grupo A ($\bar{x} = 4,33$, $\sigma_X = 0,59$) como no Grupo B ($\bar{x} = 4,31$, $\sigma_X = 0,63$) os alunos consideraram apropriado o ambiente da experiência (resultados superiores a 3). Desta forma, o fator de desconforto com o ambiente da experiência pode ser descartado como ameaça à validação dos resultados.

2 – Gostei da experiência de jogo apresentada: Tanto no Grupo A ($\bar{x} = 4,39$, $\sigma_X = 0,92$) como no Grupo B ($\bar{x} = 4,38$, $\sigma_X = 0,51$) os alunos gostaram da experiência de jogo (resultados superiores a 3). Desta forma, este fator pode ser descartado como ameaça à validação dos resultados.

3 – A qualidade dos gráficos do jogo não interferiu de forma significativamente negativa com toda a experiência de jogo: Tanto no Grupo A ($\bar{x} = 4,28$, $\sigma_X = 0,89$) como no Grupo B ($\bar{x} = 4,31$, $\sigma_X = 0,75$) os alunos consideraram que os gráficos do jogo não interferiram com a experiência (resultados superiores a 3). Desta forma, este fator pode ser descartado como ameaça à validação dos resultados.

4 – Gostei de jogar com o meu colega: Tanto no Grupo A ($\bar{x} = 4,56$, $\sigma_X = 0,86$) como no Grupo B ($\bar{x} = 4,46$, $\sigma_X = 0,66$) os alunos gostaram de jogar com o colega (resultados superiores a 3). Desta forma, o fator resultante do emparelhamento dos alunos pode ser descartado como ameaça à validação dos resultados.

5 – Existiram várias distrações no ambiente no qual se desenrolou a experiência: Tanto no Grupo A ($\bar{x} = 2,11$, $\sigma_X = 1,08$) como no Grupo B ($\bar{x} = 2,46$, $\sigma_X = 1,16$) os alunos consideraram que não existiram muitas distrações no ambiente da experiência (resultados inferiores a 3). No entanto, os resultados apresentados mostram um desvio maior que o esperado e para valores relativamente próximos do 3, sendo uma possível causa o emparelhamento em grupos e o ambiente de aula de laboratório em que os alunos se encontravam. Mesmo assim, os valores encontram-se dentro do esperado, podendo-se considerar que este fator pode ser descartado como ameaça à validação dos resultados.

6 – O jogo ser em inglês afetou a minha experiência de jogo: Tanto no Grupo A ($\bar{x} = 1,17$, $\sigma_X = 0,51$) como no Grupo B ($\bar{x} = 1,08$, $\sigma_X = 0,28$) os alunos consideraram que o jogo ser em inglês não afetou a sua experiência (resultados inferiores a 3). Desta forma, este fator pode ser descartado como ameaça à validação dos resultados.

Outros fatores como a motivação ou a possibilidade da experiência ir contra alguns dos valores dos alunos foram cobertos e descartados pelo facto da experiência ser voluntária.

Satisfação Geral da Experiência

No questionário feito é tido em conta a satisfação dos alunos com toda a preparação da experiência, com o objetivos de cumprir os requisitos estabelecidos em [CJMS10] para o sucesso de um estudo empírico com alunos. As questões e respetivos resultados são apresentados de seguida (tabelas 20 e 21). O questionário referente a estes aspectos encontra-se no Anexo D e os resultados na Tabela D.1 e D.2.

Satisfação Geral A			
		Média	Desvio Padrão
Questões	1	4,78	0,55
	2	3,83	0,86
	3	3,83	0,62

Satisfação Geral B			
		Média	Desvio Padrão
Questões	1	4,54	0,66
	2	3,85	0,80
	3	3,85	0,69

Tabelas 20 e 21 – Resultados das questões referentes satisfação geral da experiência dos grupos A e B respetivamente.

1 – A experiência é apropriada para a curso: Tanto no Grupo A ($\bar{x}= 4,78$, $\sigma_x= 0,55$) como no Grupo B ($\bar{x}= 4,54$, $\sigma_x= 0,66$) os alunos consideraram que a experiência foi apropriada para o curso (resultados superiores a 3). Desta forma, o desajuste da experiência ao curso pode ser descartado como ameaça à validação dos resultados.

2 – O guia atribuído é adequado a uma boa experiência (básica) de jogo: Tanto no Grupo A ($\bar{x}= 3,83$, $\sigma_x= 0,86$) como no Grupo B ($\bar{x}= 3,85$, $\sigma_x= 0,80$) os alunos consideraram que o guia fornecido foi adequado a uma boa experiência de jogo (resultados superiores a 3). No entanto, eram esperados resultados ligeiramente mais altos (4 ou superior) o que pode indicar que este guia, apesar de apropriado, podia ser melhorado. Mesmo assim, os resultados permitem descartar os descontentamento e/ou desajuste do guião como ameaça à validação dos resultados.

3 – A apresentação e tutorial apresentados são adequados a uma boa experiência (básica) de jogo: Tanto no Grupo A ($\bar{x}= 3,83$, $\sigma_x= 0,62$) como no Grupo B ($\bar{x}= 3,85$, $\sigma_x= 0,69$) os alunos consideraram que a apresentação e tutorial apresentados foram adequados a uma boa experiência de jogo (resultados superiores a 3). No entanto, eram esperados resultados ligeiramente mais altos (4 ou superior) o que pode indicar que estes pontos, apesar de satisfatórios, podiam ser melhorados. Mesmo assim, os resultados

permitem descartar os descontentamento e/ou desajuste da apresentação e tutorial como ameaças à validação dos resultados.

Aprendizagem com a Experiência

Nesta fase é necessário avaliar se o jogo SimSE, usando os padrões de design referidos no capítulo anterior, é capaz de desenvolver competências relativas à área de Engenharia de Software e, mais precisamente, à sub área de Gestão de Engenharia de Software. Para isso foi elaborado um questionário de 25 perguntas sobre essas duas áreas [Bho15], ao qual os alunos de ambos os grupos foram submetidos, nas condições referidas acima neste documento.

O questionário e a tabela com os resultados da experiência encontram-se no Anexo E e nas Tabelas E.1, E.2 e E.3.

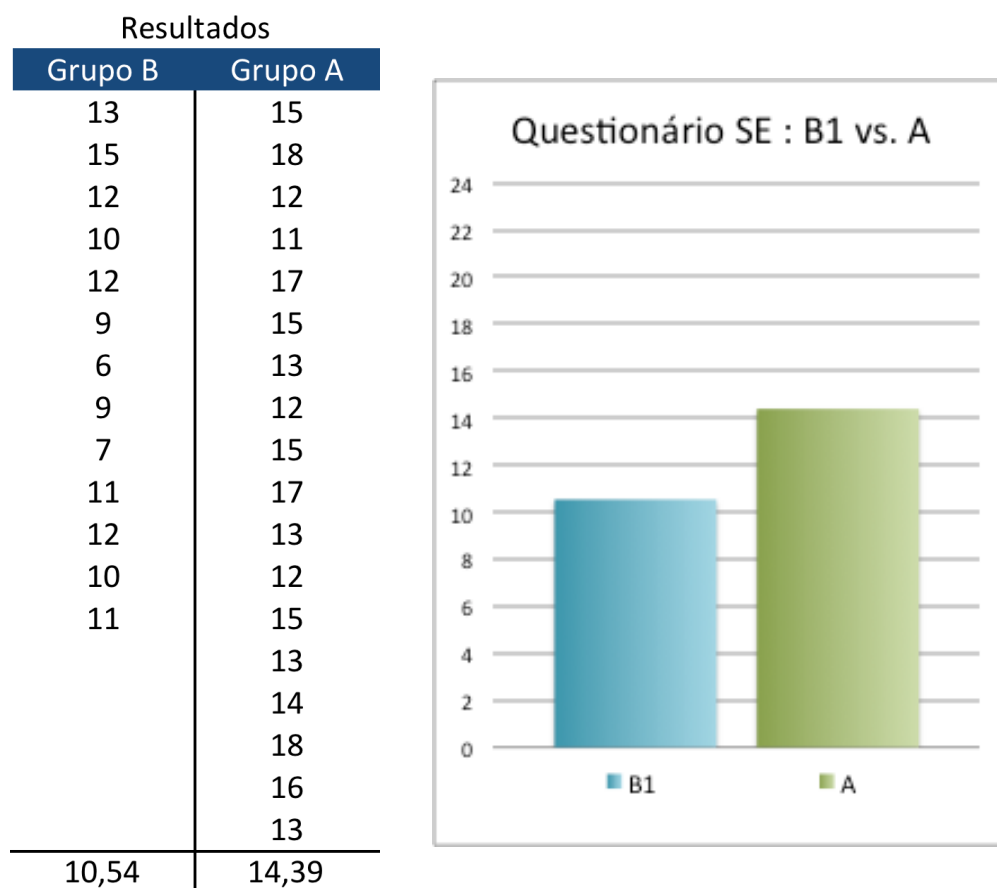


Figura 11 – Resultados do questionário do Grupo B (na primeira tentativa, antes de jogar o SimSE) vs. Grupo A (jogou o SimSE antes de responder)

Como se pode verificar na figura 11, os alunos do grupo A, que jogaram o SimSE durante 45 minutos antes de responder ao questionário, obtiveram melhores resultados do que os alunos do grupo B, que não jogaram o SimSE antes de responder.

Esta melhoria traduz-se num **aumento de 36,54%** nos resultados dos questionários, podendo-se ainda analisar que do grupo A apenas 3 alunos em 18 (16,67%) obtiveram resultados negativos (menor que 13/25 respostas certas), ao contrário do grupo B em que se registou 11 em 13 (84,62%) resultados negativos.

Foram também analisados os resultados do grupo B, mas depois de já ter jogado o SimSE.

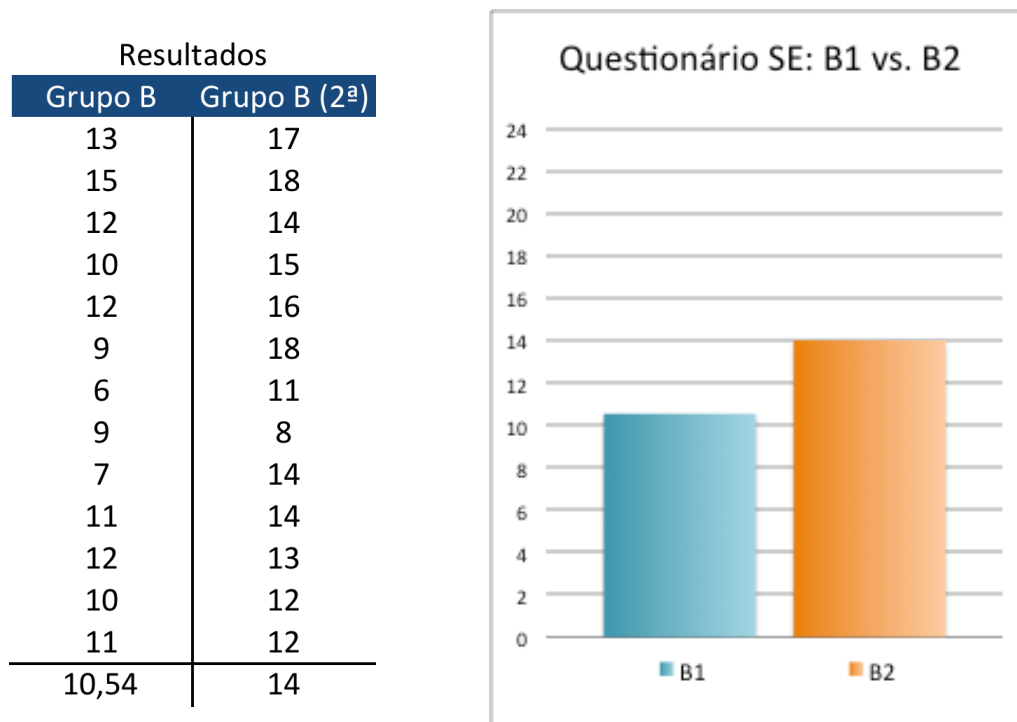


Figura 12 - Resultados do questionário do Grupo B antes e depois de jogar o SimSE.

Como se pode verificar na figura 12, os alunos do grupo B melhoraram os seus próprios resultados após jogarem o SimSE. Esta melhoria traduz-se num aumento de 32,85% nos resultados dos questionários, podendo-se ainda analisar que os resultados negativo desceram de 11 em 13 (84,62%) para 4 em 13 (30,77%).

4.5 Resumo e Conclusões

Este capítulo descreve a realização de uma experiência empírica com alunos na qual se pretendia validar o uso de um jogo contendo os padrões de design de jogos referidos no capítulo anterior, no ensino da área de Gestão de Engenharia de Software. Na experiência participaram 31 alunos do 2º ano do Mestrado Integrado em Engenharia Informática e Computação, lecionado na Faculdade de Engenharia da Universidade do Porto, divididos em dois grupos. Como ferramenta para esta validação foi utilizado jogo SimSE.

Combinando os resultados da experiência e descartadas as possíveis ameaças à validação desses resultados, conclui-se que o uso dos padrões de design de jogos selecionados e presentes no SimSE podem ser validados para o ensino de Engenharia de Software, especificamente da sub área de Gestão de Engenharia de Software. A figura 13 representa os resultados dos questionários realizados, onde é possível analisar a melhoria de resultados nos questionários dos grupos depois de jogar o SimSE (grupos A e B2), face ao grupo que não jogou o SimSE (grupo B1).

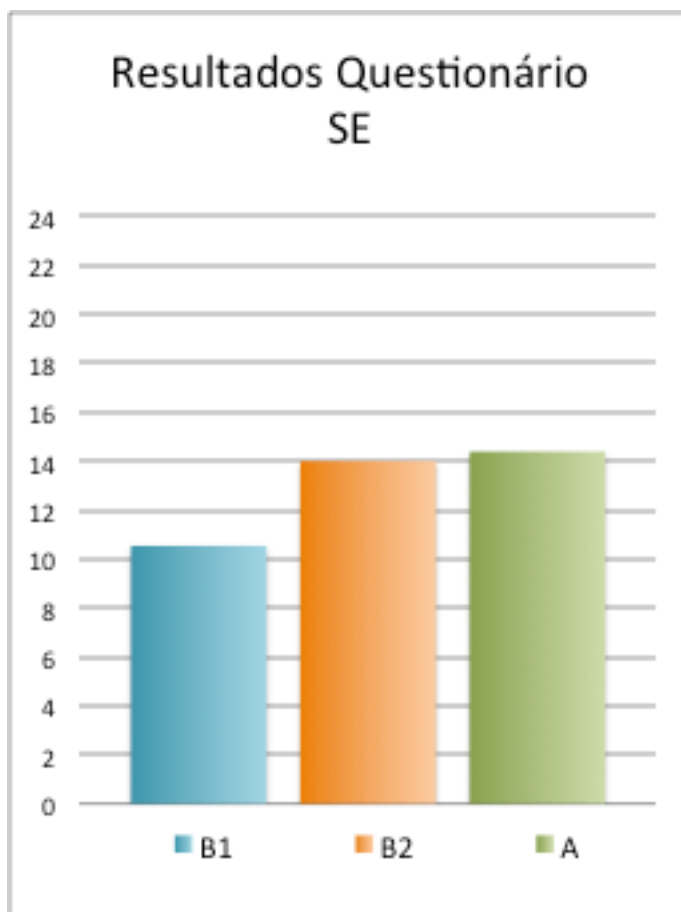


Figura 13 – Resultados dos questionários de todos os grupos.

Capítulo 5

Conclusões e Trabalho Futuro

Engenharia de Software é uma área de domínio com um vasto alcance e alta densidade de conceitos e conhecimentos. Gestão de Engenharia de Software é difícil de ensinar aos alunos, devido à sua diversidade de temas e conceitos, e à especificidade das técnicas que poderão ser adequadas.

Os jogos sérios podem ajudar no processo ensino de Gestão de Engenharia de Software, mas o design e desenvolvimento desses jogos tende a ser demasiado complexo para que o seu uso seja adoptado em larga escala. Um mapa de padrões de design de jogos para a área de Gestão de Engenharia de Software pode diminuir esse esforço e pode favorecer o desenvolvimento e utilização de tais jogos, e conseqüentemente, melhorar o processo de ensino de uma das áreas de conhecimento de Engenharia de Software, podendo futuramente estes benefícios ser aplicados às restantes áreas.

No capítulo 1 é feita a introdução do trabalho, apresenta-se o enquadramento do trabalho, o problema e motivação deste trabalho de investigação, e descreve-se os objetivos e resultados esperados do trabalho realizado.

No capítulo 2, é feito o levantamento do estado da arte relativo ao ensino de Engenharia de Software e das suas áreas de conhecimento, ao uso de jogos sérios no ensino, aos padrões pedagógicos existentes, e aos padrões de design de jogos educativos. Como conclusão do capítulo é apresentado um mapeamento já validado de funções pedagógicas para padrões de design de jogos, apropriado para validação para a área de Gestão de Engenharia de Software. Como possível ferramenta para a validação é identificado o jogo sério SimSE.

No capítulo 3, é descrito o processo de mapeamento dos padrões de design de jogos para os objetivos pedagógicos da área de Gestão de Engenharia de Software, detalhada a abordagem para a realização dessa tarefa, a escolha do artigo de Kelle como base do mapeamento, a matriz de inquérito a professores de Engenharia de Software, o mapeamento resultante da conjugação da matriz com o artigo, e a análise ao jogo sério SimSE para a tarefa de validação. Conclui-se no capítulo que o mapeamento apresentado é válido se a utilização

do SimSE para ensino da área de Gestão de Engenharia de Software se provar eficaz e adequado.

No capítulo 4 é descrita a validação ao mapeamento, feita através de uma experiência empírica com alunos. Na experiência participaram 31 alunos, divididos em dois grupos, utilizando como ferramenta o jogo SimSE. Na comparação dos resultados obtidos pelos grupos num questionário elaborado para teste de conhecimentos, concluiu-se que o uso dos padrões de design de jogos selecionados e presentes no SimSE podem ser validados para o ensino da área de Gestão de Engenharia de Software, resultado da melhoria de resultados nos questionários dos grupos depois de jogar o SimSE.

O presente capítulo 5, apresenta as conclusões deste trabalho de investigação, a satisfação dos objetivos (na secção 5.1), e o trabalho futuro (na secção 5.2).

5.1 Satisfação dos Objetivos

Ao conseguir efetuar o mapeamento dos objetivos pedagógicos de Gestão de Engenharia de Software para padrões de design de jogos apenas utilizando as funções de aprendizagem com concordância total por parte dos especialistas consultados, foi possível obter um mapeamento, no qual é possível diretamente associar a cada tópico da área de Gestão de Engenharia de Software a classe de padrões de design de jogos adequada para a sua implementação.

Através da análise da experiência, na qual se pretendia avaliar a eficácia de um jogo de ensino de Engenharia de software contendo esses padrões, foi possível verificar a melhoria significativa dos resultados dos alunos ao jogar tal jogo.

Assim, pode-se concluir que o mapeamento realizado tem todas as condições para servir de base no design de jogos sérios para Gestão de Engenharia de Software, tendo sido os objetivos deste trabalho de investigação alcançados.

Foi ainda feito um artigo submetido na conferencia CSE 2015.

5.2 Trabalho Futuro

Neste momento, é possível continuar o trabalho realizado explorando dois possíveis tópicos em que o trabalho poderia ser melhorado:

- Continuar os inquéritos a especialistas no ensino de Engenharia de Software, não só para melhorar os resultados obtidos de forma a obter uma ideia mais clara do peso de cada função de aprendizagem para cada tópico de Gestão de

Engenharia de Software, mas também para abranger este estudo às outras áreas de conhecimento de Engenharia de Software;

- Validação dos novos resultados obtidos, utilizando jogos de ensino de Engenharia de Software existentes, melhorando alguns desses jogos existentes, ou até criando um jogo utilizando os padrões de design de jogos.

Desta forma será possível continuar o trabalho realizado e aplicá-lo às restantes áreas de Engenharia de Software como pretendido.

Referências

- [Sch08] Schell, J. (2008) *The Art of Game Design: A book of Lenses*, 1st ed., Morgan Kaufmann, Waltham, MA.
- [SM94] Shuell, T.J. e Moran, K.A. (1994) 'Learning theories: historical overview and trends', in T. Husen e T.N. Postlethwaite (Eds.): *The International Encyclopedia of Education*, 2nd ed., pp.3340–3345, Pergamon, Oxford, UK.
- [SJB07] Susi, T., Johannesson, M. e Backlund, P. (2007) *Serious Games – An overview*, University of Skövde (Technical Report HS-IKI-TR-07-001), Skövde, Sweden.
- [PPP15] The Pedagogical Patterns Project (2015) Disponível em <http://www.pedagogicalpatterns.org/> (acedido a 14 de Janeiro de 2015).
- [RCV09] Ritterfeld, U., Cody, M. e Vorderer, P. (2009) *Serious Games: Mechanisms and Effects*, 1st ed., Routledge, New York, NY.
- [Kol84] Kolb, D.A. (1984) *Experiential Learning: Experience as the Source of Learning and Development*, Prentice-Hall Englewood Cliffs, NJ.
- [Kel83] Keller, J.M. (1983) 'Motivational design of instruction', em C.M. Reigeluth (Ed.): *Instructional Design Theories and Models: An Overview of their Current Status*, Vol. 1, pp.386–434, Routledge, New York, NY.
- [Hei01] Heinich, R., Michael, M., Smaldino, S.E. e Russell, J.D. (2001) *Instructional Media & Technologies for Learning*, 7th ed., Prentice-Hall, Upper Saddle River, NJ.
- [Gro07] Grösser, M. (2007) 'Effective teaching: linking teaching to learning functions', *South African Journal of Education*, Vol. 27, No. 1, pp.37–52.
- [GAD02] Garris, R., Ahlers, R. e Driskell, J.E. (2002) 'Games, motivation, and learning: a research and practice model', *Simulation & Gaming*, Vol. 33, No. 4, p.441.
- [Gam95] Gamma, E. (1995) *Design Patterns: Elements of Reusable Object-Oriented Software*, Addison-Wesley Professional, Boston, MA.
- [Gag65] Gagné, R.M. (1965) *The Conditions of Learning*, Holt, Rinehart and Winston, Austin, TX.
- [BH04] Björk, S. e Holopainen, J. (2004) *Patterns in Game Design*, Charles River Media, Boston, MA.

- [Ale78] Alexander, C. (1978) *A Pattern Language: Towns, Buildings, Construction*, Oxford University Press, Oxford, UK.
- [AC98] Agerbo, E. e Cornils, A. (1998) 'How to preserve the benefits of design patterns', pp.134–143, ACM. doi:10.1145/286936.286952, Vancouver, British Columbia, Canada.
- [KKS11] Kelle, S., Klemke, R. e Specht, M. (2011) 'Design patterns for learning games', *Int. J. Technology Enhanced Learning*, Vol. 3, No. 6, pp.555–569.
- [DLM11] Maciuszek, D. Ladhoff, S. e Martens, A. (2011) 'Content Design Patterns for Game-Based Learning'. *IJGBL* 1(3):65-82.
- [MM10] Maciuszek, D., e Martens, A. (2010). Patterns for the design of educational games. In F. Edvarsen, & H. Kulle (Eds.), *Educational Games: Design, Learning, and Applications*. Hauppauge, NY: Nova.
- [HM07] Harrer, A., e Martens, A. (2007). A step towards a pattern language for e-learning systems. In C. Pahl (Ed.), *Architecture Solutions for E-Learning Systems*. Hershey, PA: IGI Global.
- [Cra05] Crawford, C. (2005). *Chris Crawford on Interactive Storytelling*. Berkeley, CA: New Riders Games.
- [HH01] Hallford, N., e Hallford, J. (2001). *Swords & Circuitry: A Designer's Guide to Computer Role-Playing Games*. Roseville, CA: Prima Tech.
- [MM09] Maciuszek, D., e Martens, A. (2009). Story patterns for the design of game-based learning experiences. In M. Pivec (Ed.), *Proceedings of the 3rd European Conference on Games-Based Learning (ECGBL 2009)* (pp. 248-255). Reading: Academic Publishing.
- [SC95] Schank, R. C., e Cleary, C. (1995). *Engines for Education*. Hillsdale, NJ: Lawrence Erlbaum.
- [HJ04] Holmer, T., e Jödick, F. (2004). Kooperation in kleineren Lerngruppen. In J. Haake, G. Schwabe, & M. Wessner (Eds.), *CSCL-Kompodium* (pp. 66-79). München: Oldenbourg. Em Alemão.
- [Pre01] M. Prensky, "Fun, Play and Games: What Makes Games Engaging," In: *Digital Game-Based Learning*, 2001.
- [Nav03] Navarro, J. I., Marchena, E., Alcalde, C., Ruiz, G., Llorens, I. e Aguilar, M. (2003) Improving attention behaviour in primary and secondary school children with a computer assisted instruction procedure. *International Journal of Psychology* 38(6), 359–365.

- [Nav06] Navarro, E. (2006). SimSE: A Software Engineering Simulation Environment, University of California, Irvine
- [Swe14] Swebok. (2014). *Guide to software engineering body of knowledge*. IEEE Computer Society. Retirado a 16 de Janeiro 2015, from <http://www.swebok.org/>
- [Rad00] Radford, A. (2000) Games and learning about form in architecture. *Automation in Construction*, 9(4), 379–385.
- [MS04] Mitchell, A. e Savill-Smith, C. (2004) *The use of computer and video games for learning: A review of the literature*. Learning and Skills Development Agency.
- [Eno04] Enochsson, L., Isaksson, B., Tour, R., Kjellin, A., Hedman, L., Wredmark, T. e Tsai- Fellander, L. (2004) Visuospatial skills and computer game experience influence the performance of virtual endoscopy. *Journal of Gastrointestinal Surgery*, 8(7), 874–880.
- [Cor06] Corti, K. (2006) *Games-based Learning; a serious business application*. PIXELearning Limited.
- [Lik32] Likert, R. (1932) *A technique for the measurement of attitudes*. *Archives of Psychology* 22, 1-55.
- [CJMS10] Carver, J., Jaccheri, L., Morasca, S., Shull, F. (2010) *A checklist for integrating student empirical studies with research and teaching goals*. *Empirical Software Engineering*.
- [Bho15] Bhojasia, M. (2015) *Software Engineering Questions and Answers*. Disponível em <http://www.sanfoundry.com/software-engg-mcqs-project-management/> (acedido a 14 de Maio de 2015).

Anexo A

Matriz de Inquérito a Professores de Engenharia de Software

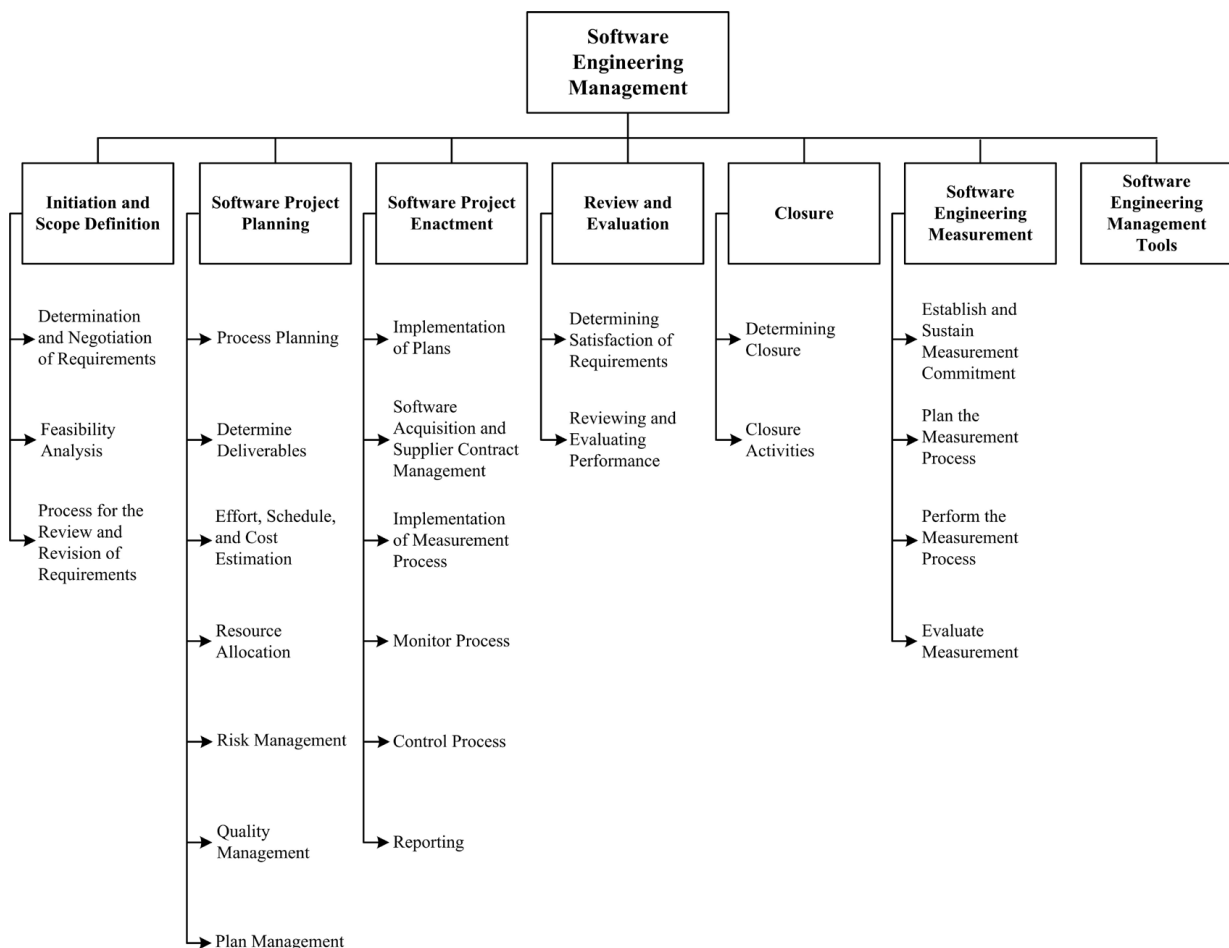
Nas próximas duas páginas encontra-se anexado a matriz de inquérito enviada a professores de Engenharia de Software, com o objetivo de relacionar e validar as funções de aprendizagem para os tópicos da área de Gestão de Engenharia de Software.

Nela constam um pequeno resumo do que se pretendia atingir com o inquérito e a matriz a preencher.

Questionnaire on Software Engineering Management Learning Functions

In the ambit of my dissertation on Game Design Techniques for Software Engineering, that aims to identify game design patterns that best fit to Software Engineering body of knowledge, I would be grateful if you could help completing this questionnaire on Software Engineering Management Learning Functions.

The objective of this questionnaire is to identify which Shuell and Moran’s learning functions are more relevant to each Swebok’s knowledge area of Software Engineering Management, illustrated in the image below. Complete the table in the next page by selecting (X mark) the most relevant learning functions to each knowledge area of Software Engineering Management (columns) according to you. You can select more than 1 learning function per knowledge area.



		Software Engineering Management Knowledge Areas							
Learning Functions	Teaching Functions	Initiation and Scope Definition	Software Project Planning	Software Project Enactment	Review and Evaluation	Closure	Software Engineering Measurement	Software Engineering Management Tools	
Preparation	Prior knowledge activation	Reminding students of prerequisite information or asking oneself what is already known about the topic being learned							
	Motivation	Learner persistence and contribution need to be nurtured							
	Expectations	Learners need to have a general idea of what is to be accomplished from the learning task.							
	Attention	Enabling learners to focus on relevant information, disregarding the irrelevant information							
Knowledge Manipulation	Encoding	Assisting learners to add personal meaning to new information							
	Comparison	Making comparisons in searching for similarities and differences that permit the formation of higher-order relationships characteristic of understanding							
	Repetition	The inducement of multiple perspectives and engaging in systematic reviews are two ways in which this function can be initiated							
	Interpreting	Assisting learners in converting information from one form of representation to another							
	Exemplifying	Motivating learners to illustrate by making use of new examples							
Higher Order Relationships	Combination, integration, synthesis	Isolated pieces of information must be combined in ways that permit integration and synthesis. Developing organizational schemes such as tables and diagrams are examples of how this function can be initiated							
	Classifying	Enabling learners to determine categories of concepts							
	Summarising	Guiding learners in writing short statements that represent information							
	Analysing	Guiding learners to break material into constituent parts and to determine how the parts are related							
Learner Regulation	Feedback	Learners need to interpret feedback on the adequacy and accuracy of their understanding							
	Evaluation	Providing learners with the opportunity to interpret and evaluate the feedback, as well as the opportunity to evaluate their own work against set criteria and standards							
	Monitoring	Providing learners with the opportunity to monitor their own learning progress, to determine if reason able progress is being made							
	Planning	Assisting learners in devising methods for accomplishing tasks							
Productive Actions	Hypothesis generation	Encouraging learners to try alternate courses of action or generating alternative solutions							
	Inferring	Assisting learners to draw conclusions from presented information							
	Explaining	Guiding learners in constructing mentally and using cause and-effect models							
	Applying	Teaching learners how to utilise procedures to perform exercises or solve problems							
	Producing and constructing	Guiding learners to invent a product							

Instructions:

Select the **relevant (in your opinion)** Learning Functions for each Knowledge Area of Software Engineering Management
 The Teaching Functions are present to provide a better understanding of each Learning Function
 You can select **more than 1** learning function per Knowledge Area (column)

Observations:

Name:

Age:

Gender: (a) Male (b) Female

E-mail:

Job/Occupation:

Institution:

Anexo B

Pré questionário da Experiência

Nas próxima página encontram-se anexadas as médias de classificação escolar dos alunos dos grupos A e B.

	Grupo A	Grupo B
	14,62	10,98
	11,52	12,74
	11,66	15,41
	15,49	14,77
	11,55	14,71
	11,42	16,42
	14,53	12,88
	15,25	13,32
	13,74	11,53
	14,41	11,28
	12,26	12,65
	13,18	12,09
	12	15,39
	15,91	
	11,88	
	12,01	
	12,26	
	12,44	
Média	13,12	13,40
Desv. Padrão	1,54	1,77

Tabela B.1 – Médias de classificação escolar dos alunos dos grupos A e B

Anexo C

Pré questionário da Experiência

Nas próximas três páginas encontra-se anexado o pré questionário respondido pelos alunos na experiência, assim como os resultados desse questionário.

Valores fora do intervalo pretendido estão representados a vermelho nas tabelas de resultados.

Pré-Questionário

Obrigado por participar nesta experiência. Pedíamos agora que respondesse a este breve questionário sobre os seus conhecimentos prévios sobre os temas a abordar. Deve colocar um “X” na opção com que melhor se identifica, sendo de “1” (não se identifica nada com a afirmação) a “5” (identifica-se totalmente com a afirmação).

	1	2	3	4	5
1. Tenho conhecimentos vastos na área de Engenharia de Software.					
2. Tenho experiência em gestão de projetos de software					
3. Tenho experiência de trabalho em equipas de desenvolvimento de projetos de média/grande dimensão.					
4. Estou familiarizado com o jogo SimSE.					
5. Tenho experiência de jogo do SimSE.					

Obrigado pelo seu tempo

		Perguntas				
		1	2	3	4	5
Questionários	Q1	4	2	1	1	1
	Q2	2	2	1	1	1
	Q3	3	3	2	1	1
	Q4	3	3	2	1	1
	Q5	3	3	3	1	1
	Q6	3	3	2	1	1
	Q7	2	2	2	1	1
	Q8	2	2	2	1	1
	Q9	2	2	2	1	1
	Q10	3	3	2	1	1
	Q11	2	2	1	1	1
	Q12	4	3	3	1	1
	Q13	3	3	3	1	1
	Q14	3	1	2	1	1
	Q15	1	1	2	1	1
	Q16	4	3	3	1	1
	Q17	1	1	1	1	1
	Q18	1	1	1	1	1

Tabela C.1 – Resultados do pré questionário do Grupo A.

		Perguntas				
		1	2	3	4	5
Questionários	Q1	2	2	2	1	1
	Q2	3	1	1	1	1
	Q3	3	2	2	1	1
	Q4	3	3	2	1	1
	Q5	3	2	3	1	1
	Q6	4	3	3	1	1
	Q7	2	2	2	2	2
	Q8	3	3	3	1	1
	Q9	3	3	2	1	1
	Q10	2	3	2	1	1
	Q11	3	3	4	1	1
	Q12	3	2	2	1	1
	Q13	3	3	4	1	1

Tabela C.2 – Resultados do pré questionário do Grupo B.

Anexo D

Questionário de Fatores Externos /Satisfação Geral da Experiência

Nas próximas quatro páginas encontra-se anexado o questionário sobre os fatores externos (pontos 1.) e satisfação geral (pontos 2.) respondido pelos alunos na experiência, assim como os resultados desse questionário.

Valores fora do intervalo pretendido estão representados a vermelho nas tabelas de resultados.

Questionário sobre Experiência

Obrigado por participar nesta experiência. Pedíamos agora que respondesse a este breve questionário sobre o ambiente desta experiência de jogo.

As questões apresentadas estão relacionadas com a experiência que acabou de vivenciar. Estas estão divididas em secções. Deve colocar um “X” na opção que melhor identifica a sua opinião das afirmações apresentadas, sendo:

- 1 - Discordo Totalmente**
- 2 - Discordo Parcialmente**
- 3 - Não Concordo nem Discordo**
- 4 - Concordo Parcialmente**
- 5 - Concordo Totalmente**

Fatores Externos

	1	2	3	4	5
1. Achei o ambiente da experiência apropriado					
2. Gostei da experiência de jogo apresentada					
3. A qualidade dos gráficos do jogo não interferiu de forma significativamente negativa com toda a experiência de jogo					
4. Gostei de jogar com o meu colega					
5. Existiram várias distrações no ambiente no qual se desenrolou a experiência					
6. O jogo ser em inglês afetou a minha experiência					

Satisfação com a preparação da Experiência

	1	2	3	4	5
1. A experiência é apropriada para o curso					
2. O guia fornecido é adequado a uma boa experiência (básica) de jogo					
3. A apresentação e tutorial apresentados são adequados a uma boa experiência (básica) de jogo					

Comentários

Obrigado pelo seu tempo

		Perguntas								
		1.1	1.2	1.3	1.4	1.5	1.6	2.1	2.2	2.3
Questionários	Q1	5	4	5	5	1	2	5	3	3
	Q2	5	5	4	5	3	1	5	3	4
	Q3	4	5	5	4	2	1	5	5	4
	Q4	4	4	3	4	2	1	5	4	4
	Q5	4	5	4	5	2	3	5	5	5
	Q6	4	5	4	5	1	1	5	5	4
	Q7	4	5	4	2	2	1	5	4	4
	Q8	4	5	5	5	3	1	4	3	4
	Q9	5	2	5	5	2	1	5	4	4
	Q10	3	3	4	3	2	1	4	2	3
	Q11	5	5	4	5	4	1	5	4	4
	Q12	4	5	2	5	1	1	5	4	5
	Q13	5	5	5	5	1	1	5	3	3
	Q14	4	4	5	5	1	1	5	4	3
	Q15	4	4	5	5	4	1	3	3	3
	Q16	5	5	5	5	2	1	5	4	4
	Q17	5	5	5	5	4	1	5	5	4
	Q18	4	3	3	4	1	1	5	4	4

Tabela D.1 – Resultados do questionário sobre fatores externos e satisfação geral do Grupo A.

		Perguntas								
		1.1	1.2	1.3	1.4	1.5	1.6	2.1	2.2	2.3
Questionários	Q1	3	4	4	4	2	2	3	3	4
	Q2	4	4	5	4	3	1	5	4	4
	Q3	4	4	3	4	1	1	4	4	3
	Q4	4	4	4	4	2	1	4	3	3
	Q5	4	5	5	3	2	1	5	4	4
	Q6	5	4	3	5	3	1	5	4	3
	Q7	5	5	5	5	2	1	5	4	4
	Q8	4	4	4	4	1	1	4	4	4
	Q9	5	5	5	5	5	1	5	5	5
	Q10	4	5	4	5	2	1	5	2	3
	Q11	4	4	4	5	2	1	4	4	4
	Q12	5	4	5	5	4	1	5	5	5
	Q13	5	5	5	5	3	1	5	4	4

Tabela D.2 – Resultados do questionário sobre fatores externos e satisfação geral do Grupo B.

Anexo E

Questionário de Avaliação de Conhecimentos de Engenharia de Software

Nas próximas onze páginas encontra-se anexo de avaliação de conhecimentos de Engenharia de Software, composto por 25 perguntas, e os resultados da sua resposta por parte dos alunos dos grupos A e B.

Questionário sobre Conhecimentos de Engenharia de Software

Obrigado por participar nesta experiência. Pedíamos agora que respondesse a este breve questionário.

As questões apresentadas estão desenhadas para testar alguns conhecimentos básicos de Engenharia de Software, com foco especial na área de *Software Engineering Management*. Pedíamos que respondesse às questões de acordo com os seus conhecimentos. Deve colocar um CÍRCULO na opção que considera correta. No caso de se enganar basta riscar e marcar (ou não) outra resposta.

1. Gestão eficaz de projetos de software foca-se em:

- a. Pessoas, desempenho, ganhos (lucro), produto
- b. Pessoas, produto, desempenho, processo
- c. Pessoas, produto, processo, projeto
- d. Pessoas, processo, ganhos (lucro), produto

2. O primeiro passo no planeamento de um projeto é:

- a. Determinar o orçamento
- b. Selecionar um modelo organizacional de equipa
- c. Determinar as restrições do projeto
- d. Estabelecer o âmbito e objetivos

3. Quais dos seguintes NÃO é considerado um interveniente no processo de software?

- a. Clientes
- b. Utilizadores finais
- c. Gestores de projeto
- d. Vendedores do produto

4. Qual o fator MENOS importante na altura de escolher a estrutura organizacional para uma equipa de software?

- a. Grau de comunicação desejado
- b. Tamanho previsto do programa resultante
- c. Rigidez da data de entrega
- d. Tamanho do orçamento do projeto

5. Uma das melhores maneiras de evitar frustração e quebra no rendimento durante o desenvolvimento de software é:

- a. Dar aos elementos de equipa maior autonomia sobre o projeto e compensá-los monetariamente.
- b. Dar aos elementos de equipa menor autonomia sobre o projeto e compensá-los monetariamente.
- c. Esconder as más notícias dos membros da equipa até as coisas melhorarem.
- d. Diminuir o trabalho dos elementos da equipa.

6. Qual destas características NÃO É um fator que contribui para dificuldades na coordenação do projeto?

- a. Interoperabilidade do Software
- b. Desempenho do Software
- c. Dimensão do Projeto de Software
- d. Incerteza no plano a seguir

7. Qual destas atividades é MENOS importante na gestão de um projeto de software?

- a. Avaliação da usabilidade do produto.
- b. Análise do produto tendo em conta os padrões de qualidade definidos.
- c. Estimação de custos do projeto.
- d. Gestão de riscos do projeto.

8. No modelo *Waterfall*, ao longo do projeto, ambos produto e processo vão-se adaptando às necessidades que vão surgindo.

- a. Verdadeiro
- b. Falso

9. Quando é que certas atividades do processo podem ser omitidas durante a adaptação do projeto a mudanças?

- a. Quando é um projeto de pequenas dimensões.
- b. Sempre que o desenvolvimento do software é urgente.
- c. Prototipagem rápida não requer o uso dessas atividades.
- d. Nunca.

10. Que atividade o gestor de projetos de software necessita fazer para minimizar o risco do projeto falhar?

- a. Duplicar o tamanho da equipa
- b. Pedir um orçamento maior
- c. Não permitir absolutamente nenhuma derrapagem na calendarização
- d. Definir *milestones* e monitorizar progresso

11. Quais dos seguintes NÃO é considerado um risco na gestão de projetos?

- a) Atrasos de Especificação
- b) Produtos concorrentes
- c) Testes
- d) Substituição de pessoal

12. Qual/Quais destes é/são parâmetros essenciais a ter em conta quando se calcula os custos de um projeto de software?

- a) Ambiente de Desenvolvimento
- b) Custos de Hardware
- c) Recursos Humanos
- d) Todos os mencionados

13. O Planeamento da Qualidade é um processo de desenvolver um plano de qualidade que garanta a máxima qualidade relativamente a:

- a) a equipa
- b) o projeto
- c) os clientes
- d) o gestor de projetos

14. Identifique a atividade que procura o melhoramento de processos

- a) Introdução de Processos Novos
- b) Análise dos Processos em Vigor
- c) Des-Processificação
- d) Seccionamento de Processos

15. O único produto resultante do trabalho a entregar é o código fonte.

- a) Verdadeiro
- b) Falso

16. Diga qual das seguintes atividades é uma atividade fundamental nos processos de software?

- a) Validação e Verificação de Software
- b) Design e Implementação de Software
- c) Especificação do Software
- d) Todas as mencionadas

17. Quanto mais tempo uma falha existir no software...

- a) ...mais morosa é a sua remoção.
- b) ...mais custosa é para detetar e corrigir.
- c) ...menos provável é que seja corrigida convenientemente.
- d) Todas as mencionadas.

18. Rearranje os seguintes passos de forma ilustrar a ordem de um processo de engenharia de software comum.

i. Testes

ii. Design

iii. Integração

iv. Especificação (Requisitos)

v. Construção do Produto

- a) 2, 4, 5, 3, 1
- b) 4, 2, 5, 3, 1
- c) 2, 4, 5, 1, 3
- d) 4, 2, 5, 1, 3

19. No modelo *Waterfall*, a gestão de requisitos tem um custo inicial elevado mas não necessita (à partida) de mais fundos durante o projeto.

- a) Verdadeiro
- b) Falso

20. O feedback fornecido pelos elementos da equipa é importante para a gestão do projeto?

- a) Não.
- b) Sim, mas apenas as informações relativas ao projeto.
- c) Sim, todas as informações são relevantes e devem ser geridas.
- d) Sim, todas as informações, mas só devem ser geridas informações relativas diretamente ao projeto.

21. Qual das características seguintes é mais importante num gestor de projetos eficaz?

- a) Capacidade de resolução de problemas
- b) Capacidade de gestão
- c) Influência e capacidade de manter coesão da equipa
- d) Todas as anteriores têm igual importância

22. As ferramentas de desenvolvimento do projeto são importantes para a gestão do projeto de software?

- a) Sim, e em todas as fases do projeto.
- b) Sim, mas apenas na fase de desenvolvimento.
- c) Sim, mas apenas na fase de planeamento.
- d) Não.

23. Qual das seguintes é uma razão para o software ser entregue para lá do prazo de entrega?

- a) Alteração dos requisitos o cliente não refletidas no plano temporal do projeto.
- b) Dificuldades tecnológicas não previstas .
- c) Dificuldades humanas não previstas.
- d) Todas as mencionadas

24. Uma das razões pelas quais o software é difícil de concretizar é:

- a) Controlo de versões
- b) Falta de reusabilidade do software
- c) Falta de monitorização do processo
- d) Todas as mencionadas

25. Qual/Quais dos seguintes elementos é/são parte do trabalho resultante de um processo de software?

- a) Programas de computador
- b) Documento de análise de requisitos
- c) Documento de design de software
- d) Todos os mencionados

Comentários

Obrigado pelo seu tempo

		Perguntas																									Resultado	
		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25		
Questionários	Q1	0	1	0	0	1	0	0	0	1	1	0	1	1	0	1	1	0	1	1	1	1	1	1	1	0	15	
	Q2	0	0	1	0	1	1	0	1	1	1	1	1	1	1	1	1	1	0	0	1	1	0	1	0	1	17	
	Q3	0	0	0	0	0	1	1	0	0	1	0	0	1	1	1	1	0	1	0	1	1	1	1	0	1	13	
	Q4	0	0	1	0	1	0	0	1	1	1	0	1	0	0	1	0	0	1	1	1	0	0	0	1	1	12	
	Q5	0	1	1	0	0	0	1	0	1	1	0	1	0	1	1	0	1	1	1	1	1	1	0	1	0	1	15
	Q6	0	0	1	1	1	1	0	0	0	1	0	1	1	0	1	1	1	0	0	0	1	0	1	0	1	13	
	Q7	1	0	1	1	1	1	0	0	0	1	0	1	0	1	1	0	1	0	0	1	0	1	1	1	0	14	
	Q8	0	1	1	0	1	0	0	0	0	1	1	1	1	1	1	1	1	1	1	0	1	1	1	1	1	18	
	Q9	0	1	1	0	1	0	0	0	1	1	0	1	1	0	1	1	1	1	1	1	1	1	0	0	1	16	
	Q10	0	0	1	0	1	1	0	0	1	1	0	0	1	0	1	0	0	0	1	1	1	1	1	0	1	13	
	Q11	0	1	1	0	1	0	1	0	1	1	0	1	1	1	1	1	1	0	0	1	0	1	0	0	1	15	
	Q12	0	0	1	1	1	1	0	0	0	1	1	1	1	1	1	1	1	1	1	1	0	1	0	1	1	18	
	Q13	0	0	1	0	0	0	0	0	0	1	0	1	1	1	1	1	0	1	1	0	0	1	0	1	1	12	
	Q14	0	0	1	0	0	0	0	0	0	1	0	1	0	1	1	1	0	0	0	1	1	1	0	1	1	11	
	Q15	0	0	1	0	1	1	0	0	1	1	1	1	1	1	1	1	1	1	0	1	1	1	1	0	0	17	
	Q16	0	1	0	1	0	1	1	0	1	1	0	0	1	1	1	0	0	0	0	1	1	1	1	1	1	15	
	Q17	0	1	1	0	0	0	0	0	0	1	0	1	0	0	1	1	0	0	1	1	1	1	1	1	1	13	
	Q18	0	1	1	0	1	0	1	0	0	1	0	0	0	0	1	0	0	1	1	1	0	1	1	1	0	12	
		14,39 = Média																										

		Perguntas																									Resultado	
		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25		
Questionários	Q1	0	1	1	0	1	0	0	1	0	1	1	0	0	0	1	0	1	0	1	0	0	0	0	0	0	9	
	Q2	1	0	1	0	1	0	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	1	0	0	0	6	
	Q3	0	1	1	0	0	0	0	0	0	0	1	1	1	0	0	1	0	0	0	0	1	1	0	1	0	9	
	Q4	0	0	0	0	0	0	0	0	0	0	1	0	1	0	0	1	0	0	1	0	0	0	1	1	0	1	7
	Q5	0	1	0	0	0	0	0	0	0	1	1	0	1	0	0	1	1	1	0	0	1	1	1	0	0	1	11
	Q6	0	1	0	0	1	0	0	0	0	0	1	0	1	1	0	1	1	1	1	0	1	1	0	0	0	1	12
	Q7	0	1	0	0	1	0	0	0	0	0	1	0	1	0	0	1	1	1	0	0	1	1	0	0	0	1	10
	Q8	0	1	1	0	1	0	0	0	0	0	1	0	0	0	0	1	1	0	0	0	1	1	0	1	1	1	11
	Q9	0	1	1	0	1	1	0	1	0	0	1	0	1	0	0	1	1	0	0	1	1	0	0	1	0	1	13
	Q10	0	1	1	0	0	0	0	1	1	0	0	1	1	1	1	1	1	1	0	1	1	0	0	1	0	1	15
	Q11	0	1	1	0	0	0	0	0	0	1	1	0	1	0	0	1	1	1	1	0	0	0	1	1	0	1	12
	Q12	0	1	1	0	0	0	0	0	0	0	1	1	0	1	0	1	1	0	0	1	1	1	0	0	0	0	10
	Q13	0	1	1	0	0	0	0	0	0	0	1	1	0	1	1	1	1	1	1	0	1	0	0	0	0	1	12
																											10,54 = Média	

Tabela E.2 – Resultados do questionário de conhecimentos de Engenharia de Software do Grupo B (antes de jogar o SimSE).

		Perguntas																									Resultado
		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	
Questionários	Q1	1	1	1	0	1	1	0	1	0	1	1	1	0	0	1	1	1	1	1	1	1	1	0	1	0	18
	Q2	1	0	0	0	0	0	1	0	0	1	0	1	0	0	1	1	1	1	0	0	0	1	0	1	1	11
	Q3	0	0	0	0	0	0	1	0	1	0	0	0	1	0	1	1	1	0	0	0	1	0	0	0	1	8
	Q4	0	0	1	0	1	0	0	0	1	1	0	0	1	0	1	1	0	1	1	1	1	0	1	1	1	14
	Q5	1	0	1	1	0	0	0	0	1	1	0	1	1	0	1	1	0	0	0	1	1	1	1	0	1	14
	Q6	0	0	0	1	0	0	0	0	1	1	0	1	0	0	1	1	1	1	0	1	1	1	1	0	1	13
	Q7	0	1	0	0	0	0	0	0	1	1	0	1	0	0	1	1	1	0	1	1	1	1	0	0	1	12
	Q8	0	0	0	0	0	0	0	1	0	1	0	1	1	0	1	1	1	0	1	1	1	1	1	0	0	12
	Q9	1	0	1	0	0	0	0	1	0	1	1	1	1	1	1	1	1	1	1	1	1	1	0	1	0	17
	Q10	1	0	1	1	1	0	0	1	0	1	1	1	0	1	1	1	1	0	1	1	1	1	1	0	1	18
	Q11	1	0	1	0	1	0	1	0	1	1	0	1	0	1	1	1	1	1	0	0	0	1	0	0	1	14
	Q12	0	1	0	0	1	0	1	0	1	1	0	1	1	0	1	1	1	0	0	1	1	1	0	1	1	15
	Q13	0	1	1	0	1	1	1	1	0	1	0	1	1	0	1	1	1	0	1	0	1	0	1	0	1	16
																											14 = Média

Tabela E.3 – Resultados do questionário de conhecimentos de Engenharia de Software do Grupo B (depois de jogar o SimSE).

Anexo F

Guião do Jogo

Na próxima página encontra-se o guião de jogo, fornecido na experiência durante a fase de jogo aos alunos dos grupos A e B.

Guia de Jogo

Bem vindo à experiência de jogo do SimSE!

A tua tarefa de jogo é gerir um projeto de desenvolvimento de software, o Groceries@Home. Para isso podes contar com:

- **280,000\$ de orçamento inicial** – está atento que pode aumentar (com novos requisitos);
- **1,350 clock ticks (tempo)** – está atento que pode aumentar (com novos requisitos);
- **Uma equipa de desenvolvimento** – cada elemento tem a sua experiência nas diferentes tarefas (tem isso em conta), o seu salário, a sua energia e motivação. Tem em conta que, se estiverem contratados (não os despedires), os membros da equipa recebem mesmo que não estejam a fazer nenhuma tarefa.
- **Ferramentas de desenvolvimento** – ajudam na tarefa, mas podem ter um custo.

O modelo utilizado na versão do jogo que vamos jogar é o modelo *Waterfall*, ilustrado abaixo, que refere que apenas devemos passar para a fase seguinte de um projeto (pela ordem ilustrada), se a fase anterior estiver completa e revista.

