



Bruno Emanuel Nunes Sampaio

Licenciatura em Engenharia Informática

EpiK: Plataforma para Desenvolvimento de Jogos para Aprendizagem Colaborativa e Interativa

Dissertação para obtenção do Grau de Mestre em
Engenharia Informática

Orientadora : Fernanda Barbosa, Prof. Auxiliar, Departamento de Informática, Faculdade de Ciências e Tecnologia da Universidade Nova de Lisboa

Co-orientadora : Carmen Morgado, Prof. Auxiliar, Departamento de Informática, Faculdade de Ciências e Tecnologia da Universidade Nova de Lisboa

Júri:

Presidente: [Nome do presidente do júri]

Arguente: [Nome do primeiro arguente]

Vogal: [Nome do primeiro vogal]

Epik: Plataforma para Desenvolvimento de Jogos para Aprendizagem Colaborativa e Interativa

Copyright © Bruno Emanuel Nunes Sampaio, Faculdade de Ciências e Tecnologia, Universidade Nova de Lisboa

A Faculdade de Ciências e Tecnologia e a Universidade Nova de Lisboa têm o direito, perpétuo e sem limites geográficos, de arquivar e publicar esta dissertação através de exemplares impressos reproduzidos em papel ou de forma digital, ou por qualquer outro meio conhecido ou que venha a ser inventado, e de a divulgar através de repositórios científicos e de admitir a sua cópia e distribuição com objectivos educacionais ou de investigação, não comerciais, desde que seja dado crédito ao autor e editor.

Agradecimentos

Em primeiro lugar, gostaria de agradecer às minhas orientadoras, Professora Doutora Fernanda Barbosa e Professora Doutora Carmen Morgado, por todo o seu apoio, críticas, sugestões e dedicação durante a realização desta dissertação.

Na faculdade, agradeço também ao Departamento de Informática por ter fornecido o servidor utilizado para a instalação do Moodle e do Epik e ao Professor Doutor Paulo Lopes pela sua ajuda na configuração e instalação do servidor.

Em casa, agradeço à minha irmã por me ter ajudado em algumas decisões a nível de *design* do tema Cubic e do Epik e por ter criado algumas das imagens e atividades utilizadas em alguns dos jogos criados no Epik. Agradeço também à minha mãe e ao meu pai por todo o apoio que me deram ao longo desta fase e pelo seu esforço para que conseguisse terminar o curso.

Agradeço também à Sónia Martins pela criação de grande parte dos ícones utilizados no tema Cubic e nos jogos do Epik. E pela sua constante paciência e dedicação para com todas as minhas exigências.

Agradeço ainda ao Rui Lopes pela ajuda que me deu na criação das atividades e recursos para o jogo de matemática de secundário.

Finalmente, agradeço a todos os anteriores e ao António Guiomar, à Hélia Marçal, ao João Gouveia e ao Ricardo Fernandes por todo o apoio, concelhos e motivação que me deram durante a realização deste trabalho.

Resumo

O uso das novas tecnologias, em particular de ferramentas Web sociais e colaborativas, é uma questão cada vez mais comum na atualidade. Essas ferramentas possibilitam várias formas de trabalhar, comunicar, interagir e colaborar, e em alguns casos podem ainda ser integradas a outros sistemas. Um exemplo de ferramentas com essas funcionalidades são os videojogos.

Vários estudos confirmam que a aplicação dos jogos no ensino seria uma valia para o processo de aprendizagem. Estes viriam permitir formas de ensino e aprendizagem mais motivadoras, divertidas e adaptáveis aos comportamentos sociais da atualidade. No entanto, nem sempre os professores têm ao seu dispor ferramentas que permitem o desenvolvimento de jogos educativos, ou mesmo quando têm, de um modo geral, essas ferramentas não possibilitam a reutilização de materiais didáticos, meios de colaboração, nem formas práticas de distribuir os jogos pelos estudantes. Porém, os jogos poderiam ser distribuídos através de Sistemas de Gestão de Aprendizagem (LMS), os quais permitem a integração de outras atividades educativas além das que já disponibilizam. Contudo, essas atividades são geralmente individuais, não permitindo interação entre estudantes.

Nesta dissertação propõe-se uma abordagem para fornecer a professores ou instrutores uma forma simples de desenvolvimento e distribuição de jogos educativos com meios de colaboração. Para isso foi desenvolvida uma plataforma para desenvolvimento, distribuição e execução de jogos que pode ser usada por professores sem que estes necessitem de conhecimentos de programação. Esta plataforma (Epik) possibilita ainda a importação de conteúdos de um LMS existente (Moodle) e a distribuição dos jogos desenvolvidos nesse mesmo LMS.

Palavras-chave: Ambientes de Aprendizagem, Colaboração, Interação, Jogos Educativos.

Abstract

Nowadays, the use of new technologies, specifically social and collaborative web tools, is an increasingly common question. These tools allow for various forms of working, communicating, interacting and collaborating, and in some cases they may even be integrated with other systems. An example of tools that provide all those functionalities are video games.

Several studies confirm that the use of games in education would enhance the traditional learning process. These would allow forms of teaching and learning more motivating, fun and adaptable to today's social behaviors. However, teachers do not always have at their disposal tools that enable the development of educational games, or even when they have, in general, these tools do not allow the reuse of learning materials and do not provide means of collaboration between students, nor practical ways to distribute games by them. Though, games could be distributed through Learning Management Systems (LMS), which allow the integration of other educational activities beyond those they already provide. However, these activities are usually individual which means they do not allow interaction between students.

This thesis proposes an approach to provide teachers with an easy way to develop and distribute educational games with collaborative means. In order to achieve this, was developed a platform for development, distribution and execution of games that can be used by teachers without needing programming knowledge. This platform (Epik) also allows the import of educational contents from an existing LMS (Moodle) and the distribution of the developed games in this same LMS.

Keywords: Collaboration, Interaction, Educational Games, Learning Environments

Conteúdo

1	Introdução	1
1.1	Motivação	1
1.2	Enquadramento e contexto	3
1.3	Objetivos da dissertação	3
1.4	Estrutura do documento	4
2	Trabalho Relacionado	5
2.1	Ambientes e plataformas de ensino	5
2.1.1	Interação entre utilizadores	7
2.1.2	Perceção (<i>Awareness</i>)	8
2.1.3	Registos de atividades	9
2.1.4	Análise comparativa de LMS	10
2.2	Colaboração no processo de aprendizagem	11
2.3	Jogos no ensino	14
2.3.1	Conceitos, estrutura e propriedades dos jogos	15
2.3.2	Classificação de jogos educativos	17
2.3.3	Desenho e desenvolvimento	19
2.3.4	Ambientes de desenvolvimento de jogos	22
2.3.5	Integração dos jogos nos LMS	23
2.4	Conclusões	25
3	Plataforma Epik	27
3.1	Características gerais	28
3.2	Tipos de jogos	28
3.2.1	Jogos de Questionário	30
3.3	Desenvolvimento de jogos	36
3.3.1	Criação de atividades e recursos	39
3.3.2	Criação de projetos	39
3.3.3	Parametrização das propriedades gerais	40

3.3.4	Criação dos cenários e seus conteúdos	41
3.3.5	Geração do jogo	43
3.4	Distribuição de jogos	44
3.5	Execução de jogos	45
3.6	Integração no Moodle	46
3.7	Conclusões	49
4	Implementação	51
4.1	Servidor de gestão e desenvolvimento	51
4.1.1	Camada de apresentação	52
4.1.2	Camada de lógica	54
4.1.3	Camada de dados	55
4.2	Servidor de execução de jogos	60
4.2.1	Camada de comunicação	60
4.2.2	Camada de lógica	61
4.2.3	Camada de dados	62
4.2.4	Execução de um jogo	64
4.3	Extensões para o Moodle	71
4.4	Conclusões	73
5	Avaliação	75
5.1	Descrição dos inquéritos e grupos de utilizadores	75
5.2	Utilização dos jogos do Epik no Ensino	77
5.2.1	Opiniões dos professores	78
5.2.2	Opiniões dos estudantes	81
5.3	Desenvolvimento de jogos no Epik	83
5.3.1	Satisfação com a <i>dashboard</i>	84
5.3.2	Satisfação com o ambiente de desenvolvimento	87
5.3.3	Satisfação com a plataforma Epik	88
5.4	Jogabilidade dos jogos do Epik	89
5.4.1	Satisfação com a aparência e fluidez dos jogos	92
5.4.2	Satisfação com os jogos do Epik	94
5.4.3	Sugestões por parte dos jogadores	94
5.5	Utilização do tema Cubic	96
5.5.1	Satisfação com as funcionalidades do tema e do <i>chat</i>	97
5.5.2	Satisfação geral	98
5.6	Conclusões	99
6	Conclusões e Trabalho Futuro	101
6.1	Conclusões	101
6.2	Trabalho Futuro	103

A Inquéritos de Testes de Utilizadores	115
A.1 Utilização e desenvolvimento de jogos	115
A.2 Utilização e jogabilidade dos jogos	117
A.3 Utilização do tema Cubic para o Moodle	119
B Diagramas de Atividades	121

Lista de Figuras

2.1	Modelo 3C de Colaboração [Koc08].	12
2.2	Sistema da teoria da atividade num ambiente de aprendizagem (adaptado de [CL96, PMP10]).	12
2.3	Aplicações colaborativas classificadas segundo o Modelo 3C de Colaboração.	13
2.4	Taxonomia de categorização dos jogos educativos (adaptada de [WOMS09]).	18
2.5	Arquitetura de dois sistemas que comunicam entre si através do IMS LTI (adaptada de [IMS12, Cha10]).	25
3.1	Fluxo geral de um jogo do Epik.	30
3.2	Tipos de cenários dos jogos de Questionário do Epik.	31
3.3	Painel de jogadores dos jogos de Questionário do Epik.	33
3.4	Situação de colaboração num jogo de Questionário Colaborativo.	35
3.5	Área de gestão de conteúdos do Edutainment by Playing and Interacting with Knowledge (Epik) com a secção de atividades selecionada.	37
3.6	Ambiente de desenvolvimento de um jogo no Epik.	37
3.7	Barra de ferramentas do ambiente de desenvolvimento de jogos do Epik.	38
3.8	Janela de escolha de <i>template</i> durante a criação de um projeto no Epik.	40
3.9	Exemplos do conteúdo do painel de propriedades do ambiente de desenvolvimento do Epik.	42
3.10	Área de consulta dos registo de sessões de um jogo.	45
3.11	Área de consulta dos registo de uma sessão de um jogo.	46
3.12	Interface de utilizador do tema Cubic.	47
3.13	Integração do Epik no Moodle.	48
4.1	Arquitetura do servidor de gestão e desenvolvimento do Epik.	53
4.2	Esquema de tabelas da base de dados de utilizadores do Epik.	56
4.3	Representação em árvore da estrutura interna de um projeto do Epik.	58
4.4	Arquitetura do servidor de execução de jogos.	61
4.5	Esquema de tabelas da base de dados de jogos do Epik.	63

4.6	Diagrama de atividades para o pedido de registo de jogador.	65
4.7	Diagrama de atividades para o pedido de iniciação de jogo.	66
4.8	Diagrama de atividades da notificação de resposta a pergunta.	67
4.9	Diagrama de atividades da notificação de uso de ajuda.	68
4.10	Diagrama de atividades da notificação de resposta a um pedido de ajuda.	69
4.11	Diagrama de atividades de uma mudança de cenário.	70
5.1	Graus de instrução dos participantes na avaliação de utilização e desenvolvimento de jogos.	76
5.2	Graus de instrução dos participantes na avaliação de utilização e jogabilidade dos jogos.	77
5.3	Graus de instrução dos participantes na avaliação do tema Cubic.	77
5.4	Interesse dos professores na utilização de jogos no ensino.	78
5.5	Interesse dos professores na utilização dos jogos de Questionário do Epik no ensino.	79
5.6	Interesse dos professores na ajuda de remoção de respostas.	80
5.7	Interesse dos professores nos jogos de Questionário do Epik.	80
5.8	Interesse dos estudantes pela utilização de jogos no ensino.	81
5.9	Interesse dos estudantes pela utilização de atividades de questionário no ensino.	82
5.10	Interesse dos estudantes pela utilização dos jogos de Questionário do Epik no ensino.	82
5.11	Avaliação geral dos jogos do Epik como atividade educativa por parte dos estudantes.	84
5.12	Facilidade na realização de tarefas de criação no Epik.	85
5.13	Interesse da possibilidade de importar conteúdos do Moodle.	86
5.14	Utilidade dos <i>templates</i> de projetos do Epik.	86
5.15	Nível de satisfação com a <i>dashboard</i> de um modo geral.	87
5.16	Utilidade dos <i>templates</i> de cenários.	87
5.17	Organização do ambiente de desenvolvimento do Epik.	88
5.18	Facilidade do processo de desenvolvimento de um jogo no Epik.	89
5.19	Palavras caracterizadoras do Epik.	90
5.20	Grau de satisfação com a simplicidade de utilização do Epik.	90
5.21	Quantidade de pessoas que recomendariam o Epik a outros amigos/colegas.	91
5.22	Interesse do painel de jogadores como forma de percepção do estado do jogo.	92
5.23	Simplicidade do fluir dos jogos de Questionário do Epik.	93
5.24	Satisfação com os jogos do Epik como atividades educativas.	94
5.25	Palavras caracterizadoras dos jogos do Epik.	95
5.26	Estudantes que recomendariam os jogos do Epik a outros amigos/colegas.	95
5.27	Utilidade da barra de utilizador do Cubic.	97

5.28 Utilidade do conceito de aplicações no Modular Object-Oriented Dynamic Learning Environment (Moodle).	97
5.29 Palavras caracterizadoras do tema Cubic.	98
B.1 Diagrama de atividades da notificação de clique em botão de continuar ou saltar.	121
B.2 Diagrama de atividades da notificação de fim do tempo de permanência no cenário.	122
B.3 Diagrama de atividades da notificação de término de jogo.	122
B.4 Diagrama de atividades da notificação de desconexão de um jogador. . . .	123

Lista de Siglas e Acrónimos

API

Application Programming Interface. [10](#), [13](#), [15](#), [48](#), [52](#), [54](#), [71](#), [72](#)

CSS

Cascading Style Sheets. [53](#), [72](#)

Epik

Edutainment by Playing and Interacting with Knowledge. [xv](#), [3](#), [4](#), [27–30](#), [36–41](#), [43–46](#), [48](#), [49](#), [51](#), [52](#), [54–56](#), [59](#), [60](#), [62](#), [63](#), [71–73](#), [75–78](#), [80](#), [81](#), [83](#), [84](#), [86](#), [88](#), [89](#), [92](#), [94](#), [96](#), [99](#), [101–106](#), [115](#), [117](#)

HTML

HyperText Markup Language. [52](#), [53](#), [72](#), [84](#)

HTTP

Hypertext Transfer Protocol. [52](#), [54](#), [73](#)

IMS LTI

IMS Learning Tools Interoperability. [24–26](#), [44](#), [47](#), [52](#), [54](#), [63](#), [71](#), [102](#), [104](#)

JSON

JavaScript Object Notation. [54](#), [55](#), [57](#), [60](#), [63](#)

LMS

Learning Management System. [2](#), [3](#), [5–13](#), [15](#), [22–28](#), [30](#), [33](#), [44–46](#), [49](#), [51](#), [52](#), [54](#), [55](#), [57](#), [63](#), [71–73](#), [77](#), [98](#), [101](#), [102](#), [104](#)

Moodle

Modular Object-Oriented Dynamic Learning Environment. [xvii](#), [2–4](#), [6](#), [10](#), [11](#), [15](#), [23](#), [24](#), [26](#), [27](#), [30](#), [39](#), [44–49](#), [51](#), [52](#), [54](#), [55](#), [71–73](#), [75–78](#), [83](#), [85](#), [96](#), [97](#), [99](#), [101](#), [102](#), [104](#)

MVC

Model-View-Controller. [52](#)

REST

REpresentational State Transfer. [48, 54](#)

SCORM

Sharable Content Object Reference Model. [24](#)

XML

Extensible Markup Language. [23, 54, 55, 57–59, 63, 64](#)

1

Introdução

As novas tecnologias fazem hoje em dia parte das nossas vidas, tendo vindo com o tempo a simplificar a realização de tarefas complexas ou anteriormente impossíveis, deixando estas, em alguns casos, de estar dependentes do espaço e do tempo. As pessoas começam a interagir com estas tecnologias cada vez mais cedo, sendo surpreendente a facilidade de aprendizagem e compreensão das gerações mais novas. Estas tecnologias caracterizam-se por fomentar a colaboração entre pessoas de uma forma muito interativa e, por vezes, também divertida. No entanto, o ensino baseia-se ainda em formas de aprendizagem mais tradicionais que são por alguns consideradas menos apelativas devido à sua falta de interatividade e de formas de colaboração. Por essas razões, o ensino necessita de se adaptar aos comportamentos atuais e para isso é necessário desenvolver atividades educativas que tirem partido das novas tecnologias. Essas atividades devem fornecer aos estudantes uma forma mais divertida de aprender que englobe principalmente meios de colaboração e interação entre estes.

1.1 Motivação

Desde o aparecimento dos computadores e da internet que se realizam inúmeros estudos sobre a utilização das novas tecnologias no ensino. Estas tecnologias fornecem uma componente extra fora da sala de aula, possibilitando a aprendizagem em qualquer lugar e em qualquer momento, tornando o processo de aprendizagem mais colaborativo, interativo e dinâmico.

Os meios de comunicação e colaboração entre estudantes e professores são a base do ensino, seja ele presencial ou à distância. É através destes meios que os professores ensinam aos estudantes novos conceitos e lhes transmitem a importância do trabalho em

equipa e do sentido de entreajuda. Porém, estes meios são ainda um dos maiores obstáculos quando se fala da sua integração em atividades educativas que tiram partido das novas tecnologias. Por vezes, as opções de comunicação que estas tecnologias disponibilizam são bastante limitadas, não sendo sequer possível comunicar em tempo real.

No entanto, já existem ferramentas de colaboração que não necessitam da presença de todos os participantes num só local físico, mas o seu público-alvo nem sempre é o ensino [KRCB10, RF11, HBL⁺09, Ash09]. Um exemplo dessas tecnologias são os videojogos (como o The Swedish Armed Forces [For12b]), os quais proporcionam já formas bastante ricas de comunicação, interação e colaboração entre os seus utilizadores. Esses jogos fazem, atualmente, parte da cultura dos estudantes, sendo algo que os pode cativar mais a aprender do que um livro devido à sua componente mais interativa, dinâmica e competitiva.

Contudo, os jogos educativos raramente proporcionam formas de comunicação ou colaboração entre os seus jogadores, tornando-se desinteressantes ao fim de algum tempo [WOMS09, MTJV09, CMH08, JED⁺11]. Estes geralmente são utilizados para ensinar tópicos específicos seguindo um modelo base, que consiste em intercalar a resolução de exercícios com uma história contada ao longo do jogo, utilizando texto e elementos multimédia, que se relacionam de alguma forma com os exercícios. A competição é também uma forte componente destes jogos quando jogados por mais do que uma pessoa, porém apesar de esta contribuir para um aumento da motivação dos estudantes, quando aplicada sem meios de colaboração poderá levar a uma diminuição da capacidade de trabalho em equipa [PB07, VC03].

São muitos os trabalhos de investigação que defendem que a utilização de jogos seria uma mais valia para o processo de aprendizagem [RF11, Gra09, McG11, TMGMOFM09]. No entanto, é necessário encontrar soluções que facilitem o seu desenvolvimento e a sua distribuição pelos estudantes. Tema que tem sido tópico de investigação na atualidade [TMGMOFM09, BTMGM09, Moo12b].

Atualmente, para o desenvolvimento de jogos educativos são geralmente necessários conhecimentos de programação e a distribuição destes jogos pelos estudantes também não se revela uma tarefa fácil [TMGMOFM09]. Normalmente, isso requer a instalação dos jogos no seu computador pessoal e para os obterem poderá ser necessário fazer um investimento prévio. Além disso, estes jogos deveriam guardar informação acerca do desempenho dos jogadores ao longo do jogo, o que raramente acontece [MGBMO⁺08]. Sem essa informação um professor não pode avaliar ou ajudar os jogadores (estudantes), nem pode determinar se estes terminaram o jogo com sucesso ou insucesso.

Por outro lado, os sistemas de gestão de aprendizagem (*Learning Management System (LMS)*), como o *Moodle* [Moo12a], oferecem já formas de distribuição de atividades que armazenam informação acerca do desempenho dos jogadores [Gav12]. Por essa razão foram já considerados como uma solução para a distribuição dos jogos [TMGMOFM09]. Porém, a integração de jogos que disponibilizem formas de interação entre jogadores nestes ambientes está ainda em investigação.

1.2 Enquadramento e contexto

Esta dissertação está inserida num projeto de investigação que pretende desenvolver ambientes e ferramentas colaborativas, tendo como domínio de aplicação o processo de aprendizagem no ensino. O principal objetivo deste trabalho é a implementação de ferramentas educativas e colaborativas que integram as novas tecnologias e os novos comportamentos sociais de uma forma harmoniosa para alunos e professores, possibilitando assim uma aprendizagem mais apelativa e com espírito de entreajuda que se poderá realizar em qualquer momento e em qualquer lugar.

Os tópicos de investigação neste domínio de aplicação estão relacionados com:

- **Ambientes de aprendizagem** - o que engloba sistemas computacionais disponíveis na internet, destinados ao suporte de atividades mediadas pelas tecnologias da informação e comunicação, permitindo desenvolver interações entre pessoas (professores e estudantes) e objetos do conhecimento, e socializar tendo em vista atingir determinados objetivos (aprendizagem). Estes ambientes normalmente são implementados tendo como base os Sistemas de Gestão de Aprendizagem ([LMS](#));
- **Software colaborativo** - o que engloba as ferramentas computacionais que apoiam o trabalho em grupo e que podem ser usadas no processo de aprendizagem num ambiente colaborativo, o qual está dependente de variáveis de interação e localização;
- **Jogos educativos** - o que engloba *software* colaborativo cujo principal objetivo é o ensino, e permite estimular/motivar os alunos para a aquisição de conhecimentos.

1.3 Objetivos da dissertação

Esta dissertação tem como principal objetivo a criação e avaliação de uma plataforma para o desenvolvimento, distribuição e execução de jogos educativos. Esta plataforma, intitulada [EpiK](#), deve permitir o desenvolvimento de jogos educativos com as seguintes características:

1. Adaptáveis a qualquer área de ensino;
2. Adaptáveis e úteis para a aprendizagem colaborativa;
3. Fáceis de distribuir nos ambientes de ensino [LMS](#).

O desenvolvimento de jogos nesta plataforma deverá ser uma tarefa simples que poderá ser realizada por professores ou instrutores de qualquer área, sem necessitarem de conhecimentos de programação. Esta plataforma deverá ainda ser integrável num ambiente de ensino ([Moodle](#)) e permitir a utilização de conteúdos didáticos já existentes nesse ambiente nos jogos.

A avaliação da plataforma e dos jogos nela desenvolvidos será feita através de testes de utilização e inquéritos, realizados num grupo de professores e alunos de diferentes níveis e áreas de ensino.

1.4 Estrutura do documento

Este documento está divido em seis capítulos principais. No capítulo 1, foi introduzido o tema do trabalho, começando por apresentar a sua motivação, a área em que se enquadra e os seus objetivos.

No capítulo 2, são apresentados alguns trabalhos e tecnologias existentes, nomeadamente as características desejadas nos ambientes de aprendizagem, e a utilização de *software* colaborativo no processo de aprendizagem, com especial ênfase aos jogos educativos. Este estudo permitirá identificar as características desejadas e as tecnologias existentes que devem ser usadas na plataforma a desenvolver, assim como nos jogos educativos nela desenvolvidos.

No capítulo 3, é apresentada a plataforma proposta ([Epik](#)), no que diz respeito aos tipos de jogos que nesta podem ser desenvolvidos, ao processo de desenvolvimento, à distribuição e à execução de um jogo e à sua integração no [Moodle](#).

No capítulo 4, é descrita a implementação da plataforma [Epik](#) no que diz respeito às tecnologias e arquiteturas utilizadas. Do mesmo modo, é ainda descrita a implementação do trabalho realizado no [Moodle](#) para permitir a integração da plataforma neste último, entre outros.

No capítulo 5, apresenta-se a avaliação realizada à plataforma [Epik](#), no que diz respeito ao desenvolvimento de jogos e à utilização de jogos. Para além disso, será apresentada também uma avaliação à utilização dos componentes desenvolvidos para o [Moodle](#).

Por fim, no capítulo 6 são apresentadas as conclusões desta dissertação e algumas sugestões para trabalho futuro.

2

Trabalho Relacionado

As atividades educativas baseadas nas novas tecnologias possibilitam formas de interação entre professores e/ou alunos, que contribuem para um aumento da sua motivação e capacidade de trabalho em equipa (colaboração), podendo este ser realizado em qualquer momento e em qualquer lugar [GHS⁺⁰⁹]. Essas atividades e os meios de colaboração que oferecem devem ser integrados em ambientes de aprendizagem, como os LMS. Estes últimos têm características e regras bem definidas, desejáveis para o processo de aprendizagem, mas que necessitam de se adaptar a esse tipo de atividades. Um exemplo de atividades colaborativas, são os jogos educativos, os quais incentivam a aprendizagem de uma forma mais motivadora e divertida para os estudantes [Don07, WOMS09, ZSG⁺⁰⁹, McG11]. No entanto, esses jogos são geralmente difíceis de desenvolver e de integrar nos LMS, tornando-se necessário encontrar soluções para esses problemas.

2.1 Ambientes e plataformas de ensino

Um ambiente de aprendizagem, de modo geral, é um sistema computacional onde professores e estudantes podem partilhar vários tipos de informações, independentemente do espaço e do tempo utilizando as tecnologias da informação. Estes sistemas disponibilizam, por norma, formas de avaliação e treino para os estudantes, áreas de apoio e discussão, gestão de disciplinas e materiais de estudo, organização de eventos, entre outros [SBS99, IG12]. Estes ambientes são por norma desenvolvidos com base numa plataforma de ensino, ou mais precisamente um LMS. Essas plataformas definem muitas das necessidades das instituições de ensino, dos professores e dos estudantes, a nível de gestão de conteúdos, partilha de informação e atividades de aprendizagem. Exemplos

dessas plataformas são o [Moodle](#) [Moo12a], o [Blackboard](#) [Bla12], o [Sakai](#) [Sak12] e o [Desire2Learn](#) [Des12], que são apresentados em maior detalhe na subsecção 2.1.4.

O aparecimento dos [LMS](#) deu origem a novas formas de ensino e de aprendizagem. O ensino à distância é uma delas, o qual veio permitir que pessoas com menos disponibilidade pudessem continuar os seus estudos sem necessitarem de ter aulas presenciais, tendo assim a possibilidade de estabelecer os seus próprios ritmos de estudo [IG12, Cou12, Mas12]. As aulas são-lhes disponibilizadas como gravações que podem ser visualizadas em qualquer altura e lugar, método esse que começa também a ser utilizado no ensino tradicional como forma de complemento às aulas presenciais. Para além disso, os professores podem disponibilizar aos estudantes vários tipos de atividades *online*, como questionários relacionados com uma determinada matéria, discussões de ideias em fóruns, e vídeos com conteúdo didático [Cou12].

Apesar da utilidade das funcionalidades oferecidas pelos [LMS](#), a forma como estas são apresentadas ao utilizador nem sempre é simples, prática e atrativa. Por essa razão, muitos professores preferem não os usar, ou não aprender a usá-los porque os consideram complicados e desinteressantes, acabando assim por afastar também os estudantes destes sistemas [CMH08, Mec08]. Em alguns casos, os professores preferem inclusivamente criar a sua própria página de disciplina, em vez de utilizarem os sistemas de ensino que lhes são disponibilizados. Do mesmo modo, os estudantes não encontram muitas ferramentas nos [LMS](#) que lhes facilitem a aprendizagem colaborativa, o que os leva a utilizar ferramentas externas de caráter social para partilharem informação com outros colegas e realizarem trabalhos de grupo.

A utilização desse tipo de ferramentas nos ambientes de aprendizagem poderia aumentar a interação entre os intervenientes no processo de aprendizagem, contribuindo assim de forma positiva para a sua motivação e espírito de entreajuda [MD07, LaP03, Gra09]. Esse facto é confirmado por estudos realizados em anos mais recentes [JT10, RR09, HBL⁺09, LP08, LT08, RF11, KRCB10, GHS⁺09, Mec08], que vieram chamar à atenção da importância dos processos de comunicação, colaboração e partilha de informação no ensino, utilizando ferramentas de caráter social. Essas ferramentas são hoje em dia disponibilizadas pelos serviços existentes na Web 2.0, os quais fazem já parte do quotidiano da maioria das pessoas. Exemplos dessas ferramentas são: os *chats*, fóruns, *wikis*, sistemas de videoconferência, de partilha de ficheiros e escrita colaborativa, e redes sociais. Todas elas oferecem formas distintas de colaboração, o que poderia não só resolver alguns dos problemas identificados anteriormente nos [LMS](#), como proporcionar formas ainda mais variadas de ensino e de aprendizagem.

Algumas dessas ferramentas já são disponibilizadas nos [LMS](#) na forma de atividades educativas. Contudo, essas atividades geralmente não proporcionam formas muito ricas de interação e colaboração entre os estudantes, em parte porque os [LMS](#) não proporcionam ainda os mecanismos necessários para a realização de atividades colaborativas, necessitando de se adaptar a esse conceito [Sab11]. Para isso, um ambiente de aprendizagem, de um modo geral, deverá garantir as seguintes características:

- **Meios de interação** [SBS99, KKJ02, Gra09, KRCB10] – a interação entre utilizadores deverá poder ser feita através de comunicação assíncrona e síncrona. Na subsecção 2.1.1 são apresentados alguns exemplos deste tipo de comunicação e porque é que ambos são importantes;
- **Indicadores de Perceção (Awareness)** [SBS99, CMH08] – conjunto de fatores ou indicadores que permitem a um utilizador de um sistema ter consciência da presença e das ações dos demais utilizadores. Na subsecção 2.1.2 são apresentados em maior detalhe os vários tipos de indicadores de *awareness* e exemplos destes;
- **Registros de atividades** [MD07] – as atividades educativas devem registar informação acerca do desempenho dos estudantes enquanto estes as realizam, como por exemplo, a sua participação em discussões de ideias, os seus resultados em avaliações, o seu desempenho num jogo, a sua interação com outros colegas, entre outros. Os tipos de registo de atividades que devem ser suportados por um ambiente de aprendizagem são apresentados em maior detalhe na subsecção 2.1.3.

Nas subsecções que se seguem, para além das características mencionados acima, é apresentada no final desta secção uma comparação entre os LMS mais utilizados atualmente no ensino, e que poderiam ser utilizados como forma de distribuição de atividades mais colaborativas e interativas.

2.1.1 Interação entre utilizadores

Para permitir a interação e por conseguinte a colaboração entre utilizadores a maioria dos sistemas existentes na Web 2.0 permitem comunicação assíncrona e síncrona. A primeira permite que uma pessoa contacte outra ou outras sem ser necessário que estas últimas estejam presentes, não sendo portanto comunicação em tempo real. Exemplos deste tipo de comunicação são o envio de cartas, e-mails, mensagens privadas (bastante comum em fóruns) e mensagens curtas (SMS). A segunda permite que duas ou mais pessoas comuniquem em tempo real quer seja via texto, áudio, ou vídeo. Exemplos deste tipo de comunicação são chamadas telefónicas, videoconferências e envio de mensagens via serviços de *chat*.

No processo de aprendizagem, a comunicação é obviamente bastante importante, por conseguinte, ambos os modos devem ser disponibilizados nos ambientes de aprendizagem [Gra09, KRCB10]. A comunicação assíncrona é necessária, por exemplo, para que um professor possa enviar mensagens aos seus alunos em qualquer momento, com anúncios de eventos, ou com conteúdo urgente. Já a comunicação síncrona será especialmente importante em atividades colaborativas, como por exemplo, aulas e trabalhos de grupo.

Um aspeto também bastante importante de ambas as formas de comunicação é a possibilidade de comunicar de um para um, ou de um para muitos. A primeira permite uma comunicação mais pessoal e privada, sendo mais apropriada para tirar dúvidas com um

professor, ou para discutir assuntos privados com um colega. A segunda, em alguns casos, já não é tão apropriada para realizar essas atividades, sendo geralmente utilizada para dar aulas, trabalhos de grupos, discussões de ideias e partilha de informação.

A maioria dos LMS permitem comunicação assíncrona de um para um, ou para muitos, a qual está presente no envio de mensagens e/ou fóruns. No entanto, a comunicação síncrona, geralmente só pode ser realizada via *chats* disponibilizados apenas como atividades dentro de um determinado contexto. Um exemplo de uma ferramenta que disponibiliza formas mais ricas de comunicação e interação e que já foi inclusivamente integrada no Moodle é a apresentada em [Wiz12]. Esta ferramenta pode ser utilizada para dar aulas à distância, pois permite a comunicação, de forma textual, visual e auditiva, oferece uma área para apresentação de slides ou desenho, permite a gravação da aula e fornece indicadores de percepção do meio envolvente.

2.1.2 Perceção (Awareness)

A *awareness* é a consciência que uma pessoa tem da presença e das ações de outras pessoas num mesmo meio [CMH08, SBS99]. Esta capacidade está sempre presente em atividades presenciais, mas é mais difícil de garantir em sistemas computacionais. Um dos grandes problemas causados pela falta desta capacidade nos ambientes de aprendizagem é a perda do *feedback* transmitido pelos estudantes (como contacto visual e atitude), que pode transmitir o estado de aprendizagem destes ao professor. Adicionalmente, os professores e estudantes necessitam de sentir que não estão sozinhos quando estão a ensinar ou a aprender, pois a dimensão social é crucial para os processos cognitivos implícitos nas atividades de aprendizagem [CMH08].

Esses são alguns dos problemas existentes nos ambientes de aprendizagem em geral, e a forma proposta para os solucionar foi a criação de vários indicadores de *awareness* relacionados com as seguintes categorias [CMH08]:

- **Informal** – diz respeito à consciência geral de presença, capacidade, ou disponibilidade de outros. Representa todas as coisas que são implicitamente sentidas por um professor numa sala de aula real. Formas de lidar com este problema são, por exemplo, a representação de cada utilizador através de um avatar, ou a utilização de vídeo em tempo real que capte as suas expressões;
- **Social** – diz respeito à sensação que é dada por outra pessoa num contexto de diálogo. Será que ele ou ela está a ouvir? Será que ele ou ela está interessado na discussão? Qual é o seu estado emocional? Uma forma já bastante utilizada para solucionar este problema é o uso de *smileys* na comunicação por via textual. Em comunicação via áudio ou vídeo este problema não se verifica tanto, visto que as componentes auditiva e visual estão presentes, transmitindo de melhor forma o estado emocional dos participantes;
- **Estrutura de Grupo** – diz respeito à consciência organizacional de um grupo em

termos do papel desempenhado, responsabilidade, ou estado de cada utilizador. São portanto necessárias formas de distinguir tipos de utilizadores diferentes (como professor e estudante), as suas responsabilidades (como regente e responsável de uma cadeira, ou chefe de um grupo) e os seus estados (como conectados, ausentes, ou desconectados);

- **Área de Trabalho** – representa a consciência de atividade em grupo (de cada utilizador) em relação à área de trabalho. Formas de transmitir essa consciência, do lado do professor, poderão ser os registos de atividades, ou, do lado do estudante, registos da participação dos seus colegas nas atividades do grupo.

Grande parte dos sistemas computacionais existentes hoje em dia, principalmente as redes sociais, já fornecem muitos indicadores que abordam alguns aspetos das três primeiras categorias de *awareness* identificadas acima. Exemplos desses indicadores são o uso de diferentes cores e texto para determinar o estado de um utilizador e as imagens de perfil. A última categoria, para além de ser a mais difícil de implementar, é também fundamental no processo de aprendizagem [CMH08], pois está mais relacionada com o trabalho colaborativo.

Nos **LMS**, os indicadores de *awareness* existentes estão relacionados principalmente com aspetos sociais (como o perfil de utilizador e o uso de *smileys*), estrutura de grupo (tipos e estados de utilizadores) e área de trabalho (registos de atividades).

2.1.3 Registos de atividades

Os registos de atividades guardam dados relacionados com atividades realizadas pelos estudantes, que ficam depois disponíveis para consulta do lado do professor. Estes dados ajudam o professor a avaliar o desempenho dos estudantes na realização de atividades de treino relacionadas com uma ou mais matérias, para que depois lhes possa oferecer apoio naquelas em que apresentam maiores dificuldades.

A apresentação destes registos é normalmente feita utilizando tabelas, no entanto, os dados nem sempre estão organizados de forma lógica, tornando-os difíceis de ler, ou até mesmo incompreensíveis, acabando assim por não servir o seu propósito. Esse foi um dos motivos que levou à realização do estudo apresentado em [MD07], onde o objetivo dos autores era a criação de formas mais apelativas e lógicas (utilização de gráficos) para apresentar os seguintes tipos de informação: (i) **Aspetos sociais** – interações entre diferentes tipos de utilizadores, tais como, a participação em discussões de ideias e trabalhos de grupo; (ii) **Aspetos cognitivos** – desempenho global de um estudante no curso e em comparação com outros estudantes, em questionários e trabalhos práticos, em tópicos específicos do curso, em jogos e outros; (iii) **Aspetos comportamentais** – características específicas do comportamento de um estudante, como por exemplo, participação em atividades e realização muito rápida ou muito lenta do programa do curso.

2.1.4 Análise comparativa de LMS

Os LMS mais usados na atualidade a nível mundial são o Moodle [Moo12a], o Blackboard [Bla12], o Sakai [Sak12] e o Desire2Learn [Des12]. Para cada um deles é fornecida, na lista que se segue, informação sobre o tipo de produto, qualidade da documentação e suporte, tipos de customização permitidos a nível de desenvolvimento, meios de comunicação disponibilizados, indicadores de *awareness*, registos de atividades e tipos de atividades.

Moodle [Moo12a] – é um sistema *open source*, atualmente na versão 2.3, escrito em PHP, suporta vários sistemas de gestão de bases de dados, disponibiliza cerca de 100 pacotes de línguas e é um dos mais utilizados a nível mundial. Pode ser instalado em quaisquer infraestruturas, inclusivamente na nuvem (*cloud*). Suporta vários tipos de *plugins* que permitem adicionar funcionalidades ao sistema, como uma nova aparência para a interface de utilizador (**temas**¹), novas atividades educativas (**atividades**²), novos blocos de informação (**blocos**³), entre outros. Oferece uma documentação bastante completa e organizada, e existe uma grande comunidade *online* disposta a responder a dúvidas e questões [Mec08]. Os meios de comunicação disponíveis por omissão, são apenas o envio de mensagens privadas, estando os restantes, como *chat* e fóruns, apenas disponíveis dentro de um curso criado por um professor. Permite a criação de diferentes tipos de atividades para as quais podem ser gerados registos de atividades. Fornece alguns indicadores de *awareness* social e de estrutura de grupo;

Blackboard [Bla12] – é um sistema comercial, que disponibiliza várias plataformas, sendo a mais utilizada o Blackboard Learn, atualmente na versão 9.1, escrito em Java. Não é claro se disponibiliza uma **Application Programming Interface (API)** que permite a customização do sistema, mas oferece vários blocos que permitem às instituições alterar determinados elementos do sistema, como a interface de utilizador. Para além disso, existem várias extensões desenvolvidas por parceiros que podem ser integradas no sistema. Oferece sistema de *chat*, *blogs*, *wikis*, fóruns e áreas pessoais, mas apenas se o professor os incluir num curso. Permite a integração com sistemas externos, como flickr⁴, Youtube⁵ e SlideShare⁶. Por último, disponibiliza vários indicadores de *awareness* e registos de atividades para avaliação dos estudantes;

Sakai [Sak12] – é um sistema *open source*, que disponibiliza neste momento duas plataformas diferentes, sendo a mais recente o Sakai Open Academic Environment (OAE) na versão 1.2, escrita em Java. A documentação disponibilizada é bastante

¹<http://docs.moodle.org/dev/Themes> consultado em 15-06-2012.

²http://docs.moodle.org/dev/Activity_modules consultado em 15-06-2012.

³<http://docs.moodle.org/dev/Blocks> consultado em 15-06-2012.

⁴<http://www.flickr.com/> consultado em 16-05-2012.

⁵<http://www.youtube.com/schools> consultado em 09-05-2012.

⁶<http://www.slideshare.net/> consultado em 25-05-2012.

completa e existem áreas de suporte de desenvolvimento. Permite que os estudantes adicionem, ou removam *widgets* ao sistema, que podem servir para integrar ferramentas externas, os quais são desenvolvidos e disponibilizados pela comunidade do Sakai, ou pelo próprio estudante. Disponibiliza várias formas de colaboração e comunicação, como envio de mensagens privadas, *chat*, fóruns, entre outras. Além disso, fornece vários indicadores de *awareness*, mas não foi encontrada informação relativamente a registos de atividades;

Desire2Learn [Des12] – é um sistema comercial, escrito em C# e .Net, que disponibiliza as suas próprias infraestruturas para instalar o sistema de cada instituição de ensino na nuvem, não sendo aparentemente possível instalar noutra local. Disponibiliza um *chat*, e-mail, *blogs*, áreas de discussão, e de partilha e visualização de ficheiros, sendo que todos esses elementos têm indicadores de *awareness*. Oferece a possibilidade de criar atividades de avaliação, como questionários e exames, a partir das quais são gerados registos de atividades.

Dos LMS aqui descritos, o mais utilizado é o **Moodle**, com cerca de 66000 *sites* registados no mundo inteiro [Moo12c]. Comparativamente aos outros é aquele que oferece um maior equilíbrio entre qualidade da documentação e suporte, e possibilidades de customização do sistema. Para além disso, é também o que oferece um maior número e variedade de atividades de aprendizagem. Por essas razões e por ser um sistema *open-source* foi o **LMS** escolhido para utilizar nesta dissertação.

2.2 Colaboração no processo de aprendizagem

Colaborar é trabalhar em conjunto com uma ou mais pessoas para atingir objetivos. Este é um processo que está presente nas nossas vidas desde muito cedo e que é bastante importante no processo de aprendizagem e no trabalho. Um ambiente colaborativo, como mostra a Figura 2.1, é sempre composto por três componentes fundamentais: comunicação, cooperação e coordenação (modelo 3C de colaboração) [Koc08]. Este facto é baseado no conceito de que para colaborar, os membros de um grupo comunicam, coordenam e cooperam. Vários estudos confirmam que os estudantes se sentem mais motivados para aprender, e são mais positivos em relação aos outros à sua volta, quando são utilizados métodos colaborativos [PB07, VC03, ZSG⁺09]. No entanto é necessário fornecer as condições necessárias para que a colaboração possa ocorrer, as quais podem estar relacionadas com aspectos sociológicos, psicológicos, e/ou tecnológicos.

Na Figura 2.2 são apresentados os vários elementos que fazem parte de um processo de colaboração e a forma como estes se interligam. A cada um desses elementos estão associados exemplos do que estes poderão representar na realização de uma atividade de aprendizagem. Para a realização de uma atividade deste género é, em primeiro lugar, necessário formar um sentido de **comunidade** entre os **indivíduos** que vão colaborar e em que estes se sintam à vontade [KKJ02]. A interação social é bastante importante

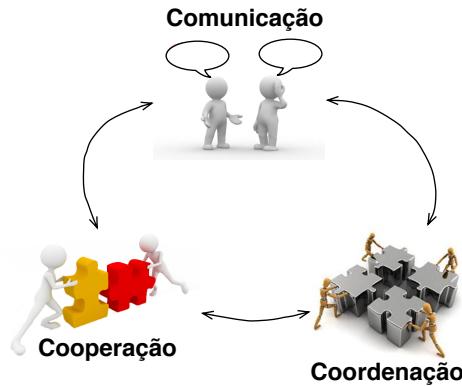


Figura 2.1: Modelo 3C de Colaboração [Koc08].

na criação desse sentido de comunidade, permitindo que se estabeleça um espaço social com uma estrutura que abrange as relações sociais, coesão de grupo, confiança e identificação com os restantes participantes. Esses aspectos contribuem para uma comunicação mais aberta, pensamento crítico, interações de apoio e negociação social, levando a uma **divisão de tarefas** entre os participantes. Além disso a aprendizagem colaborativa é geralmente mais efetiva em ambientes onde as pessoas sentem que têm um objetivo em comum (**resultado**), como no planeamento de um projeto, ou na discussão de ideias [SBS99]. Por último, é ainda necessário que a área de trabalho colaborativo disponibilize as **ferramentas** necessárias para a realização de atividades (**objeto**) dentro desse contexto social, de acordo com determinadas **regras**.

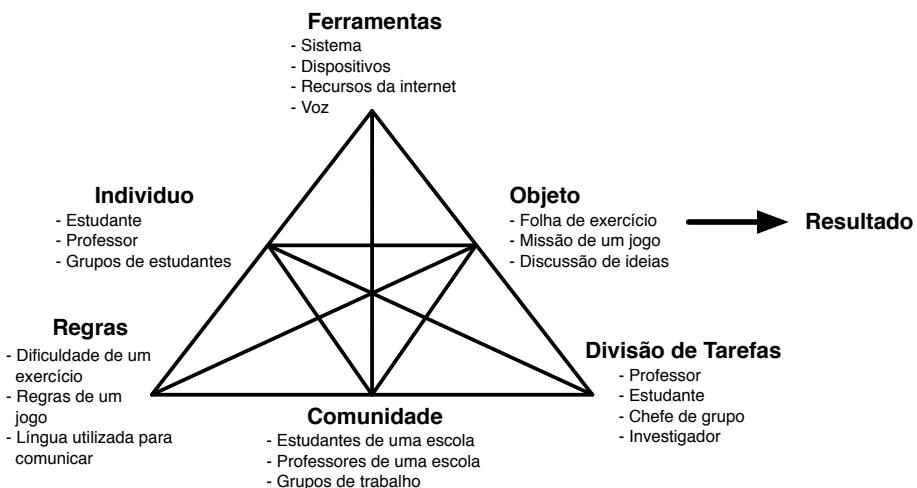


Figura 2.2: Sistema da teoria da atividade num ambiente de aprendizagem (adaptado de [CL96, PMP10]).

Dada a importância das atividades colaborativas no processo de aprendizagem, é necessário que existam ferramentas que permitam a definição dessas atividades nos **LMS**, pois geralmente é necessário utilizar aplicações externas que não interagem com o ambiente, perdendo-se assim o contexto social e colaborativo oferecido por este. Por exemplo,

dois estudantes envolvidos num trabalho de grupo que envolva a escrita de um documento, podem ter que alternar entre o ambiente de aprendizagem, para consultar conteúdos, e o editor de texto, para escrever e comunicar.

Com o aparecimento dos serviços de computação em nuvem (*cloud*) e com as funcionalidades oferecidas pela Web 2.0, começaram a surgir aplicações que permitem o trabalho colaborativo e que disponibilizam API's para integração com outros sistemas (um exemplo é o Google Drive [Goo13]). Alguns LMS também já permitem a integração com sistemas externos, possibilitando a troca de informação entre estes, no entanto, nem sempre se tira partido desta capacidade. Na Figura 2.3 são apresentadas algumas das aplicações colaborativas existentes na Web, classificadas segundo o modelo 3C de colaboração.

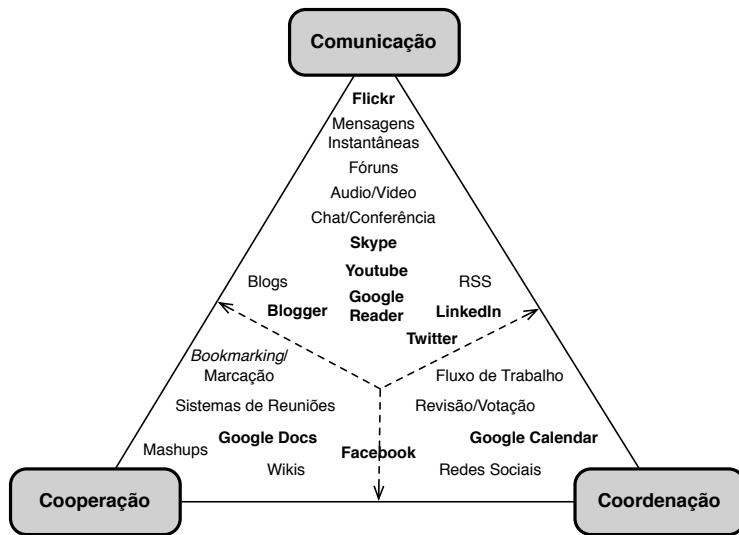


Figura 2.3: Aplicações colaborativas classificadas segundo o Modelo 3C de Colaboração.

Alguns exemplos de ferramentas colaborativas disponibilizadas por grande parte dos LMS e geralmente utilizadas pelos estudantes são os *blogs*, as *wikis*, os fóruns, os glossários e os *chats*. Na lista abaixo é apresentado o propósito de cada uma delas e as formas como são utilizados no processo de aprendizagem.

- Os *blogs* são páginas Web geridas, normalmente, por uma só pessoa, utilizadas para partilhar ideias, opiniões ou novidades. Qualquer pessoa com acesso à página pode fazer comentários às informações partilhadas. No ensino estes são geralmente utilizados com o objetivo de discutir assuntos e partilhar opiniões [Att07, KRCB10, HBL⁺09, Mec08]. Os estudantes apontam, como principais vantagens destes sistemas, o facto de permitirem a utilização de texto, imagens, vídeos e fontes externas para partilhar ideias, e o facto de uma discussão poder ser sempre continuada independentemente do espaço e do tempo [HBL⁺09];
- As *wikis* são semelhantes aos *blogs*, no que diz respeito à forma de colaboração que

oferecem, que é a discussão e partilha de ideias. A diferença é que todos os participantes podem alterar o texto introduzido por outros e contribuir para o melhorar. No ensino podem ser utilizadas em vários contextos que envolvam a escrita colaborativa, ou o desenvolvimento de uma ideia [Att07]. Para além das vantagens já referidas para os *blogs*, agrega-se o facto de ser possível alterar o conteúdo, o que é, por vezes, uma forma mais eficaz de consolidar uma ideia e de melhorar a escrita de um documento [HBL⁺09]. No entanto, nem sempre é possível garantir que a informação introduzida por outros é de confiança [Mec08];

- Os **fóruns** funcionam um pouco como os *blogs*, no entanto nestes qualquer pessoa pode iniciar um tópico de discussão. No ensino são geralmente utilizados para tirar dúvidas e partilhar informações que podem ser importantes para todos, ficando estas sempre disponíveis;
- Os **glossários** são dicionários de termos e respetivas definições partilhados e atualizados pelos estudantes, um pouco como o que acontece nas *wikis*, sendo o objetivo da sua utilização a definição em grupo de termos considerados importantes num dado contexto de ensino;
- Os **chats** permitem a comunicação síncrona entre dois ou mais elementos que se encontram *online*, de forma textual. São geralmente utilizados na realização de trabalhos de grupo, discussão de ideias e esclarecimento de dúvidas.

Um outro tipo de ferramenta/atividade que pode fornecer meios de colaboração e, inclusivamente, uma forma de aprendizagem mais divertida, são os jogos. No entanto, estes últimos não são ainda muito utilizados como atividade educativa quer esta seja individual ou em grupo.

2.3 Jogos no ensino

O uso de jogos no ensino como atividade educativa não é uma questão recente. Estes proporcionam formas de colaboração, interação e/ou competição que podem tornar uma atividade de aprendizagem muito mais divertida, interessante e envolvente, contribuindo assim para um aumento da motivação dos estudantes e aquisição de competências [Don07, WOMS09, SNJ09, PMP10]. Um jogo educativo pode ser utilizado no lugar dos exercícios tradicionais, não para os substituir, mas para proporcionar uma nova forma de aprender.

A natureza adaptável, interativa e dinâmica, dos jogos da atualidade, torna-se ideal para explicar determinados conceitos mais complexos, geralmente mais difíceis de entender na teoria, visto que oferecem também a possibilidade de ajustar a dificuldade da informação didática [PMP10]. Uma outra vantagem significativa é que os estudantes recebem imediatamente *feedback* sobre as suas ações e decisões, convidando-os a explorar e

experimentar, levando assim ao desenvolvimento de competências estratégicas por tentativa e erro [PMP10]. Em [Sch93, WOMS09] afirmam ainda que os jogos contribuem para um aumento das seguintes capacidades cognitivas: interpretação, análise, tradução, aplicação, síntese, avaliação, comunicação, imaginação, reutilização e *I-witnessing*⁷.

Apesar das vantagens referidas, os jogos não são ainda muito utilizados no ensino, em parte devido aos custos envolvidos no seu desenvolvimento e no investimento necessário por parte das instituições de ensino, e/ou dos estudantes [TMGMOFM09]. Por essas razões, torna-se necessário encontrar outras soluções para fornecer aos professores e estudantes meios de desenvolvimento e utilização dos jogos, respetivamente [TMGMOFM09]. A forma mais eficiente e prática de disponibilizar esses meios seria através dos LMS, tal como já acontece em algumas redes sociais. Atualmente, já existem trabalhos que tratam de incorporar os jogos nos LMS, como por exemplo, o *plugin* desenvolvido para o Moodle, apresentado em [Moo12b], o qual disponibiliza vários jogos tradicionais, como a Forca, o Sudoku e as Palavras Cruzadas. Ou ainda, o <e-Adventure> [e-U12, MGBMO⁺08, TMGMOFM09], ferramenta para o desenvolvimento de jogos educativos por parte de professores. No entanto, ferramentas como esta última são pouco conhecidas e, por vezes, não fornecem muita informação sobre como desenvolver um bom jogo educativo, o que não é uma tarefa fácil. Para além disso, algumas delas necessitam ainda de conhecimentos de programação, algo que nem todos os professores estão dispostos a aprender.

De modo a entender melhor o desenvolvimento, uso e integração dos jogos no ensino, nas subsecções que se seguem são apresentados e explicados alguns conceitos relacionados com estes, formas de os classificar, desenhar e desenvolver.

2.3.1 Conceitos, estrutura e propriedades dos jogos

Um jogo, independentemente do seu objetivo, é sempre composto por três elementos principais: o núcleo do jogo, o motor de jogo e a interface de utilizador [ZSG⁺09]. O **núcleo do jogo** é o elemento mais importante, pois é este que define todas as características do jogo, como o género, as regras, os desafios e os objetivos deste. O **motor de jogo** é um sistema que possibilita o desenvolvimento de um determinado género de jogos, composto por um conjunto de rotinas que permitem definir as ligações entre os vários elementos base do jogo. Estes motores fornecem geralmente uma **API** que pode ser usada pelos programadores no processo de desenvolvimento. As principais funcionalidades tipicamente oferecidas por estes incluem um motor de processamento de gráficos 2D ou 3D, um motor de física ou de deteção de colisões, sons, *scripting*, animações, inteligência artificial, *networking*, *streaming*, gestão de memória, *threading*, suporte de localização, e um grafo de cenários. Alguns exemplos desse tipo de motores são o EaselJS⁸, o Crafty⁹ e

⁷Testemunho de uma situação que de outra forma seria impossível devido a fatores como o tempo, distância, cultura, ou outros.

⁸<http://www.createjs.com/#!/EaselJS> consultado em 26-05-2012.

⁹<http://craftyjs.com/> consultado em 26-05-2012.

o GameQuery¹⁰ em HTML5, e o Flixel¹¹, o FlashPunk¹² e o PushButton¹³ em Flash. Relativamente à **interface de utilizador**, esta representa o meio pelo qual o jogador interage com o jogo, sendo essa interação caracterizadora do género do jogo que se está a jogar.

O **género de jogo** é uma forma de classificar um jogo quanto ao estilo de interação que disponibiliza. Existem várias taxonomias para esse efeito, sendo a mais usada a dos jogos comerciais, a qual os divide em oito categorias: jogos de ação, aventura, combate, quebra-cabeças, *role playing*, simulação, desportos, e estratégia [WOMS09]. Para cada uma dessas categorias podem estar disponíveis dois **modos de jogador**: *singleplayer* e *multiplayer*. No primeiro caso o jogador joga sozinho, tendo apenas a possibilidade de colaborar ou competir com o sistema na realização de algumas tarefas, ou missões, enquanto que no segundo caso há vários jogadores a competir ou colaborar na realização de tarefas no jogo, num mesmo local físico, ou distribuídos geograficamente. Por norma ambos os modos de jogador disponibilizam ainda a possibilidade de definir a **dificuldade** do jogo, que geralmente pode ser uma das seguintes: fácil, normal, ou difícil.

Os tipos de ações disponibilizadas ao jogador são definidas e influenciadas pelo tipo de fantasia utilizado no jogo, essa fantasia pode ser [WOMS09]: extrínseca, ou intrínseca. Uma **fantasia extrínseca** depende do uso de habilidades que não são influenciadas pela fantasia do jogo, isto é, o jogador utiliza as suas próprias habilidades e não outras que só existem no jogo. Um exemplo de jogo com este tipo de fantasia é a força, em que a salvação do homem depende da habilidade do jogador de desvendar cada letra da palavra. Por outro lado, uma **fantasia intrínseca** depende do uso de habilidades que fazem parte da fantasia, isto é, habilidades imaginadas ou visualizadas pelo jogador que não dependem dos seus conhecimentos ou capacidades. Este tipo de fantasia é o mais utilizado nos jogos, por exemplo, num simulador de situações reais o jogador utiliza habilidades que lhe são introduzidas pelo jogo e de que este não tinha conhecimento anteriormente.

Um jogo pode ser estruturado como um conjunto de cenários, níveis, ou missões [WOMS09, SC09, TMGMOFM09]. Um **cenário** representa uma determinada situação, ou área do jogo em que são dadas várias possibilidades de interação ao jogador. Algumas dessas possibilidades permitem alterar o estado do cenário, ou mudar para outro. Os cenários encontram-se normalmente ligados entre si de forma linear, ou hierárquica. Um **nível**, ou **missão** é composto por um ou mais cenários interligados. Toda esta estrutura representa o fluxo do jogo, ou o grafo de cenários, que se inicia geralmente num cenário e pode terminar em vários cenários distintos dependendo do caminho seguido pelo jogador. Geralmente um jogo é constituído, pelo menos, pelos seguintes tipos de cenários [SC09]:

- **Título** – cenário de início que contém informação sobre o objetivo e tema do jogo;

¹⁰<http://gamequeryjs.com/> consultado em 26-05-2012.

¹¹<http://flixel.org/> consultado em 09-06-2012.

¹²<http://flashpunk.net/> consultado em 09-06-2012.

¹³<http://pushbuttonengine.com/> consultado em 09-06-2012.

- **Menu Principal** – cenário a ser apresentado após iniciar o jogo. Fornece informações sobre este, opções para iniciar, ou carregar um jogo e para sair;
- **Cenários de Jogo** – cenários interativos onde o jogo se desenvolve;
- **Menu de Pausa** – geralmente idêntico ao menu principal, mas com uma opção para continuar, em vez de iniciar um novo jogo;
- **Cenários de Vitória** – aparecem depois de terminar um nível, uma missão, ou o jogo e costumam apresentar as pontuações e recompensas do jogador;
- **Cenários de Derrota** – aparecem sempre que o jogador perde todas as vidas, ou falha objetivos.

Um elemento também muito importante, presente na maioria dos jogos, são as personagens. Cada jogador é normalmente representado por uma personagem e podem ainda existir outras controladas pelo sistema. A **personagem** do jogador costuma ser o principal foco do jogo e é nesta que estão centradas a maioria das ações possíveis de realizar. Existe também a possibilidade de o jogador controlar não uma, mas várias personagens, podendo interagir com estas, ou atribuir-lhes tarefas, como em simuladores de sistemas. Uma personagem, ou outros elementos animados do jogo são geralmente representados por aquilo a que se chama um *sprite*. Os *sprites* são conjuntos de imagens *bitmap* que representam os vários estados de, por exemplo, uma personagem. Estes são utilizados para criar animações que dão a sensação de movimento, alternando rapidamente entre imagens.

2.3.2 Classificação de jogos educativos

Na secção anterior foi apresentada a taxonomia utilizada para classificar os jogos comerciais, no entanto esta não é tão apropriada para classificar os jogos educativos. Isto porque estes jogos encaixam-se principalmente nas categorias de quebra-cabeças, simulação e estratégia, as quais não são muito explícitas em relação a outros fatores importantes no ensino. Com o objetivo de resolver esse problema, os autores do estudo realizado em [WOMS09] definiram uma taxonomia para a classificação de jogos educativos, que categoriza os jogos segundo os seguintes critérios:

- **Interação entre jogadores** – modos de jogador que o jogo disponibiliza;
- **Fantasia e Habilidades** – tipo de fantasia usada no jogo;
- **Conceito ou Ideia do Jogo** – género, conceito ou ideia base do jogo.

Na Figura 2.4 é apresentada a taxonomia como uma hierarquia com três níveis. Cada nível diz respeito aos critérios apresentados acima pela ordem que foram citados.

Reduzindo a taxonomia na Figura 2.4 ao terceiro nível da hierarquia é possível identificar dez categorias de jogos únicas:

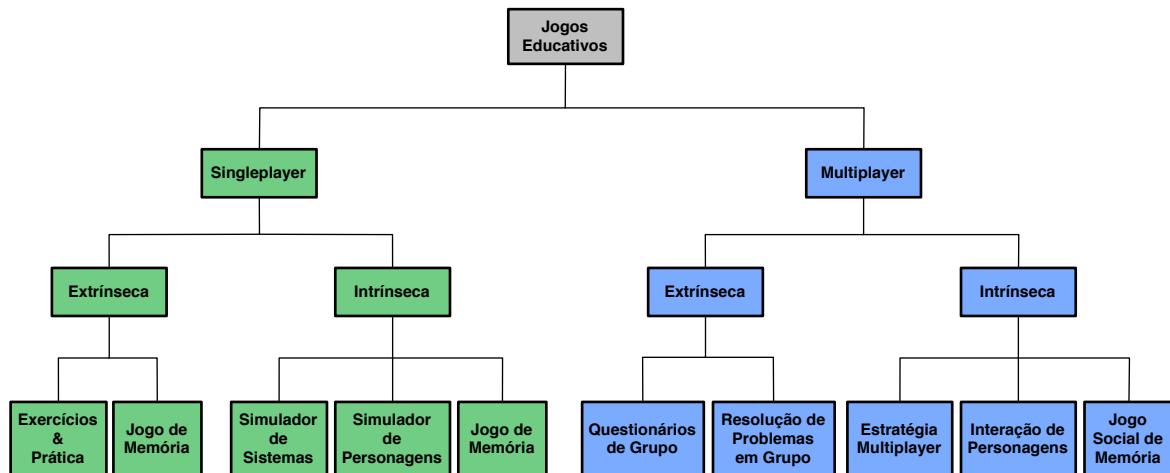


Figura 2.4: Taxonomia de categorização dos jogos educativos (adaptada de [WOMS09]).

- **Exercícios & Prática** – o jogador responde a perguntas por reconhecimento visual, ou inserção de texto e é testado quanto à lembrança e reconhecimento de factos. Exemplos: Quest4K [CRE12] e Questionaut [Ama12];
- **Jogo Extrínseco de Memória** – apresenta desafios que envolvem alguma forma de raciocínio não relacionado com a fantasia do jogo. Exemplos: Minesweeper [Emm12] e Bloxorz [Min12];
- **Simulador de Sistemas** – simulação de um sistema complexo, constituído por muitas partes e regras diferentes. Tipicamente num simulador deste tipo o jogador controla mais do que uma personagem ou aspeto do jogo em simultâneo. Exemplos: Civilization [Tak12], CyberCiege [Riv12], SimCity [Ele12c] e SimSE [Emi12];
- **Simulador de Personagens** – simulação do ponto de vista de uma personagem virtual. O jogador controla apenas uma, ou um número muito limitado de personagens e não tem qualquer controlo sobre o ambiente em redor destas. Exemplos: NanoQuest [For12a], Tomb Raider¹⁴ e Mass Effect¹⁵ (os dois últimos não são educativos);
- **Jogo Intrínseco de Memória** – apresenta desafios que envolvem alguma forma de raciocínio relacionado com a fantasia do jogo. Exemplos: Incredibots [Ele12b];
- **Questionários de Grupo** – jogos compostos por questões de vários tipos onde os jogadores necessitam de interagir uns com os outros, dentro ou fora do jogo, de forma a avançar. Exemplos: Buzz!: The Schools Quiz [Rel12] e Scene it? [Scr12];
- **Resolução de Problemas em Grupo** – jogos de memória em que certos parâmetros afetam o jogo de outros jogadores, sem que a fantasia do jogo afete o desafio em si. Exemplos: The Swedish Armed Forces [For12b];

¹⁴<http://www.tombraider.com/> consultado em 15-04-2012.

¹⁵<http://masseffect.bioware.com/> consultado em 16-05-2012.

- **Estratégia Multiplayer** – Jogos *multiplayer* onde as ações de diferentes jogadores afetam o jogo de todos e em que a fantasia do jogo é relevante. Exemplos: Civilization [Tak12] e Command & Conquer¹⁶ (não é educativo);
- **Interação de Personagens** – simulador de personagens com modo *multiplayer*, onde diferentes jogadores controlam diferentes personagens. Exemplos: Wii Sports [Nin12] e Star Wars The Old Republic¹⁷ (não é educativo);
- **Jogo Social de Memória** – jogos que envolvem a discussão de ideias entre vários jogadores para superar determinados desafios e em que a fantasia do jogo oferece *feedback* na resolução desses desafios. Exemplos: Tractor Multiplication [Arc12].

Na nossa opinião e como é também referido em [WOMS09], esta taxonomia é bastante mais apropriada para classificar os jogos educativos, no entanto tem um problema em relação à dos jogos comerciais. Um jogo que suporte ambos os modos *singleplayer* e *multiplayer*, segundo esta taxonomia poderá ser categorizado de duas formas diferentes. Por exemplo, um jogo intrínseco de memória pode ter também um modo *multiplayer* que lhe atribui propriedades de um jogo social de memória. Nestes casos, ou se dá relevância a um dos modos de jogador para classificar o jogo, ou então são-lhe atribuídas duas classificações distintas relativamente ao seu modo de jogador.

2.3.3 Desenho e desenvolvimento

O desenho de um jogo educativo deve satisfazer um equilíbrio entre diversão e conteúdo educacional [Don07, MGBMO⁺08], no entanto, isso nem sempre acontece. O grande desafio está em incluir o conteúdo educacional num jogo, sem que este se torne menos interessante ou divertido que um jogo tradicional. Contudo, a forma mais eficaz de abordar essa situação nem sempre é óbvia e na maioria dos casos depende do conteúdo educacional a introduzir no jogo.

O primeiro passo para iniciar o desenho de um jogo educativo consiste em definir o género do jogo. Para o fazer é necessário considerar alguns aspetos como, as idades, preferências e experiência com tecnologias do público alvo, e o tipo de informação didática a colocar no jogo. Feita essa escolha, é depois possível escolher um motor de jogo apropriado, de preferência destinado ao desenvolvimento de jogos educativos [MGBMO⁺08, ZSG⁺09].

Após a escolha do motor de jogo é necessário começar a pensar na implementação do jogo, para isso devem ser tidos em conta os critérios apresentados abaixo [WOMS09]:

- **Controlo instrucional variável** – o nível de dificuldade deve ajustar-se às capacidades do jogador, ou este deverá poder ajustá-la, funcionando assim como uma

¹⁶<http://www.commandandconquer.com/> consultado em 15-04-2012.

¹⁷<http://www.swtor.com/> consultado em 22-03-2012.

forma de adaptabilidade, ou personalização, respetivamente. No caso de jogos *multiplayer*, as capacidades dos jogadores devem ser comparadas de forma a encontrar jogadores ao mesmo nível;

- **Presença de suporte instrucional** – sempre que um jogador é incapaz de resolver um problema, devem ser-lhe fornecidas sugestões ou deverá haver informação adicional que lhe pode ser disponibilizada. Em jogos *multiplayer* devem ainda ser fornecidos sistemas de *chat*, ou áreas de discussão onde os jogadores possam entreajudar-se, comunicando de forma assíncrona ou síncrona;
- **Necessidade de suporte externo** – para garantir que o uso de um jogo num contexto educacional é bem sucedido devem considerar-se também fatores externos, como, a disponibilidade de *hardware*, o acompanhamento pessoal e disponibilidade de orientadores;
- **Design convidativo** – os jogadores devem sentir que estão a jogar um jogo, em vez de trabalhar com um programa. Um *design* convidativo pode fazer com que os jogadores se sintam mais motivados. Em jogos *multiplayer* é ainda importante que os jogadores possam personalizar as suas personagens de forma única para que se possam diferenciar das dos outros, funcionando estas como indicadores de *awareness* informal;
- **Estratégia de prática** – os jogadores devem conseguir praticar sem afetar os seus pontos, ou estado de forma negativa. Em jogos *multiplayer* deve ser disponibilizado, sempre que se justifique, um modo *singleplayer*;
- **Princípio sólidos de ensino** – Alguns exemplos desses princípios são abstrações motivadoras do programa teórico, aprendizagem colaborativa, ou o uso de princípios psicológicos de cognição, como, a repetição e a aprendizagem incremental;
- **Credibilidade dos conceitos** – a teoria ou as habilidades devem ser abstraídas de uma forma que garanta a integridade da informação a ensinar ao jogador;
- **Conceito de jogo inspirador** – o conceito do jogo deve incentivar o jogador a investir nele o seu tempo. A única forma de garantir esta característica é realizando estudos empíricos para determinar o nível de satisfação dos jogadores.

Os oito critérios listados acima são os mais relevantes, mas não são os únicos. Além destes, os jogos educativos, como qualquer outro ambiente de aprendizagem, devem ainda englobar registos de atividades [MGBMO⁺08], e é também importante ter em conta as plataformas de *hardware* a utilizar, o *software* usado na implementação, o lado artístico do jogo e a duração do jogo. Este último critério tem sido um dos problemas mais apontados nos jogos comerciais [PMP10] e que deve ser evitado nos educativos. Para isso, estes não devem ser muito extensos, devem permitir um número limitado de repetições e devem de alguma forma impedir o estudante de continuar a partir de uma determinada

parte por um período limitado de tempo. Algumas sugestões para determinar quando este deve parar de jogar são, por exemplo, estabelecer um limite máximo de horas por dia, criar áreas que necessitam da permissão do professor para continuar, ou a necessidade de recorrer a materiais de estudo para conseguir avançar.

Ao definir a forma como todas essas características serão abordadas e implementadas é também necessário ter em consideração como vão estar estruturados os vários cenários. Cada cenário terá ainda as suas próprias características, como cores ou imagens de fundo, animações, eventos, personagens, objetos, indicadores de *awareness*, registos de atividades, entre outras. Todas essas propriedades já dependem do género do jogo e do material didático que este contém. A título de exemplo são apresentados na lista abaixo alguns jogos educativos e respetivas características a nível de jogabilidade, história e desenho:

Questionaut [Ama12] – jogo de exercícios & prática dirigido ao ensino primário que abrange temas relacionados com línguas, matemática e ciências no geral. Os vários cenários deste são compostos por animações que necessitam da interação do jogador para este poder avançar na história. Essa interação consiste em clicar em zonas chave que desencadeiam um conjunto de reações sobre um ou mais elementos presentes no cenário. Após terminar a sequência de animações num cenário, antes de passar ao próximo, o jogador é defrontado com várias perguntas didáticas a que deverá responder corretamente em sequência para poder continuar;

NanoQuest [For12a] – simulador de personagens dirigido ao ensino básico e secundário relacionado com nanotecnologia e outros conceitos de ciências. Ao longo do jogo é apresentada uma história ao jogador sobre a forma de imagens em sequência que podem envolver conversas entre personagens e/ou a apresentação de determinados tipos de acontecimentos. Para avançar na história, o jogador deverá utilizar a sua personagem e os seus conhecimentos de ciências para resolver os vários desafios que lhe são apresentados;

Buzz!: The Schools Quiz [Rel12] – jogo de questionários de grupo dirigido ao ensino primário e básico relacionado com línguas, matemática e ciências. O jogo é composto por várias rondas de perguntas com diferentes regras e desafios, um pouco como em concursos televisivos. Ao longo dessas rondas os jogadores devem responder a perguntas de escolha múltipla relacionadas com diferentes áreas de ensino. Por cada resposta certa ganham pontos, os quais permitem determinar no final quem é o vencedor, sendo portanto um jogo de competição;

CyberCiege [Riv12] – simulador de sistemas dirigido ao ensino de engenharia informática que cobre aspetos relacionados com computadores e segurança de redes. Neste o jogador controla um grupo de trabalhadores (personagens) responsáveis pela manutenção e segurança de uma rede de computadores. O seu objetivo é utilizar dinheiro virtual para obter servidores, sistemas operativos, aplicações, dispositivos

de rede, entre outros, que lhe permitam aumentar a segurança da rede e a produtividade dos trabalhadores. Ao longo das várias fases do jogo, o principal objetivo é obter o material necessário para proteger a rede de um determinado tipo de ataques e para isso o jogador deverá utilizar os seus conhecimentos de forma a realizar as melhores escolhas.

Os jogos descritos acima são exemplos de diferentes géneros de jogos educativos dirigidos a diferentes idades e ao ensino de diferentes matérias. Em cada um deles são utilizadas formas de apresentação da história e dos conteúdos que mais se adequam ao seu público alvo e aos conteúdos que pretendem ensinar. Todos eles com exceção do Buzz! são disponibilizados na internet como aplicações independentes, no entanto não são muito divulgados nem podem ser facilmente integrados nos LMS.

2.3.4 Ambientes de desenvolvimento de jogos

Os ambientes de desenvolvimento de jogos são concebidos principalmente para facilitar a criação de jogos, fornecendo formas mais práticas de implementar certos elementos, como personagens e animações, e de visualizar o resultado gráfico. Estes ambientes utilizam como base um motor de jogo, sendo portanto destinados ao desenvolvimento de um determinado género de jogo. Exemplos de sistemas deste género são o Unity¹⁸, o Hero Engine¹⁹ e o BigWorld²⁰, todos eles muito utilizados no desenvolvimento de jogos comerciais. Esses sistemas permitem a criação de jogos complexos, necessitam de conhecimentos de programação e geralmente não são fáceis de utilizar.

Para o desenvolvimento de jogos educativos por parte dos professores, os dois últimos aspetos referidos acima devem ser evitados, já que os professores, na sua maioria, não têm conhecimentos de programação, nem experiência na utilização de sistemas deste género. Assim sendo é necessário fornecer-lhes um ambiente de desenvolvimento de jogos que seja simples e prático de usar, e que não necessite de conhecimentos de programação [TMGMOFM09]. Idealmente, esse ambiente deverá ainda ter a possibilidade de integração com os LMS existentes, de forma a permitir a utilização de outros conteúdos e/ou atividades didáticas disponibilizadas por estes. Para além disso, é também importante que esse ambiente ajude os professores no processo de desenvolvimento e os ensine a desenvolver jogos melhores e mais complexos à medida que ganham experiência. Na atualidade já existem sistemas que satisfazem algumas dessas características, tanto para o desenvolvimento de programas que envolvem elementos gráficos, como para desenvolvimento de jogos educativos. No entanto, não são muito conhecidos nem divulgados e na maioria dos casos não permitem a integração com os LMS existentes, nem facilitam o processo de instalação dos jogos, tornando assim a tarefa mais difícil para o utilizador.

¹⁸<http://unity3d.com/> consultado em 26-05-2012.

¹⁹<http://www.heroengine.com/> consultado em 26-05-2012.

²⁰<http://www.bigworldtech.com/> consultado em 26-05-2012

Na lista abaixo são apresentados dois exemplos de sistemas de desenvolvimento de jogos para o ensino. Note que o primeiro é mais focado no ensino da programação.

Alice [[Car12](#)] – ambiente desenvolvido para ensinar a programação de objetos 3D, sem a necessidade de utilizar código, sendo grande parte da interação com o programa feita por botões, menus e *drag & drop* de objetos do jogo. Apesar de o seu objetivo não ser o desenvolvimento de jogos, engloba vários elementos característicos desse tipo de ambientes. A interface de utilizador está dividida em 5 áreas diferentes [[KCC⁺02](#)]: (i) **janela de cenário** - onde o utilizador pode visualizar o aspeto gráfico de um cenário; (ii) **árvore de objetos** - organizada hierarquicamente por objetos e pequenas partes destes; (iii) **área de detalhes dos objetos** - que oferece informação sobre o objeto selecionado, como cor de fundo, opacidade, nome, eventos, entre outros; (iv) **área de animações** - que permite criar animações com um, ou mais objetos; (v) e **área de comportamentos** - que permite associar eventos a um objeto. Para além dessas áreas o sistema oferece ainda a possibilidade de guardar os mundos criados e personagens com animações e eventos associados, de forma a facilitar a reutilização e partilha. No entanto, não é disponibilizada qualquer forma de integração com os [LMS](#);

<e-Adventure> [[e-U12](#)] – ambiente desenvolvido para a criação de jogos educativos de aventuras *point & click* por parte dos professores. Permite a integração destes no [Moodle](#), [Blackboard](#) e [Sakai](#), no entanto não possibilita a utilização de conteúdos e atividades disponibilizados por esses [LMS](#). Utiliza [Extensible Markup Language \(XML\)](#) para representação dos metadados de um jogo, como exercícios, cenários e personagens, e estes são escritos em Java, não sendo no entanto necessários conhecimentos de programação para usar o sistema [[MGBMO⁺08](#)]. A interface de utilizador está dividida em três áreas principais: (i) **barra de menus no topo da janela** - idêntica à que se encontra na maioria das aplicações Desktop; (ii) **área lateral de opções e objetos** - disponibiliza botões para voltar atrás ou à frente e pesquisar, e permite a visualização dos vários grupos de informações e objetos de um jogo por secções (como metadados, cenários, personagens e itens); (iii) **janela de edição** - que permite a visualização e edição dos elementos gráficos de um cenário, ou da informação de outros objetos do jogo, como eventos, animações, cores e imagens.

Dos ambientes apresentados acima, o **<e-Adventure>** é o que mais se aproxima dos nossos objetivos. No entanto, segundo o que verificámos, esse ambiente apesar de possibilitar o desenvolvimento de jogos educativos não é fácil de utilizar e está organizado de uma forma confusa.

2.3.5 Integração dos jogos nos LMS

Para a integração de conteúdos didáticos nos [LMS](#), como os jogos educativos, podem ser desenvolvidos *plugins* para esses [LMS](#), como por exemplo o [Moodle](#), ou utilizados

standards como o Sharable Content Object Reference Model (SCORM) [Rus12] e o IMS Learning Tools Interoperability (IMS LTI) [IMS12].

O desenvolvimento de *plugins* para os LMS é uma tarefa mais simples que as outras referidas, pois facilita a integração com todos os conteúdos oferecidos pelo sistema e não é necessário estabelecer a comunicação entre dois sistemas diferentes. No entanto, esta solução não é tão apropriada quando o objetivo é permitir a integração com os LMS no geral, pois seria necessário desenvolver um *plugin* diferente para cada um, mas que oferece as mesmas funcionalidades. Além disso, para a sua instalação no LMS é geralmente necessária a intervenção do administrador do sistema.

O SCORM, atualmente na versão 2004, é um *standard* usado para integrar pacotes com conteúdos textuais, multimédia, ou outros num LMS [Mec08, TMGMOFM09, BTMGM09]. Este *standard* permite que se estabeleça um canal de comunicação entre os conteúdos e o LMS que possibilita a recolha de informação sobre o desempenho dos estudantes. No caso do Moodle, por exemplo, esses conteúdos são integrados no sistema na forma de atividades SCORM²¹ que podem ser colocadas pelos professores dentro de um curso. Um exemplo da utilização deste *standard* para integrar jogos com o Moodle pode ser encontrado em [eLe12a]. Um outro exemplo são também os jogos desenvolvidos pela ferramenta <e-Adventure> [e-U12], apresentada anteriormente.

Por outro lado, o IMS LTI, atualmente na versão 1.1.1, é um *standard* que permite a um LMS aceder a aplicações ou atividades disponibilizadas por um sistema externo que também suporte o *standard* [IMS12]. Este *standard* foi desenvolvido principalmente por existir a necessidade de introduzir novos conteúdos nos LMS que disponibilizam formas mais sofisticadas de avaliação, colaboração, interação, entre outros. Atualmente já pode ser considerado como um substituto do SCORM, pois oferece um maior número de possibilidades no que diz respeito à interação e colaboração entre estudantes. No caso do Moodle, por exemplo, para a criação de uma atividade disponibilizada por um sistema externo que utilize este *standard* deverá ser utilizada a opção de “Ferramenta Externa”²². Uma demonstração de como este *standard* pode ser utilizado no Moodle em diferentes contextos é apresentada em [Gav12].

Na Figura 2.5 é apresentado como se realiza a comunicação entre dois sistemas que suportam o IMS LTI. Segundo este *standard*, um sistema externo que fornece atividades que podem ser acedidas por outros sistemas é um **fornecedor**. Uma atividade existente no fornecedor é vista como uma **ferramenta**. Esta ferramenta deverá ser acessível a partir de um URL e deverá ter uma chave e um segredo associados. Esses dados deverão ser fornecidos ao **consumidor** da ferramenta que é neste caso o LMS. Este último irá criar um componente executável que é responsável por executar o procurador da ferramenta de cada vez que esta for iniciada. O procurador, por sua vez, é responsável por enviar para o fornecedor todos os dados necessários para identificar e iniciar a ferramenta. Ao receber esses dados, o fornecedor utiliza-os para identificar e devolver a ferramenta ao utilizador.

²¹http://docs.moodle.org/22/en/SCORM_module consultado em 29-06-2012.

²²http://docs.moodle.org/22/en/External_tool consultado em 29-06-2012.

A versão atual deste *standard* permite ainda que seja enviado de volta para o [LMS](#), em qualquer momento após a atividade ter sido iniciada, informação sobre o desempenho dos estudantes, algo que não acontecia nas versões anteriores. Esta informação é um valor numérico entre 0 e 1 utilizado para atribuir uma nota ao estudante na forma de percentagem.



Figura 2.5: Arquitetura de dois sistemas que comunicam entre si através do IMS LTI (adaptada de [IMS12, Cha10]).

Todo esse processo de comunicação entre os dois sistemas é realizado de forma segura utilizando o método de assinatura de mensagens do protocolo OAuth [Mic13]. Este protocolo utiliza a chave e o segredo da ferramenta para assinar as mensagens enviadas nas duas direções. A chave faz sempre parte das mensagens enviadas e esta deverá ser utilizada para obter o segredo da ferramenta. Este segredo é utilizado antes de enviar uma mensagem para encriptar o seu conteúdo gerando assim a assinatura. Quando a mensagem é recebida o segredo é utilizado para gerar de novo a assinatura com base no conteúdo da mensagem e o resultado obtido deverá ser igual à assinatura recebida, garantindo assim a validade do pedido.

Ambos os *standards* aqui apresentados oferecem uma solução para a integração dos jogos nos [LMS](#), mas para isso é necessário que estes sistemas os suportem. O Moodle, como foi mencionado, já suporta ambos. Por essa razão, e por o [IMS LTI](#) ser um *standard* mais recente e mais sofisticado no que diz respeito ao suporte de meios de colaboração e interação, este foi o escolhido para utilizar nesta dissertação. Deste modo, os jogos a desenvolver serão externos aos [LMS](#) e poderão ser distribuídos através de um simples URL, não havendo a necessidade de fornecer aos utilizadores ficheiros que necessitam de ser colocados nesses sistemas.

2.4 Conclusões

Face ao estudo apresentado neste capítulo, pode-se concluir que as atividades colaborativas são essenciais para o processo de aprendizagem, por incentivarem a participação e o espírito de entreajuda entre os participantes nesse processo. Além disso, se estas atividades forem divertidas, como é o caso dos jogos, contribuem para um aumento da motivação e da aquisição de competências.

Apesar das vantagens envolvidas, estas atividades (jogos) não são ainda muito utilizadas no ensino, em parte porque não são fáceis de desenvolver, nem de distribuir pelos estudantes. Os [LMS](#), como foi visto, oferecem a melhor solução para as integrar e distribuir, contudo, por vezes, não fornecem meios de colaboração ou interação muito ricos que o possibilitem.

Além disso, os jogos, como qualquer outra atividade colaborativa ou ambiente de aprendizagem, devem fornecer aos jogadores meios de interação e indicadores de *awareness*, e aos professores registo de atividades que permitam avaliar o desempenho desses jogadores.

Por essas razões, consideramos que é necessário implementar uma plataforma que facilite o desenvolvimento de jogos educativos por parte de professores, e a distribuição desses jogos nos [LMS](#). Para a sua distribuição será, como referido anteriormente, utilizado o *standard IMS LTI*. Além disso, deverá ser possível integrar essa plataforma num [LMS](#), neste caso o [Moodle](#), permitindo assim que esta seja utilizada a partir de um ambiente de aprendizagem, sem perder o contexto deste último.

Em resumo, pela pesquisa bibliográfica realizada pode-se também concluir que o tópico central desta dissertação é atual, e que são muitos os trabalhos que referem que os jogos são fundamentais para o ensino em geral.

3

Plataforma Epik

O [Epik](#) é uma plataforma Web que permite a gestão de conteúdos didáticos e o desenvolvimento, distribuição e execução de jogos educativos. O principal objetivo desta plataforma é facilitar o desenvolvimento de atividades educativas que são mais divertidas e que fomentam a participação através de mecanismos de colaboração e interação. Os conteúdos didáticos que podem ser geridos no [Epik](#) podem ser criados na própria plataforma ou importados de um sistema [Moodle](#). Todo o processo de desenvolvimento de jogos no [Epik](#) é feito a partir de uma ambiente gráfico que não requer quaisquer conhecimentos de programação. Para além disso, o [Epik](#) pode ser integrado num sistema [Moodle](#), permitindo que este seja usado no contexto de um ambiente de ensino.

Os jogos desenvolvidos no [Epik](#) podem adaptar-se a qualquer área de ensino, e podem ser usados como atividades educativas dentro ou fora de aulas. Estes jogos oferecem diferentes mecanismos de colaboração, percepção do meio envolvente e permitem reutilizar conteúdos didáticos já existentes nos [LMS](#). A distribuição destes jogos pode ser feita a partir de um [LMS](#), ou diretamente na plataforma [Epik](#), sendo esta última sempre responsável pela execução dos mesmos.

Para a integração do [Epik](#) no [Moodle](#) foram criadas várias extensões para esse [LMS](#), entre elas um novo tema intitulado Cubic. Essas extensões têm como objetivo permitir a integração dos dois sistemas e fornecer novas formas de interação e colaboração ainda não disponibilizadas por este [LMS](#).

Ao longo deste capítulo serão descritas as várias características desta plataforma, os tipos de jogos que nesta podem ser desenvolvidos, o processo de desenvolvimento, distribuição e execução de jogos e também a integração desta plataforma com o [Moodle](#).

3.1 Características gerais

A plataforma [Epik](#) tem como principais objetivos facilitar o desenvolvimento e distribuição de atividades educativas na forma de jogos, que fomentam a participação e colaboração entre estudantes no processo de aprendizagem. Para o desenho e implementação desta plataforma foi tido como base o estudo realizado sobre os ambientes de desenvolvimento de jogos existentes e os ambientes de aprendizagem colaborativa. O [Epik](#) distingue-se dos ambientes existentes pelas seguintes características:

- **Facilita o desenvolvimento de jogos educativos** – a interface gráfica do [Epik](#) fornece aos seus utilizadores todas as ferramentas necessárias para o processo de desenvolvimento de jogos educativos, não sendo necessários conhecimentos de programação para ser utilizada. Para além disso, se o utilizador não se sentir pronto para criar um jogo de raiz pode sempre escolher utilizar um dos *templates* fornecidos que definem já grande parte das propriedades de um jogo;
- **Facilita a distribuição de jogos como atividades educativas** – os jogos criados com o [Epik](#) podem ser facilmente distribuídos a partir de um [LMS](#), ou na própria plataforma. Além disso, toda a informação sobre o desempenho dos jogadores é guardada no [Epik](#) e é possível enviá-la para o [LMS](#) na forma de uma nota com valor percentual;
- **Fornece meios de colaboração e interação entre jogadores** – os jogos do [Epik](#) fornecem formas de colaboração e interação entre os jogadores que contribuem para o aumento da participação e sentido de entreajuda. Além disso, incluem também várias formas de percepção do estado do jogo que contribuem para um aumento da colaboração e competição entre os jogadores;
- **Permite a reutilização de conteúdos didáticos** – os jogos desenvolvidos no [Epik](#) podem reutilizar conteúdos didáticos previamente criados num [LMS](#). Desta forma, o processo de desenvolvimento de um jogo educativo poderá ser uma tarefa menos dispendiosa.

Para a concretização de cada uma dessas características foi necessário fornecer aos utilizadores do [Epik](#) formas de: (i) criação e gestão de jogos educativos em desenvolvimento e em formato executável; (ii) criação e gestão de conteúdos didáticos que podem ser utilizados nos jogos; (iii) visualização e gestão dos dados gerados no final de um jogo acerca do desempenho dos jogadores. Além disso o [Epik](#) é o responsável pela execução desses jogos o que facilita a sua distribuição noutros sistemas como os [LMS](#).

3.2 Tipos de jogos

Os jogos do [Epik](#) podem ser *singleplayer* ou *multiplayer*. Em ambos os casos o seu objetivo é fornecer aos jogadores uma forma mais divertida e interativa de aprender. No

caso dos jogos *multiplayer* o objetivo é também fornecer formas de colaboração e alguma competição entre os jogadores de uma equipa.

Independentemente do modo de jogo, os jogos do [Epik](#) são sempre compostos por vários cenários ligados entre si. Alguns destes cenários são comuns a todos os jogos e outros são os criados pelos utilizadores durante o processo de desenvolvimento. Os cenários comuns a todos os jogos são os seguintes:

- **Cenário de inicio** – é composto pelo nome e ícone do jogo e uma lista de botões que permitem ao jogador mudar para o cenário de sala de espera, para o cenário de instruções, ou sair do jogo, respetivamente;
- **Sala de espera** – neste cenário o jogador deve introduzir o seu nome e escolher o seu avatar para poder iniciar o jogo. No caso de um jogo *multiplayer*, após submeter a sua informação, o jogador irá ficar aqui à espera dos jogadores necessários para formar a equipa;
- **Cenário de instruções** – neste cenário o jogador pode ler as instruções do jogo e escolher voltar ao cenário de inicio, ou sair do jogo;
- **Cenário de fim de jogo** – ou **cenário de derrota**, apresentado sempre que um jogo é terminado sem sucesso, ou seja, não existem jogadores suficientes para continuar;
- **Cenário de classificações** – ou **cenário de vitória**, apresentado sempre que um jogo é terminado com sucesso. Neste cenário são apresentados os três jogadores (caso *singleplayer*), ou as três equipas (caso *multiplayer*) que obtiveram a melhor posição no jogo até à data corrente e a posição em que o jogador ou equipa corrente ficou. Estas classificações têm como objetivo aumentar a competição entre jogadores ou equipas e motivá-los a jogar novamente.

Entre o cenário de sala de espera e os cenários de vitória ou derrota são apresentados os cenários criados pelo utilizador, os quais constituem o corpo do jogo. Deste modo, o fluxo geral de um jogo do [Epik](#) pode ser representado como na Figura 3.1.

Os cenários destes jogos, sejam eles os comuns ou os de corpo, contêm sempre o **logótipo** do [Epik](#). Além disso, os cenários de corpo e de vitória contêm também o **painel de jogadores**. Neste painel são apresentados vários tipos de pontuações e informações acerca dos jogadores que servem como uma forma de percepção do meio envolvente (estado do jogo), e de fomentar a competição e a colaboração.

Uma instância de um jogo do [Epik](#) na qual participam um ou mais jogadores é denominada de **sessão** de jogo. Durante uma sessão toda a informação e ações dos jogadores, incluindo a colaboração entre estes, é gerida pelo [Epik](#). No final da sessão, se o jogo foi terminado com sucesso, a informação gerada acerca do desempenho dos jogadores é armazenada e fica disponível para consulta na forma de um **registo de sessão**.

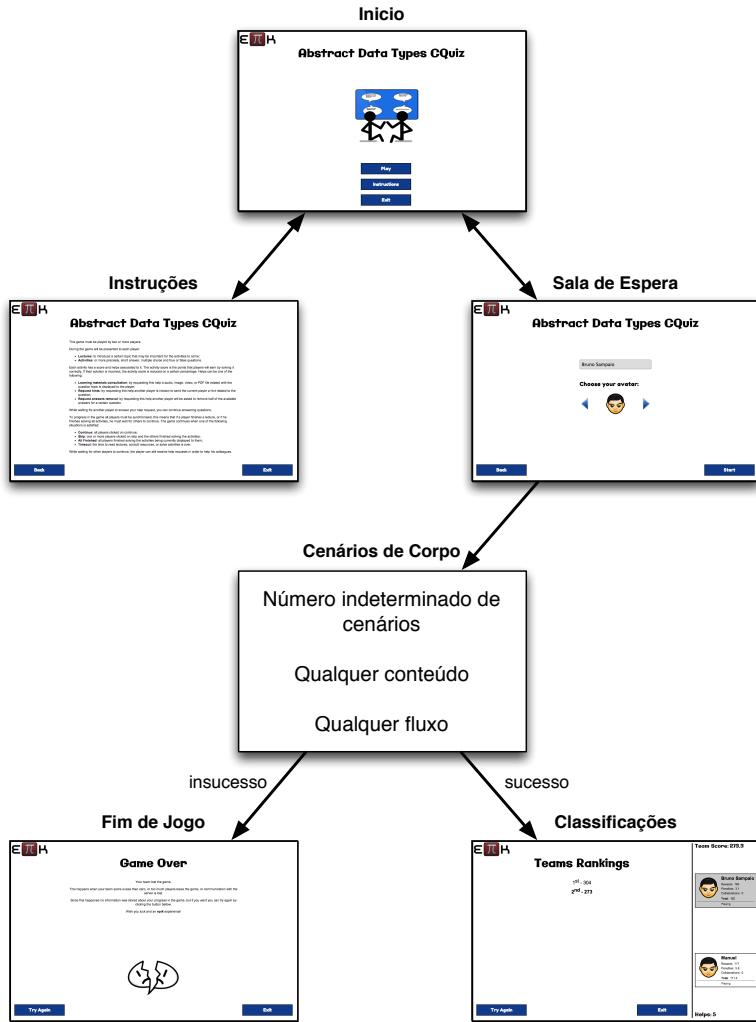


Figura 3.1: Fluxo geral de um jogo do Epik.

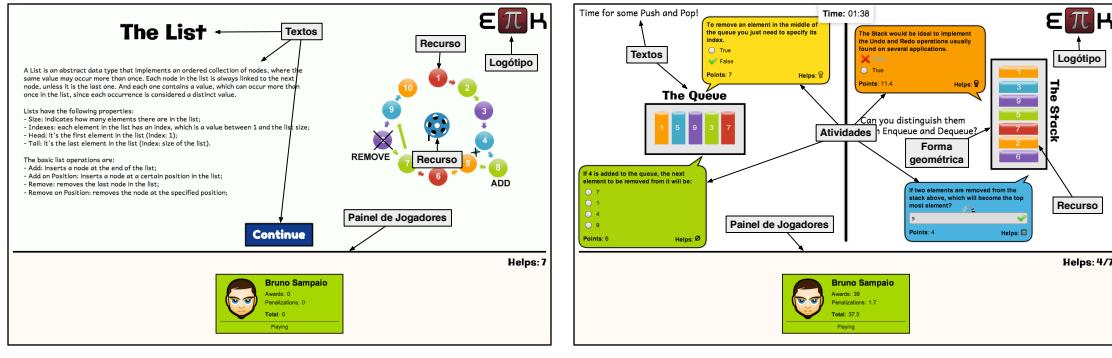
3.2.1 Jogos de Questionário

Atualmente o [Epik](#) permite apenas o desenvolvimento de jogos que se encaixam nas categorias de Exercícios & Prática e Questionários de Grupo. Dentro de cada uma dessas categorias definimos os jogos de Questionário Individual e Questionário Colaborativo, respetivamente. A principal diferença entre ambos é que os Questionários Individuais são *singleplayer* e os Questionários Colaborativos são *multiplayer*, mas as regras de ambos são bastante idênticas. Além disso, ambos são jogos a duas dimensões baseados no estilo de interação *point & click*. Estes jogos são baseados no conceito de questionário já existente no [Moodle](#) e outros [LMS](#) com algumas componentes extra.

Os cenários que compõem o corpo de um jogo de Questionário podem ser de um dos seguintes tipos:

- **Cenários de conceitos** – podem conter texto, formas geométricas e recursos. Utilizados para apresentar um tópico ou conceitos aos jogadores. Um exemplo deste tipo de cenários é apresentado na Figura 3.2a;

- **Cenários de atividades** – podem conter atividades e também texto, formas geométricas ou recursos. Utilizados para apresentar atividades que os jogadores devem realizar. Um exemplo deste tipo de cenários é apresentado na Figura 3.2b.



(a) Cenário de conceitos

(b) Cenário de atividades

Figura 3.2: Tipos de cenários dos jogos de Questionário do Epik.

Os tipos de **recursos** didáticos que podem ser colocados nos cenários destes jogos podem ser ficheiros de áudio, imagem, vídeo, ou PDF. Todos eles, com exceção das imagens, são representados nos cenários por um *sprite* que necessita de ser clicado pelos jogadores para visualizar ou reproduzir o seu conteúdo. Estes recursos podem ser utilizados como uma forma de apresentar um conceito aos jogadores ou como uma forma de ajuda numa dada atividade.

Por outro lado, as **atividades** podem ser vistas como os desafios ou exercícios a serem realizados pelos jogadores. Essas atividades podem ser dos seguintes tipos:

- **Questões** – representam uma questão de escolha múltipla, resposta curta, ou verdadeiro ou falso. As questões de escolha múltipla podem conter entre duas a cinco hipóteses;
- **Grupos de Questões** – representam um conjunto de questões do qual é escolhida uma aleatoriamente para ser apresentada aos jogadores. Isto permite a criação de jogos com conteúdos variáveis.

Ambos os tipos de atividades podem ter vários tipos de ajuda e pontuações associadas que servem como uma forma de fomentar a colaboração e a competição entre os jogadores, respetivamente.

Durante um jogo, cada jogador tem sempre à sua disposição um número limitado de ajudas que podem ser utilizadas quando estes estão em dúvida sobre a resposta a uma pergunta. No contexto de um cenário de atividades pode também existir um valor limitado de ajudas que é inferior ao número total de ajudas no jogo. De notar que o número total de ajudas num jogo é constante, pois não existe forma de ganhar mais ajudas. Estas últimas são utilizadas pelos jogadores com o objetivo de relembrar/entender o que é

pedido numa dada pergunta sem que lhe seja dada a resposta. Os tipos de ajudas que podem ser associados a uma atividade são:

- **Consulta de materiais didáticos** – é apresentado ao jogador um ficheiro de áudio, imagem, vídeo, ou PDF que contem a informação necessária associada ao tópico da atividade;
- **Pedido de dicas** – é apresentada uma dica ao jogador de uma lista de dicas (frases curtas) associadas à atividade.
- **Remoção de respostas incorretas** – são removidas metade das hipóteses de resposta disponíveis (também conhecida por 50/50). Esta ajuda só está disponível para perguntas de escolha múltipla.

Destes três tipos de ajudas apenas a primeira funciona de igual modo para ambos os tipos de questionários. Assim, quando o jogador pede essa ajuda é escolhido um recurso associado à atividade para lhe ser apresentado.

Para além das ajudas, como referido acima, podem ainda haver dois tipos de pontuações associadas a uma pergunta: os pontos de recompensa e a percentagem de penalização. Os **pontos de recompensa** são os pontos que o jogador recebe por responder acertadamente a uma pergunta. Enquanto que a **percentagem de penalização** é o valor percentual em que os pontos de recompensa da atividade devem ser reduzidos por cada tentativa de resposta incorreta, ou por cada pedido de ajuda sem sucesso. Este último caso ocorre quando o jogador pede uma ajuda e a sua primeira tentativa de resposta após receber a ajuda está incorreta. Em situações desse género a percentagem de penalização é aplicada a dobrar sobre os pontos de recompensa da atividade.

De modo a possibilitar a criação de jogos que variam com base nas ações e conhecimentos dos jogadores, a sequência de cenários que lhes é apresentada ao longo de um jogo está definida com base em certas situações que podem ocorrer em cada cenário, tais como:

- **Continuar** – todos os jogadores clicaram em continuar;
- **Término de atividades** – todos os jogadores terminaram as atividades num cenário;
- **Saltar** – a soma do número de jogadores que clicaram em saltar (sendo que pelo menos um clicou) e dos que terminaram as atividades é igual ao total de jogadores;
- **Tempo limite** – o tempo limite de permanência no cenário, independentemente do seu tipo, foi atingido. Note que cada cenário no jogo pode ter um tempo limite de permanência dos jogadores no cenário;

Quando uma dessas situações ocorre, caso esta tenha sido definida para o cenário corrente, o jogo muda para outro cenário. No caso dos cenários de atividades não é permitida a situação de continuar, e no caso de cenários de conceitos não é permitida a

situação de saltar, nem faria sentido permitir a de término de atividades. Independentemente do caminho que o jogo seguir este irá sempre terminar num dos cenários de vitória ou derrota, como apresentado na Figura 3.1.

No decorrer de um jogo, o painel de jogadores, como apresentado na Figura 3.3, disponibiliza sempre informação acerca de cada jogador e do número de ajudas que o jogador corrente ainda pode utilizar. Para cada jogador são apresentados o seu nome, avatar, pontuações e estado. As pontuações de um jogador estão divididas em: (i) **pontos de recompensa** – total de pontos recebidos por cada atividade resolvida; (ii) **pontos perdidos** – total de pontos perdidos por cada tentativa de resposta incorreta ou pedido de ajuda sem sucesso; (iii) **total de pontos** – pontuação total do jogador, ou seja, pontos de recompensa menos os pontos perdidos. Por outro lado, o estado dos jogadores está dependente das ações que estes realizaram no cenário corrente. Os estados possíveis em ambos os tipos de jogos de Questionário são: (i) está a jogar; (ii) clicou em continuar; (iii) clicou em saltar; (iv) terminou todas as atividades; (v) atingiu o tempo limite. No que diz respeito às ajudas, como é possível ver na figura, estas são apresentadas no formato k/n , onde k é o número de ajudas que ainda podem ser usadas no cenário corrente e n é o número total de ajudas do jogador. No caso de o número de ajudas do jogador já ser inferior ao limite imposto pelo cenário corrente, ou se não existir limite, apenas o n é apresentado.

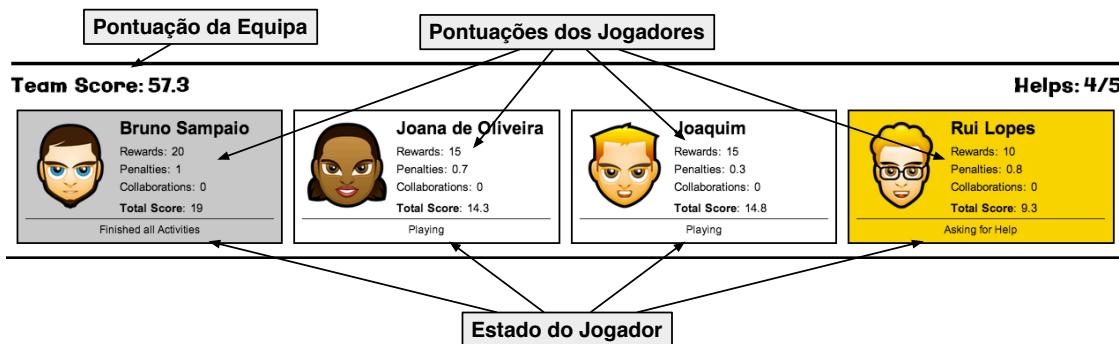


Figura 3.3: Painel de jogadores dos jogos de Questionário do Epik.

Todos os aspetos referidos nesta secção funcionam de igual modo para ambos os tipos de jogos de Questionário, com exceção das ajudas de dicas e remoção de respostas. Além disso, os Questionários Colaborativos incluem ainda outros tipos de pontuações referentes à colaboração, que são apresentadas no painel de jogadores.

3.2.1.1 Questionários Individuais

Os Questionário Individuais, apesar de não proporcionarem formas de colaboração, são um tipo de questionário mais interativo, comparativamente aos disponibilizados por outras ferramentas, como por exemplo, os LMS. Isto porque disponibilizam várias informações relativas ao desempenho do jogador ao longo do jogo, contêm tipos de conteúdos mais variados e disponibilizam ajudas.

Estes questionários não introduzem regras adicionais às referidas anteriormente, a

única diferença está no funcionamento das ajudas de dicas e remoção de respostas e na forma como o jogo decorre. Neste caso quando o jogador pede uma dessas ajudas, o sistema irá escolher aleatoriamente uma dica, ou as respostas a remover, respetivamente. Além disso, o jogador nunca tem que esperar por outros jogadores para iniciar o jogo, ou para realizar determinadas ações dentro de um cenário.

3.2.1.2 Questionários Colaborativos

Os Questionários Colaborativos introduzem novas regras para além das referidos anteriormente, como por exemplo, os bónus, os tempos de resposta, a colaboração entre jogadores, entre outros. Ao longo de um jogo deste tipo, os vários jogadores estão sempre no mesmo cenário e só podem prosseguir quando todos estiverem prontos (coordenação). O número de jogadores necessário para jogar estes jogos está sempre entre 2 e 4. Definimos este intervalo porque em certos contextos (aulas) poderá ser difícil reunir mais jogadores acima desses valores, o que iria dificultar a realização deste tipo de atividades.

No painel de jogadores destes jogos, além da informação referida anteriormente, são apresentados dois tipos de informação adicionais (Figura 3.3): os pontos totais de colaboração por jogador e a pontuação total da equipa. Os **pontos de colaboração** são os pontos recebidos por um jogador colaborar com outro jogador, os quais são adicionados à pontuação total do primeiro. Enquanto que a **pontuação total da equipa** é a soma das pontuações totais de todos os jogadores. Além dessa informação, existem dois novos estados possíveis para cada jogador: a pedir ajuda e a ajudar.

Para receberem pontos de colaboração os jogadores necessitam de receber um pedido de ajuda de outro jogador. Esses pedidos podem ocorrer em qualquer momento do jogo e são enviados sempre que um jogador pede as ajudas de dicas ou remoção de respostas, não podendo ser realizados dois em simultâneo. Neste caso em vez de ser o sistema a escolher os valores a enviar, deve ser outro jogador a fazê-lo, mas para isso é necessário que este já tenha respondido corretamente à pergunta. Quando um pedido é recebido, dentro da lista de valores apresentados, o jogador deverá escolher aqueles que na sua opinião vão ajudar o seu colega a perceber a pergunta (comunicação e cooperação). Ao responder a um pedido de ajuda a informação selecionada pelo jogador é enviada e apresentada ao jogador que pediu ajuda. Além desta é também apresentado um tempo limite para responder à pergunta, o qual irá definir se o jogador que ajudou irá ou não receber um bónus e se o seu colega irá ou não ser penalizado por ter pedido ajuda. Para que o bónus seja atribuído e não haja penalização é necessário que o jogador que pediu ajuda resolva a atividade dentro desse tempo sem falhar nenhuma tentativa.

Os **bónus** são pontos adicionais que os jogadores podem receber por realizarem determinadas ações. Estes pontos existem no contexto de cenários de atividades e podem variar consoante o cenário. Os tipos de bónus existentes são:

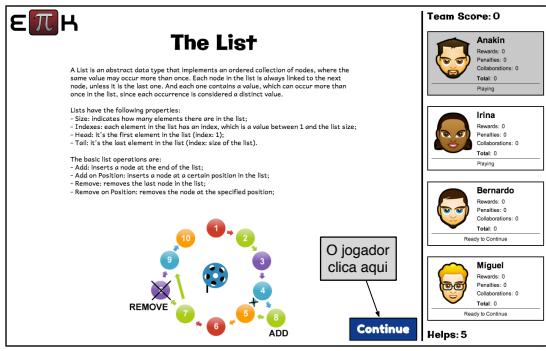
- **Primeiro a terminar** – o primeiro jogador a terminar todas as atividades de um cenário recebe este bónus. Os pontos recebidos são adicionados aos seus pontos de

recompensa.

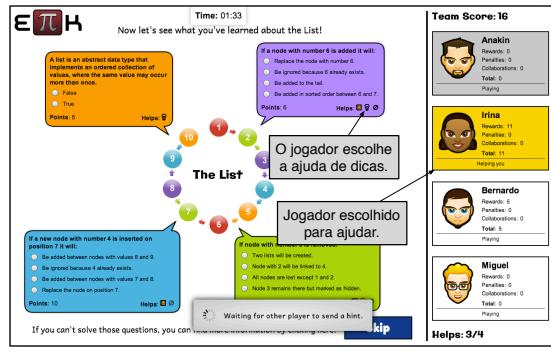
- **Colaboração** – um jogador recebe este bónus por ajudar outro com sucesso. Os pontos recebidos são adicionados aos pontos de colaboração do jogador.

As componentes de comunicação, cooperação e coordenação entre os jogadores identificadas anteriormente satisfazem os requisitos de uma atividade colaborativa [Koc08]. Apesar de o tipo de comunicação disponibilizado por estes jogos não ser livre, os jogadores têm sempre várias opções de onde podem escolher o que querem enviar (escolha de dicas ou respostas a remover).

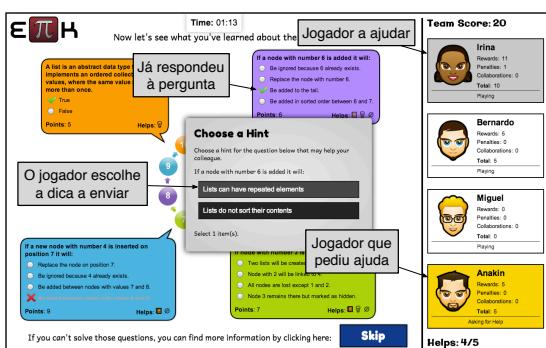
A título de exemplo é apresentada na Figura 3.4 uma situação de colaboração entre dois jogadores. Inicialmente todos os jogadores se encontram no cenário apresentado na Figura 3.4a. O primeiro jogador apresentado no painel de jogadores (no lado direito do cenário) corresponde sempre ao jogador que está a visualizar esse ecrã de jogo, neste caso é o Anakin. Nesta situação os dois últimos jogadores já clicaram em continuar e estão à espera dos outros dois. Quando todos clicarem em continuar passam ao cenário na Figura 3.4b.



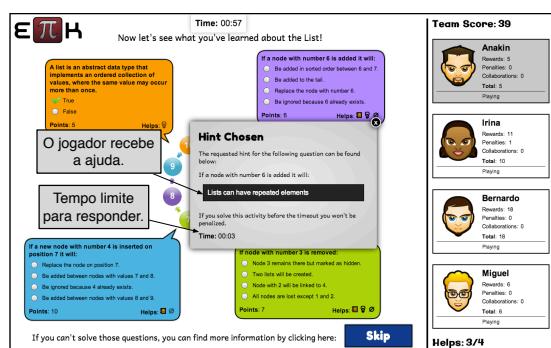
(a) Apresentação de um conceito.



(b) O Anakin pede ajuda de dicas.



(c) A Irina ajuda o Anakin.



(d) A ajuda é apresentada ao Anakin.

Figura 3.4: Situação de colaboração num jogo de Questionário Colaborativo.

Neste cenário são apresentadas quatro perguntas relacionadas com a imagem ao centro. Se o Anakin selecionar a ajuda de dicas da pergunta superior direita (com fundo roxo), será selecionado outro jogador para o ajudar. Neste caso foi selecionada a Irina,

por ser a única que já tinha respondido corretamente à pergunta, estando agora identificada a amarelo no ecrã do Anakin (Figura 3.4b).

Na Figura 3.4c é apresentado o ecrã da Irina após receber o pedido de ajuda. Neste caso também o Anakin aparece identificado a amarelo de forma a que ela saiba quem está a pedir ajuda. Ao centro é apresentada uma janela com duas dicas das quais ela pode escolher apenas uma. Depois de escolher uma dica, esta é enviada e apresentada no ecrã do Anakin, como é possível ver na Figura 3.4d. Por baixo da dica é também apresentado o tempo limite para responder.

3.3 Desenvolvimento de jogos

Um jogo em desenvolvimento no [Epik](#) denota-se por projeto. É através destes projetos que o utilizador define as várias características de um jogo, tais como, os cenários e respetivos conteúdos, o fluxo do jogo, pontuações, ajudas e a informação a guardar no final do jogo (registos de sessões).

Para o desenvolvimento de jogos no [Epik](#) foram criadas duas áreas que fornecem aos utilizadores todas as ferramentas de que necessitam. Como o objetivo era fornecer uma forma fácil e prática de criar jogos, a interface de utilizador dessas áreas utiliza uma nomenclatura, estrutura e organização familiar aos utilizadores. Essas áreas são denominadas de:

- **Dashboard** – apresentada na Figura 3.5, é a área onde os utilizadores podem consultar, criar, editar, ou remover todas as suas atividades, recursos, projetos e jogos. Esta área é composta por secções específicas a cada um desses tipos de conteúdos, as quais são acessíveis a partir do painel de navegação apresentado na figura (estando neste caso selecionada a secção de atividades);
- **Ambiente de Desenvolvimento** – apresentado na Figura 3.6, é a área onde os utilizadores podem criar os cenários de corpo que compõem o jogo e parametrizar as respetivas pontuações, ajudas, bónus, fluxo e outros. Este ambiente, como apresentado na figura, está dividido nas seguintes secções:
 - **Barra de Ferramentas** – apresentada no topo da figura 3.6. Disponibiliza várias ferramentas de criação, edição, inserção, consulta e remoção. Algumas das mais importantes são as de criação de atividades, recursos, projetos e cenários, inserção de caixas de texto, formas geométricas, recursos e atividades, e geração de um jogo. Na Figura 3.7 as várias ferramentas são apresentadas em maior detalhe. De notar que algumas ferramentas podem ou não estar disponíveis dependendo do elemento que está atualmente selecionado no painel de navegação deste ambiente. Por exemplo, se o utilizador se encontrar num cenário de conceitos as ferramentas de inserção de atividades não estarão disponíveis;

Painel de Navegação

Lista de Atividades

Figura 3.5: Área de gestão de conteúdos do Epik com a secção de atividades selecionada.

Painel de Navegação

Área de Desenho

Painel de Propriedades

Figura 3.6: Ambiente de desenvolvimento de um jogo no Epik.

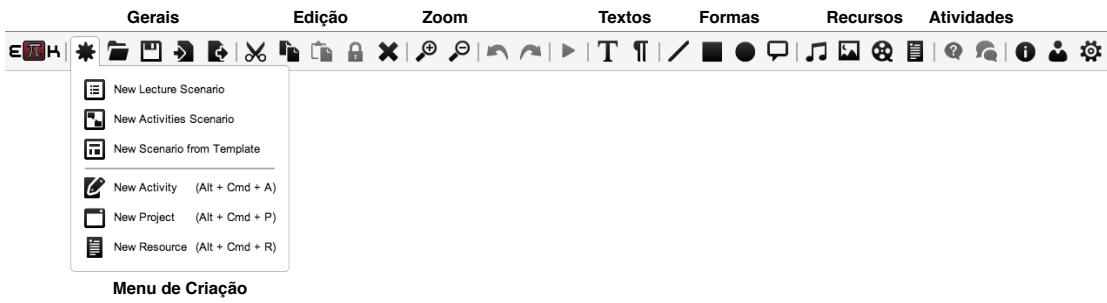


Figura 3.7: Barra de ferramentas do ambiente de desenvolvimento de jogos do Epik.

- **Painel de Navegação** – apresentado no lado esquerdo da figura 3.6. Permite ao utilizador gerir, organizar e alternar entre os vários elementos que compõem um jogo. Esses elementos, pela ordem ilustrada na figura 3.6, são as propriedades gerais, os cenários comuns e os cenários de corpo. Para cada um dos cenários de corpo são ainda listados os seus conteúdos organizados em pastas. Quando o utilizador clica num qualquer elemento neste painel, esse elemento é identificado ou apresentado na área de desenho e todas as suas propriedades parametrizáveis são apresentadas no painel de propriedades;
- **Área de Desenho** – apresentada no centro da figura 3.6. É a área em que o aspeto gráfico do jogo é apresentado. A partir desta o utilizador pode realizar várias operações como selecionar, mover e redimensionar os conteúdos do cenário atualmente selecionado no painel de navegação. No caso de estar uma propriedade selecionada, o conteúdo desta área fica vazio, contendo apenas o logótipo do Epik e o painel de jogadores;
- **Painel de Propriedades** – apresentado no lado direito da figura 3.6. Este painel apresenta os vários campos parametrizáveis do elemento atualmente selecionado no painel de navegação. Entre os campos específicos a cada tipo de elemento encontram-se geralmente neste painel campos para definir os estilos do elemento corrente como a sua posição, tamanhos e cores. No caso particular dos cenários comuns este painel apresenta apenas uma descrição visto que estes não podem ser costumizados.

O desenvolvimento de um jogo no Epik passa por várias fases que podem ser realizadas através das áreas apresentadas acima. Essas fases são a criação de atividades e recursos, criação de um projeto, parametrização das propriedades gerais de um jogo, criação dos cenários e seus conteúdos, e geração do jogo. Estas fases não têm que ser realizadas por esta ordem, podendo até algumas delas ser realizadas em simultâneo. Cada uma destas fases será apresentada em maior detalhe ao longo das próximas secções.

3.3.1 Criação de atividades e recursos

Esta fase do desenvolvimento é considerada à parte da de criação dos conteúdos dos cenários de um jogo, pois é necessário criar as atividades e recursos para que estes possam depois ser inseridos nos cenários. Optámos por esta abordagem, pois assim estes conteúdos são independentes dos jogos em que são usados, podendo ser reutilizados em vários sem ser necessário voltar a inseri-los no [Epik](#).

Durante o desenvolvimento de um jogo as atividades e recursos a utilizar podem ser criados a partir da sua respetiva secção na *dashboard* ou a partir da barra de ferramentas no ambiente de desenvolvimento. A criação destes a partir da *dashboard* poderá revelar-se especialmente útil para a criação de um jogo de raiz, pois é bastante mais fácil e rápido realizar operações sobre esses conteúdos nesta área. No entanto, a sua criação a partir do ambiente de desenvolvimento evita alternar entre as duas áreas sempre que é necessário criar uma nova atividade ou recurso. Além disso permite que o utilizador os crie à medida que são necessários se assim o preferir.

Para a criação de uma atividade ou recurso o utilizador pode inserir os dados diretamente no [Epik](#) ou pode importar esses dados do [Moodle](#). Neste último caso, os dados existentes no [Moodle](#) são copiados para o [Epik](#). Por essa razão, o conteúdo criado disponibilizará a opção de atualização que permitirá ao utilizador importar de novo os dados existentes no [Moodle](#) para substituir os já existentes no [Epik](#). Isto significa que o utilizador pode sempre alterar os seus conteúdos no [Moodle](#) sem afetar os dados correspondentes no [Epik](#), podendo no entanto atualiza-los neste último sem ter que voltar a criá-los. Esta funcionalidade permite aos utilizadores criar tipos de atividades educativas distintas que se adaptam a diferentes formas de aprendizagem reutilizando conteúdos. Por exemplo, um professor pode criar um questionário no [Moodle](#) e reutilizar as perguntas que o compõem em jogos do [Epik](#).

Independentemente do método utilizado para criar uma atividade, se esta for uma questão, será ainda possível associar-lhe dicas e recursos para serem utilizados nas ajudas de dicas e consulta presentes nos jogos, respetivamente. No caso de grupos de questões para os criar é necessário que as questões sejam criadas previamente. As questões que compõem um grupo de questões podem ser de qualquer tipo.

3.3.2 Criação de projetos

A criação de um projeto é necessária para que o utilizador possa realizar as restantes fases do processo de desenvolvimento de um jogo, além da de criação de atividades e recursos. Esta operação pode ser realizada através da secção de projetos na *dashboard*, ou através do menu de criação na barra de ferramentas da área de desenvolvimento. A criação de um projeto envolve, como apresentado na Figura 3.8, escolher o tipo de jogo (Questionário Individual ou Questionário Colaborativo) e o *template* de projeto. Para cada tipo de jogo existe sempre um ou mais *templates de projeto* que definem as propriedades gerais de um jogo. Além disso, estes templates podem também definir um ou mais cenários e

respetivos conteúdos, necessitando estes últimos de ser preenchidos. É ainda importante notar que depois de criar um projeto deixa de ser possível alterar o *template* escolhido.



Figura 3.8: Janela de escolha de *template* durante a criação de um projeto no Epik.

Depois de o projeto estar criado, este passa a estar disponível na secção de projetos da *dashboard* ou através da barra de ferramentas do ambiente de desenvolvimento. Ao abrir o projeto, o utilizador entra no ambiente de desenvolvimento de jogos a partir do qual poderá definir as propriedades e criar os vários cenários do seu jogo.

3.3.3 Parametrização das propriedades gerais

As **propriedades gerais** de um jogo associado a um projeto são o primeiro conjunto de itens apresentado no painel de navegação do ambiente de desenvolvimento (Figura 3.6). Estas definem vários aspetos do jogo que não estão relacionados com nenhum cenário em particular. Os valores destas propriedades são sempre definidos pelo *template* utilizado para criar o projeto, mas podem ser modificados se o utilizador assim o desejar. As propriedades gerais disponíveis para os jogos de Questionário são:

- **Logótipo** – quando selecionada são apresentados no painel de propriedades campos que permitem ao utilizador alterar a posição do logótipo do Epik no ecrã de jogo. Este pode ser posicionado em qualquer um dos cantos do ecrã;
- **Jogadores** – quando selecionada o utilizador pode alterar, através do painel de propriedades, o número de jogadores necessários para jogar o jogo, a posição do painel de jogadores no ecrã de jogo e várias propriedades de estilos. O número de jogadores só pode ser alterado para jogos *multiplayer*, sendo sempre necessário especificar um valor entre 2 e 4. O painel de jogadores pode ser posicionado verticalmente à esquerda ou à direita, ou horizontalmente em cima ou em baixo. Os campos de

estilos permitem ao utilizador alterar as cores de fundo do painel de jogadores, do jogador corrente, dos jogadores em estado normal e do jogador a colaborar;

- **Pontuações e Ajudas** – quando selecionada permite ao utilizador definir através do painel de propriedades os nomes de cada tipo de pontuação, o valor total de ajudas e quais os valores a registar no final do jogo. No caso das pontuações, os valores a registar são os pontos obtidos pelos jogadores num determinado tipo de pontuação, e relativamente às ajudas, são registadas as usadas por cada jogador e as que cada um forneceu a outros jogadores (apenas no caso de jogos *multiplayer*);
- **Sons** – quando selecionada permite ao utilizador ouvir e alterar, através do painel de propriedades, a música de fundo do jogo e os sons de resposta correta, resposta errada e pedido ou receção de ajuda. Estes sons são basicamente recursos do tipo áudio criados pelo utilizador. Se estes campos forem omitidos serão usados os sons do [Epik](#);

3.3.4 Criação dos cenários e seus conteúdos

Os cenários de um jogo podem ser criados a partir do menu de criação na barra de ferramentas do ambiente de desenvolvimento, apresentado anteriormente na Figura 3.7. Este menu disponibiliza três opções para a criação de cenários: cenário de conceitos vazio, cenário de atividades vazio e cenário baseado num *template*. Esta última permite ao utilizador escolher um **template de cenário**, os quais definem um cenário de um determinado tipo com vários conteúdos posicionados que necessitam de ser preenchidos.

Quando um cenário está selecionado, além dos campos para definir o seu nome e cor de fundo, são apresentados no painel de propriedades, os dos bónus e do fluxo de jogo. A forma como esses campos são apresentados ao utilizador pode ser vista na Figura 3.9a. Os bónus são apenas disponibilizados para os Questionários Colaborativos em cenários de atividades. Além do seu valor é possível definir se o valor total recebido por cada tipo de bónus no cenário corrente deve ser guardado nos registo de sessões do jogo. Em relação ao fluxo é apresentado um campo para definir se o jogo deve começar no cenário corrente e outro para cada tipo de salto permitido nesse mesmo cenário (continuar, saltar, término de atividades e tempo limite). A partir destes últimos o utilizador pode escolher qual o próximo cenário após uma dessas situações ocorrer. Dessa lista de cenários pode ser sempre escolhido qualquer cenário de corpo que não seja o primeiro nem o corrente. Além desses é ainda possível escolher o cenário de vitória para definir que após o cenário corrente o jogo deve terminar.

Os conteúdos de um cenário são apresentados no painel de navegação organizados nas pastas de textos, formas geométricas, recursos e atividades. Todos esses conteúdos, quando selecionados, disponibilizam campos para definir o seu nome, tamanho, posição, rotação, e tamanho e cor da borda. Além desses, os títulos, parágrafos, quadrados, círculos e balões disponibilizam também um campo para definir o seu conteúdo textual e

(a) Campos específicos e parametrizáveis de um cenário.

(b) Campos específicos e parametrizáveis de uma atividade.

Figura 3.9: Exemplos do conteúdo do painel de propriedades do ambiente de desenvolvimento do Epik.

tamanho, estilo e cor da *font* utilizada. Estes últimos mais as atividades permitem ainda definir a sua cor de fundo.

Relativamente aos recursos e atividades, caso o cenário tenha sido criado a partir de um *template*, o seu conteúdo estará inicialmente vazio sendo portanto disponibilizada uma opção para escolher um recurso ou atividade existente, respetivamente. Quando estes estão preenchidos disponibilizam opções de consulta, edição e atualização dos seus dados, permitindo que o utilizador consulte ou altere esses dados sem ter que voltar à *dashboard*.

As atividades disponibilizam também outros campos que permitem parametrizar as suas pontuações e ajudas. A forma como esses campos são apresentados ao utilizador no painel de propriedades pode ser vista na Figura 3.9b. Na área de pontuações o utilizador pode parametrizar os pontos de recompensa e percentagem de penalização da atividade. Além disso pode também definir se os pontos que o jogador recebeu por responder corretamente e o total de pontos que perdeu por cada tentativa errada devem ser guardados nos registos de sessões do jogo. Em relação às ajudas, o utilizador pode definir quais devem ser disponibilizadas pela atividade e no caso das de consulta e dicas pode ainda definir qual o recurso e quais as dicas que devem estar disponíveis, respetivamente. O último campo apresentado na figura diz respeito ao tempo limite para responder após receber uma ajuda, o qual só se encontra disponível para Questionários Colaborativos.

No caso de grupos de questões, a área de ajudas permite apenas definir para cada uma se esta deve ou não ser disponibilizada ao jogador. No entanto, mesmo que o utilizador

escolha todas, estas apenas serão apresentadas se a questão escolhida durante o jogo tiver os dados necessários associados. Por exemplo, caso todas as ajudas de um grupo de questões tenham sido selecionadas e se durante um jogo é escolhida uma questão que não tem recursos nem dicas associadas, as ajudas de consulta e dicas não serão apresentadas. Além disso se não for uma pergunta de escolha múltipla a ajuda de remoção também não é apresentada.

3.3.5 Geração do jogo

Quando o desenvolvimento de um jogo está terminado, ou quando o utilizador deseja testar o funcionamento do seu jogo, deve utilizar a opção de exportação de projeto para o formato de jogo. Esta opção é disponibilizada na secção de projetos na *dashboard* e também no menu de exportação na barra de ferramentas do ambiente de desenvolvimento.

Para a realização deste processo de exportação, o utilizador deverá definir primeiro vários campos como o nome, visibilidade e ícone do jogo. Este último não é obrigatório, pois no caso de ser omitido o [Epik](#) utiliza o seu próprio ícone de jogo. O nome e o ícone aqui definidos serão depois apresentado no cenário de inicio do jogo. Enquanto que o campo de visibilidade permite ao utilizador definir se a informação do jogo e respetivos registos de sessões são ou não acessíveis publicamente. De notar que independentemente do tipo de visibilidade escolhido, o jogo poderá sempre ser acedido e jogado, sem ser necessário que o jogador seja um utilizador registado no [Epik](#), bastando conhecer o URL para o jogo.

Depois de submetida essa informação o [Epik](#) irá tratar de validar os dados do jogo com base nas regras do tipo de jogo escolhido aquando da criação do projeto. Se esse processo de validação for bem sucedido o jogo é criado e fica disponível na secção de jogos na *dashboard*. Caso contrário é apresentada ao utilizador uma lista dos erros encontrados no projeto e que necessitam de ser resolvidos para que possa ser gerado um jogo a partir deste. Algumas das regras mais relevantes e avaliadas por este processo são:

- O jogo tem um início – deve ser sempre especificado o primeiro cenário de corpo do jogo;
- O jogo tem um fim – deve existir sempre pelo menos um cenário com um salto para o cenário de classificações;
- O jogo não tem ciclos – no fluxo de cenários do jogo não podem existir ciclos, ou seja, um cenário nunca pode voltar a outro que já tenha sido jogado;
- Não existem cenários isolados – existe pelo menos um caminho que vai desde o primeiro cenário a cada um dos cenários de corpo;
- Existe mais do que um cenário – um jogo deve ser sempre composto por pelo menos dois cenários;

- Não existem cenários vazios – todos os cenários devem sempre conter algum tipo de conteúdo;
- Existem atividades – um jogo deve ter pelo menos uma atividade e o mesmo se aplica aos cenários de atividades;
- Não existem elementos vazios – no caso de caixas de texto estas devem sempre conter texto e as atividades ou recursos criados a partir de um *template* devem estar sempre associados a uma atividade ou recurso do utilizador, respetivamente;
- Existem pelo menos duas dicas – as atividades com a ajuda de dicas devem ter pelo menos duas dicas selecionadas no contexto do projeto de onde pode ser escolhida uma.

A criação do jogo consiste em replicar todos os dados do projeto e dos recursos e atividades utilizados de forma a impedir que qualquer alteração que lhes seja feita posteriormente possa tornar o jogo inválido. Isto significa que os dados de um jogo são estáticos não podendo ser alterados pelo utilizador, com exceção dos dados fornecidos aquando da sua geração como o seu nome e ícone. Se os dados do projeto, recursos e atividades não fossem replicados e mais tarde uma atividade utilizada em um ou mais jogos fosse removida, esses jogos ficariam imediatamente inválidos. Isto não impede o utilizador de alterar um jogo criado, mas para isso é necessário alterar o projeto e voltar a gerar o jogo.

O jogo gerado é portanto uma réplica dos dados definidos no projeto mas armazenados num outro formato. Parte desses dados serão utilizados para criar a interface de utilizador do jogo e outros para gerir a lógica do jogo.

3.4 Distribuição de jogos

Os jogos do [Epik](#) podem ser distribuídos através da própria plataforma ou de um [LMS](#). A informação necessária para o fazer é disponibilizada aos utilizadores do [Epik](#) na página de perfil de um jogo acessível através da secção de jogos na *dashboard*. Esta página disponibiliza ao utilizador toda a informação e várias opções de gestão do jogo respetivo.

Para distribuir um jogo a partir do [Epik](#) o utilizador apenas necessita de fornecer o URL público do jogo às pessoas que ele pretende que o joguem. Os registos de sessões gerados a partir de um jogo distribuído desta forma serão armazenados no contexto do [Epik](#) como sessões públicas.

Por outro lado, para distribuir um jogo a partir de um [LMS](#) este deverá suportar o *standard IMS LTI*, o que é o caso do [Moodle](#). No geral para o fazer é sempre necessário fornecer o URL de arranque do jogo e respetiva chave e segredo. No caso do [Moodle](#) essa informação deve ser fornecida aquando da criação de uma atividade num curso utilizando a opção de “Ferramenta Externa” desse [LMS](#). Opcionalmente poderá ainda ser fornecido o URL para o ícone da atividade que corresponde ao ícone do jogo, entre outras

opções específicas do Moodle. Quando essa atividade é iniciada no LMS, informação sobre este último, o curso e o utilizador é enviada para o Epik. Esta informação será depois utilizada para encontrar uma sessão para esse jogador e para no final armazenar os registo da sessão no contexto do curso e do LMS. Além disso, no final do jogo é também enviado para o LMS um valor de 100% que serve apenas como uma forma de identificar que o jogador terminou a atividade.

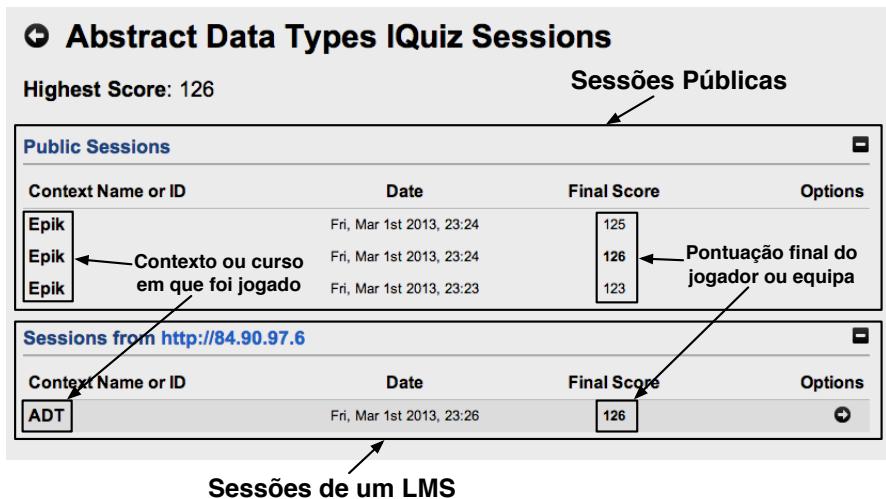


Figura 3.10: Área de consulta dos registo de sessões de um jogo.

Os registo de sessões de um jogo podem ser visualizados pelo utilizador através da área de registo, a qual é acessível a partir da página de perfil do jogo ou da secção de jogos na *dashboard*. Nesta área, como apresentado na Figura 3.10, os registo são organizados pelos vários contextos em que o jogo foi jogado, ou seja, no contexto do Epik ou de um dado LMS. Independentemente da forma como o jogo foi distribuído, o utilizador pode encontrar nesta área a pontuação do jogador ou equipa de cada sessão e qual foi o melhor jogador ou equipa até à data corrente. Para cada sessão pode ainda consultar os dados relativos a cada jogador, apresentados como na Figura 3.11. Entre esses dados podem ser encontrados os pontos recebidos para as categorias de recompensa, penalizações e colaborações, a quantidade de ajudas utilizadas e dadas, os bónus recebidos por cenário e os pontos recebidos e perdidos em cada atividade bem como o número de tentativas até acertar. No caso de o jogo ter sido distribuído a partir de um LMS, o utilizador terá ainda a opção de atribuir uma nota a cada jogador entre 0 e 100%, tendo em conta o seu desempenho na atividade. Ao definir esta nota o valor enviado para o LMS no final do jogo será substituído pelo novo valor.

3.5 Execução de jogos

Quando um jogo do Epik é acedido, seja através do seu URL público ou através de um LMS, é apresentada uma página ao jogador com o cenário de início. A partir deste cenário, como já foi referido, o jogador pode passar para o de sala de espera onde deve

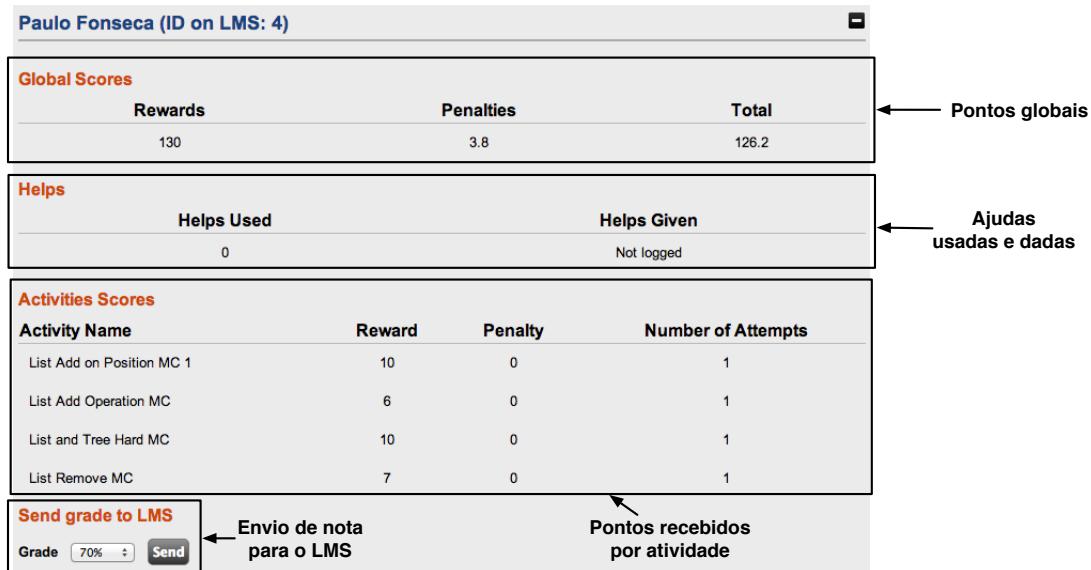


Figura 3.11: Área de consulta dos registos de uma sessão de um jogo.

colocar o seu nome e escolher o seu avatar para começar o jogo. Depois de enviar essa informação para o [Epik](#), este irá tratar de carregar os dados do jogo pretendido, associar esse jogador a uma sessão e enviar-lhe os dados das atividades e recursos do jogo. No caso de um jogo *singleplayer*, essa sessão será imediatamente iniciada. Enquanto que num jogo *multiplayer*, caso ainda não hajam jogadores suficientes, a sessão ficará em espera e o jogador terá que aguardar.

Quando a sessão é iniciada o jogo começa, ou seja, o primeiro cenário de corpo é apresentado a todos os jogadores. A partir deste momento os jogadores podem interagir com a interface de utilizador do jogo e essas interações na maioria dos casos serão processadas pelo [Epik](#), pois este é o responsável por gerir o fluxo do jogo, a validação de respostas, a colaboração entre jogadores e os dados a registar no final do jogo.

3.6 Integração no Moodle

De modo a integrar o [Epik](#) no [Moodle](#) foi necessário criar várias extensões para esse [LMS](#). Essas extensões permitem a um utilizador do [Moodle](#) aceder a aplicações Web sem perder o contexto do [LMS](#) e que essas mesmas aplicações importem conteúdos existentes nesse [LMS](#). Para este último caso foi necessário criar novos serviços para o [Moodle](#) que permitem aceder a informação de atividades do tipo questionário e recursos.

Entre essas extensões encontra-se um novo tema para o [Moodle](#) intitulado Cubic. Este tema, além de fornecer aos utilizadores novas formas de percepção e comunicação, reúne as funcionalidades das várias extensões criadas. Na Figura 3.12 é apresentada a página de um curso do [Moodle](#) com este tema e são identificados alguns dos componentes que o distinguem de outros temas existentes. Esses componentes são:

- **Barra de utilizador** – apresentada no topo da página. Esta barra está sempre presente independentemente da página em que o utilizador se encontra e disponibiliza vários menus que permitem ao utilizador aceder e consultar, de uma forma rápida, páginas institucionais e os seus cursos, aplicações, atividades, notificações, entre outros;
- **Instituições** – são um novo conceito que este tema introduz e que disponibiliza aos utilizadores uma forma de aceder a páginas Web externas ao Moodle sem abandonar o contexto deste último;
- **Aplicações** – são também um novo conceito que este tema introduz e que disponibiliza aos utilizadores uma forma de adicionar aplicações Web que suportam o IMS LTI, podendo depois ser acedidas através do Moodle;
- **Notificações** – os menus de notificações na barra de utilizador (novidades, eventos e mensagens) fornecem aos utilizadores formas de percepção do meio envolvente. Alguns exemplos de notificações são a publicação de uma mensagem num fórum de um curso, a aproximação da data para terminar uma atividade, ou a receção de uma mensagem privada, respetivamente;
- **Chat global** – fornece aos utilizadores do Moodle um novo meio de comunicação em tempo real que permite a professores e estudantes de um mesmo curso comunicarem entre si em qualquer momento independentemente da página em que se encontram.

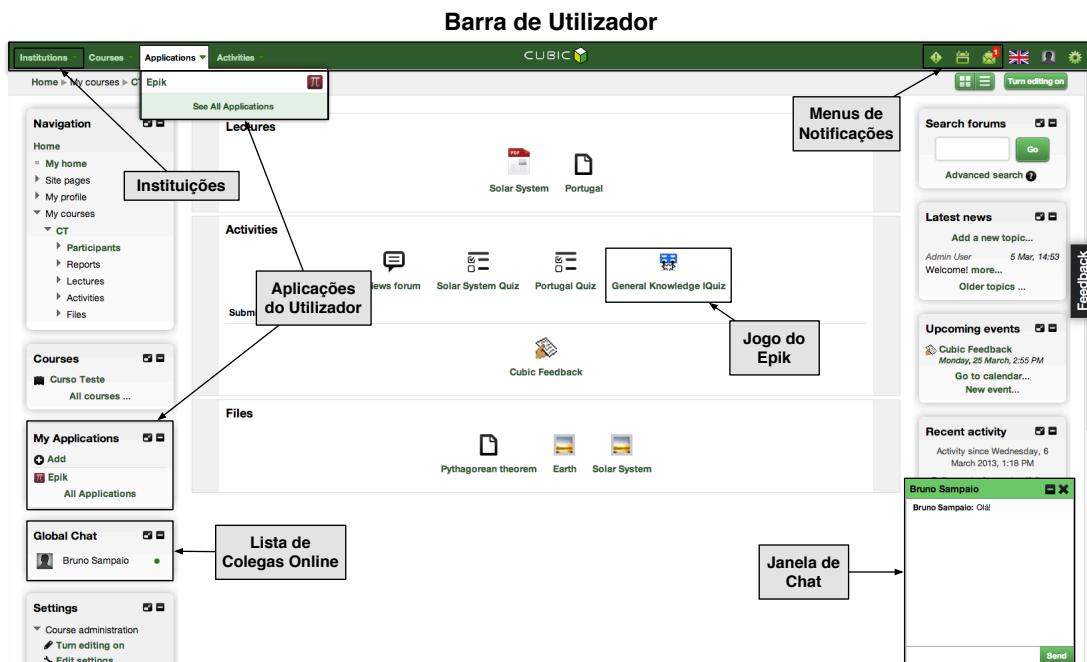
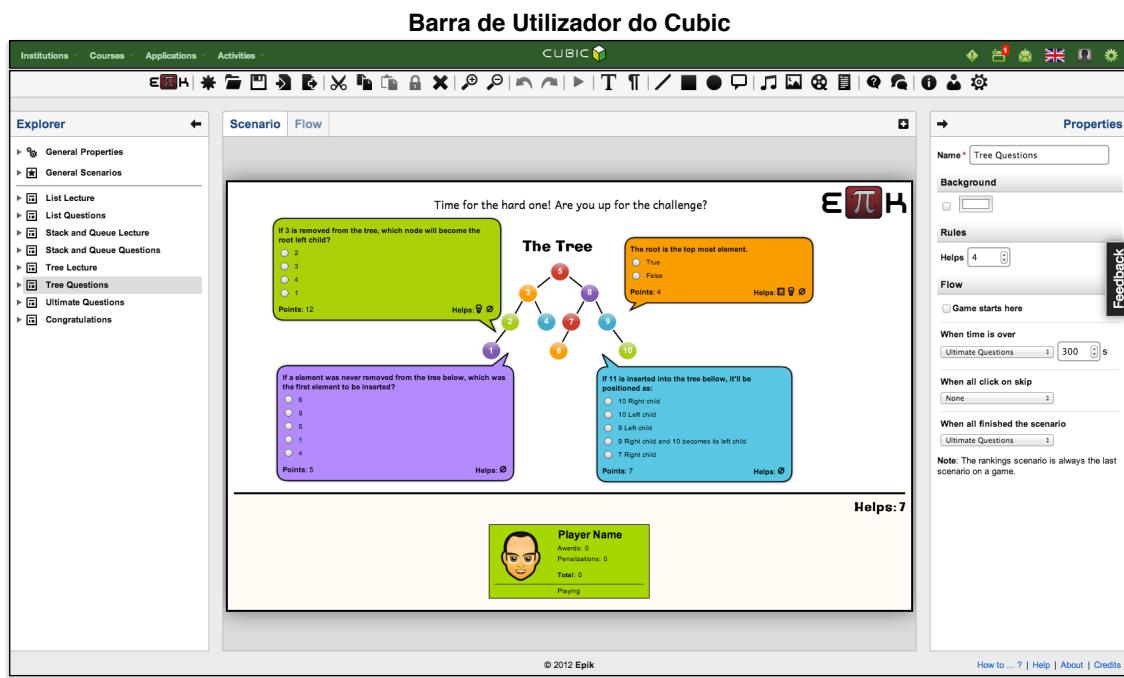


Figura 3.12: Interface de utilizador do tema Cubic.

Quando o utilizador abre uma página institucional, uma aplicação (como o [Epik](#)), ou uma atividade externa (como os jogos do [Epik](#)) a única parte da interface de utilizador que se mantém é a barra de utilizador. No restante espaço é apresentado o conteúdo da página da instituição, aplicação, ou atividade, respetivamente. Desta forma o utilizador consegue visualizar outra página Web sem perder a noção do que está a acontecer no [Moodle](#) graças à informação que lhe é disponibilizada pela barra de utilizador, podendo em qualquer momento voltar à página de um curso, atividade, ou outro. Na Figura 3.13 é apresentada uma situação em que o utilizador está a aceder ao [Epik](#) a partir do [Moodle](#).



Ambiente de Desenvolvimento do Epik

Figura 3.13: Integração do Epik no Moodle.

No caso das aplicações é ainda possível autenticar o utilizador automaticamente na aplicação quando esta é aberta. Por exemplo, se o [Epik](#) for aberto a partir do [Moodle](#) é possível reconhecer e autenticar automaticamente o utilizador correspondente no [Epik](#). Para isso quando o utilizador do [Moodle](#) adiciona o [Epik](#) às suas aplicações deverá fornecer os dados apresentados no seu perfil de utilizador do [Epik](#). Nesta página, tal como no perfil de um jogo, são apresentados o URL de autenticação e a chave e o segredo do utilizador.

Relativamente à importação de conteúdos, algumas das extensões criadas fornecem novos serviços para o [Moodle](#) que permitem a um utilizador aceder a atividades e recursos que lhe estão relacionados a partir de um sistema externo. No entanto, para que isso seja possível é necessário realizar algumas configurações a um sistema [Moodle](#), como ativar o protocolo [REpresentational State Transfer \(REST\)](#), criar um novo serviço externo com acesso às funções necessárias da [API](#) de serviços, criar um novo papel de utilizador com acesso a essa [API](#), e associar esse papel a todos os utilizadores que necessitam de

acesso externo. Estas configurações só podem ser realizadas por um administrador e requerem que, sempre que um novo utilizador é criado lhe seja atribuído o papel referido, caso contrário este não terá acesso. Com estas configurações finalizadas e com os novos serviços criados, um utilizador do [Epik](#) pode escolher importar conteúdos existentes no [Moodle](#). Para isso apenas necessitará de especificar no [Epik](#), o tipo de [LMS](#), o URL deste e o seu nome de utilizador e palavra-chave do [Moodle](#). Se o utilizador assim o desejar poderá ainda guardar no [Epik](#) o tipo e o URL do seu [LMS](#) através da sua página de perfil de utilizador, deixando assim de ter que fornecer estes dados quando pretende importar um conteúdo do seu [LMS](#).

3.7 Conclusões

A plataforma [Epik](#) foi desenvolvida tirando partido das novas tecnologias e permite a criação de atividades educativas em forma de jogos, os quais têm as características essenciais a um ambiente de aprendizagem. Esta plataforma, como foi visto, suporta atualmente jogos de Questionário Individual e Questionário Colaborativo. Estes jogos podem ser desenvolvidos e executados através da própria plataforma o que facilita a sua distribuição. O desenvolvimento destes jogos a partir do ambiente gráfico de desenvolvimento do [Epik](#) pode considerar-se uma tarefa simples por não serem necessários conhecimentos de programação para a realizar. Além disso, os jogos do [Epik](#) permitem registar informação relativa ao desempenho dos jogadores que pode ser relevante para a sua avaliação. Esses jogos são compostos principalmente por diferentes tipos de conteúdos didáticos existentes na plataforma, mas que podem ter sido importados de um sistema [Moodle](#). Para a realização desse processo de importação e para permitir a integração desta plataforma nesse [LMS](#) foram ainda desenvolvidas várias extensões para este último. Algumas dessas extensões, como foi visto, satisfazem também algumas das características atualmente consideradas como necessárias nos ambientes de aprendizagem.

Em suma, o [Epik](#) é uma plataforma que permite resolver alguns dos problemas relacionados com a utilização de jogos no ensino, permitindo que estes sejam desenvolvidos e distribuídos de uma forma simples e fácil.

4

Implementação

A plataforma [Epik](#) é composta por dois servidores, um responsável pelo desenvolvimento de jogos e gestão de conteúdos, e outro pela execução de jogos. Para a implementação destes servidores foram utilizadas diferentes tecnologias e diferentes estilos de arquitetura. Além desses servidores foram também desenvolvidos diferentes tipos de pacotes e *plugins* para o [Moodle](#) que permitem a integração do [Epik](#) nesse [LMS](#).

O servidor de desenvolvimento e gestão do [Epik](#) mantém toda a informação da plataforma e dos seus utilizadores. Este servidor processa todos os pedidos de consulta, criação, edição e remoção realizados pelos utilizadores sobre as suas atividades, recursos, projetos e jogos. Além disso, este é também o responsável por estabelecer a comunicação com um [LMS](#) e por processar os pedidos enviados por estes últimos.

Por outro lado, o servidor de execução de jogos do [Epik](#) é o responsável por gerir todas as sessões de jogos em espera e a decorrer, a interação e colaboração entre os jogadores, e os dados a guardar no final do jogo. Ao longo de um jogo, é este servidor que determina quando ocorre uma mudança de cenário, quando é que um jogador responde acertadamente a uma questão, quando é que se estabelece um processo de colaboração entre dois jogadores, entre outros.

Ao longo deste capítulo serão apresentadas as arquiteturas de ambos os servidores, o funcionamento do servidor de jogos e os diferentes tipos de extensões criadas para o [Moodle](#).

4.1 Servidor de gestão e desenvolvimento

O servidor de gestão e desenvolvimento do [Epik](#) tem como principal função devolver páginas Web por cada pedido realizado por um cliente (*browser*) utilizando o protocolo

Hypertext Transfer Protocol (HTTP). Este protocolo permite que um cliente, ao realizar um pedido, estabeleça um canal de comunicação com um servidor. Esse canal será utilizado por este último para enviar a página Web e respectivos conteúdos (como imagens, estilos e *scripts*) de volta para o cliente. No final dessa comunicação este canal é terminado.

É também responsabilidade deste servidor o tratamento de pedidos **IMS LTI** enviados por um **LMS**. Esses pedidos podem ser enviados com a informação do utilizador a autenticar, ou com a informação do jogo a jogar. Para além disso, é este servidor que realiza os pedidos à **API** de serviços do **Moodle**, aquando da importação de conteúdos.

Para a implementação deste servidor foi utilizado o CakePHP 2.2 [Cak13], uma *framework* escrita em PHP 5 [The13b] estruturada e implementada com base no padrão arquitetural **Model-View-Controller (MVC)** [KP88]. Este padrão divide a aplicação em três componentes: modelos de dados (*Model*), controlo do processamento e execução (*Controller*) e visualização dos dados (*View*). Com base nesse padrão, o CakePHP impõe várias convenções no que diz respeito à organização de ficheiros e estruturação do código e das bases de dados que tornam o processo de desenvolvimento mais simples e rápido. Além disso, disponibiliza também vários pacotes de lógica que facilitam o desenvolvimento de funcionalidades como a autenticação de utilizadores, internacionalização, validação de pedidos, segurança do sistema, e leitura e escrita de dados existentes numa base de dados ou no sistema de ficheiros.

A arquitetura deste servidor é apresentada na Figura 4.1 como uma arquitetura de três camadas. A camada de apresentação é a responsável por gerar os conteúdos a serem enviados para o cliente (*browser*). Esses conteúdos são geralmente criados com base em dados processados na camada de lógica. Esta última processa todos os pedidos realizados por um cliente (*browser* ou **LMS**), incluindo os que necessitam de informação existente num **LMS**. Na maioria dos casos esta camada necessita ainda de dados contidos numa base de dados ou no sistema de ficheiros. Esses dados são acessíveis através da camada de dados.

De notar, que o *browser* apresentado na Figura 4.1 refere-se principalmente ao Google Chrome¹, dado que este foi o utilizado para a realização de testes durante o desenvolvimento desta plataforma. Apesar disso, tentámos de um modo geral garantir o seu funcionamento nas versões mais recentes dos *browsers* mais utilizados, contudo isso não se verificou possível ao nível de algumas funcionalidades.

4.1.1 Camada de apresentação

A camada de apresentação é constituída por três componentes: a *dashboard*, o ambiente de desenvolvimento e os jogos. Esses componentes representam as três grandes áreas da interface de utilizador do **Epik**. Para a estruturação e desenho do conteúdo dessas áreas foram utilizadas tecnologias como o **HyperText Markup Language (HTML) 5** [Wor13c]

¹<http://www.google.com/chrome/> consultado em 22-03-2013.

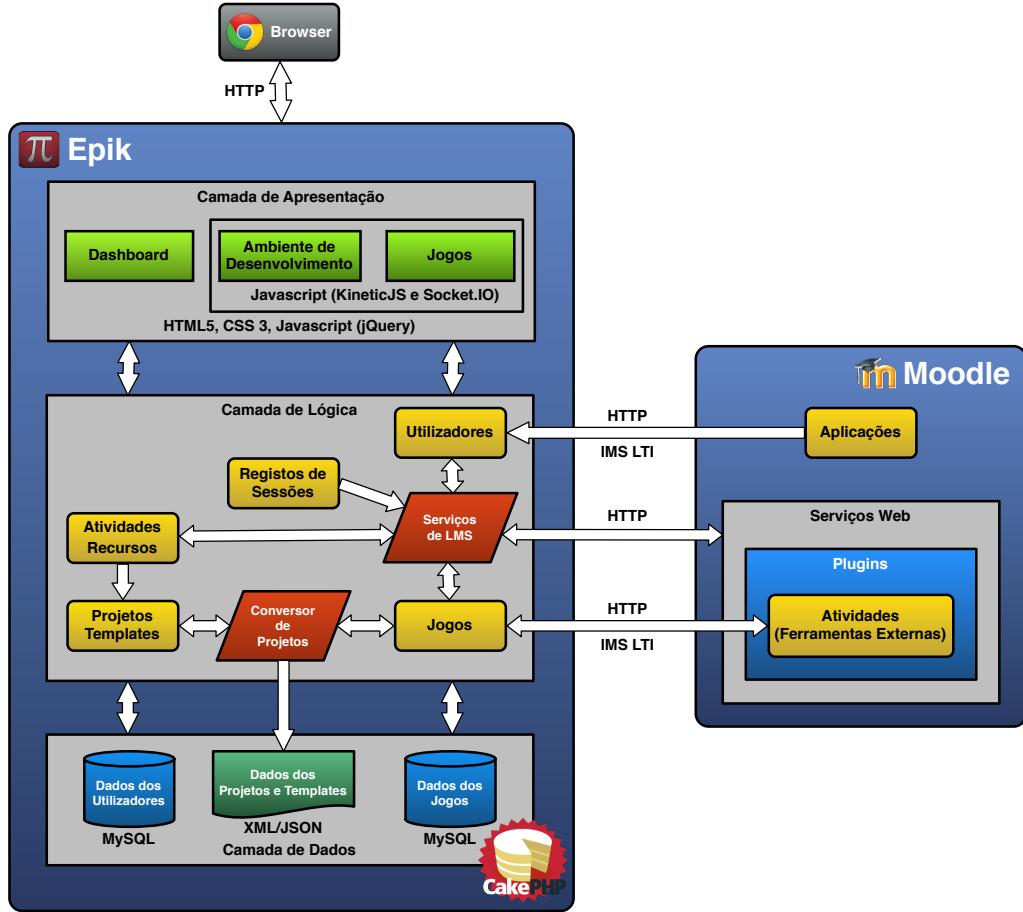


Figura 4.1: Arquitetura do servidor de gestão e desenvolvimento do Epik.

e o [Cascading Style Sheets \(CSS\) 3](#) [Wor13b], respetivamente. Decidimos utilizar estas tecnologias por serem atuais no que diz respeito ao desenvolvimento Web e por permitirem a criação de aplicações suportadas em qualquer sistema operativo e qualquer dispositivo. No caso particular do [HTML5](#), foi também por este possibilitar a criação de aplicações com meios de interação e colaboração de uma forma bastante mais fácil e rápida comparativamente a outras tecnologias existentes. Além disso, o seu uso para o desenvolvimento de jogos é também uma questão recente e que está a ser bem recebida, principalmente por possibilitar a criação de elementos gráficos e dinâmicos diretamente numa página Web, sem que sejam necessárias outras tecnologias [Sei11].

No que diz respeito à implementação de toda a lógica do cliente para cada uma dessas áreas foi utilizado o [jQuery 1.8](#) [The13a], uma biblioteca JavaScript que tem como objetivo simplificar o desenvolvimento de aplicações Web, em parte por suportar os vários *browsers* mais utilizados na atualidade.

Em conjunto com essa biblioteca foi ainda utilizado o [KineticJS 4.0.5](#) [Row13], uma *framework* também escrita em JavaScript que facilita a manipulação de elementos gráficos a serem desenhados numa página Web. Esta *framework* é apenas utilizada no contexto do ambiente de desenvolvimento e dos jogos, pois é a partir desta que todos os conteúdos

gráficos dos jogos são desenhados. Optámos por esta *framework* por ser mais generalista que os motores de jogos existentes, os quais são geralmente utilizados para o desenvolvimento de jogos comerciais que não se relacionam com os géneros de jogos do [Epik](#).

Para a componente de jogos foi ainda utilizada a biblioteca Socket.IO 9.11 [Rau13]. Esta biblioteca disponibiliza módulos para o cliente e servidor que facilitam a gestão e processamento dos vários pedidos que podem ser realizados em ambas as direções. O módulo utilizado nesta camada é o do cliente, enquanto que o do servidor é utilizado pelo servidor de execução de jogos. Por essa razão esta biblioteca será apresentada em maior detalhe na secção 4.2.

4.1.2 Camada de lógica

A camada de lógica, como é possível ver na Figura 4.1, é composta por dois tipos de componentes ilustrados com uma forma e cor diferente. Todos os pedidos recebidos de um cliente são sempre processados no contexto dos componentes de utilizadores, atividades, recursos, projetos, *templates*, jogos e registos. Cada um desses componentes gera os dados relacionados com os tipos de conteúdos a que se referem. Além desses há ainda os de conversão de projetos e de serviços de [LMS](#).

O componente de conversão de projetos é responsável por converter os dados de um projeto para outros formatos. É este componente que, por exemplo, carrega os dados de um projeto e os converte para o formato a enviar para o cliente ([XML](#) para [JavaScript Object Notation \(JSON\)](#)), que altera um projeto com base nos dados enviados pelo cliente ([JSON](#) para [XML](#)), ou que converte um projeto para o formato de jogo ([XML](#) para [JSON](#)). Este componente é utilizado apenas pelos de projetos, *templates* e jogos para processar determinados tipos de pedidos.

Por outro lado, o componente de serviços de [LMS](#) é responsável por tratar todo o tipo de pedidos relacionados com um [LMS](#). É este componente que, por exemplo, importa os dados de uma atividade ou recurso do [Moodle](#), ou valida e processa os dados de um pedido [IMS LTI](#) [IMS12].

Os pedidos de importação enviam para o [Moodle](#) informação via [HTTP](#) que permite identificar os dados a importar. Essa informação será processada pela componente de serviços Web do [Moodle](#) que disponibiliza uma [API](#) baseada num estilo de arquitetura [REST](#). Essa [API](#) permite aceder a informações relacionadas com vários tipos de *plugins* do [Moodle](#). Depois de processado o pedido, a resposta é enviada de volta via [HTTP](#) e os dados recebidos ([XML](#)) serão depois processados pela componente de serviços de [LMS](#) do [Epik](#).

No caso dos pedidos [IMS LTI](#) estes podem ser recebidos no contexto de um utilizador ou de um jogo. O pedido é enviado para o [Epik](#) via [HTTP](#) quando um utilizador do [Moodle](#) acede à aplicação ou a uma atividade (jogo), respetivamente. No caso dos jogos é ainda enviada uma resposta de volta para o [Moodle](#) que indica que o jogador finalizou o jogo. Esta resposta não se justifica no caso das aplicações visto que estas possibilitem

apenas que o utilizador do [Moodle](#) se autentique automaticamente no [Epik](#).

4.1.3 Camada de dados

A camada de dados é composta por duas bases de dados MySQL [Ora13] e pelos dados dos projetos e *templates* estruturados em [XML](#). Os dados armazenados nessas bases de dados ou nesses ficheiros são geralmente requisitados pelos componentes da camada de lógica.

A base de dados de utilizadores armazena todos os dados que estão de algum modo relacionados com um [utilizador](#), entre outros dados de sistema. Esta é a base de dados principal e a acedida por praticamente todos os componentes que compõem a camada de lógica. Além desta, há também a base de dados de jogos que armazena todos os dados relacionados com cada jogo existente no sistema, incluindo os registo de sessões. Esta base de dados só é acedida pelas componentes de jogos e registo maioritariamente para consulta de dados. No caso da geração de um jogo, são replicados para esta base de dados os dados das atividades e recursos que compõem o jogo. A replicação destes dados revelou-se necessária para que fosse possível garantir que um jogo é sempre válido após a sua geração, não sendo possível perder os dados das atividades ou recursos que o compõem.

Em relação aos dados dos projetos e *templates* de projetos, estes são armazenados e estruturados em ficheiros [XML](#) e são apenas acedidos e alterados pelo componente de conversão de projetos da camada de lógica. No caso dos *templates* de cenários, os seus dados são armazenados e estruturados em ficheiros [JSON](#).

Nas próximas secções são apresentadas a estrutura e organização da base de dados de utilizadores e dos ficheiros [XML](#) dos projetos e dos *templates*. A base de dados de jogos será apresentada no contexto do servidor de execução na secção 4.2.3.1, pois esta é mais relevante no contexto desse servidor, dado que é este que utiliza todos os dados contidos nessa base de dados e que gera os registo de sessões.

4.1.3.1 Base de dados de utilizadores

O esquema de tabelas da base de dados de utilizadores é apresentado na Figura 4.2. A maioria dos dados armazenados nesta base de dados estão relacionados com um utilizador. Os dados de um utilizador são armazenados na tabela *users*, apresentada ao centro no esquema. Esta tabela tem uma relação com a tabela de *lms* que permite determinar o tipo de [LMS](#) que o utilizador utiliza para importar conteúdos. Além dessa, a tabela *users* relaciona-se com as seguintes tabelas:

- *activities* (atividades) – uma atividade está sempre associada a um determinado utilizador e é sempre de um dos tipos existentes na tabela de *activities_types* (questão ou grupo de questões). Além disso, caso tenha sido importada estará também associada à tabela de *lms* que permite identificar o tipo de [LMS](#) usado para a realização dessa operação. As atividades podem ter dicas e recursos associadas através

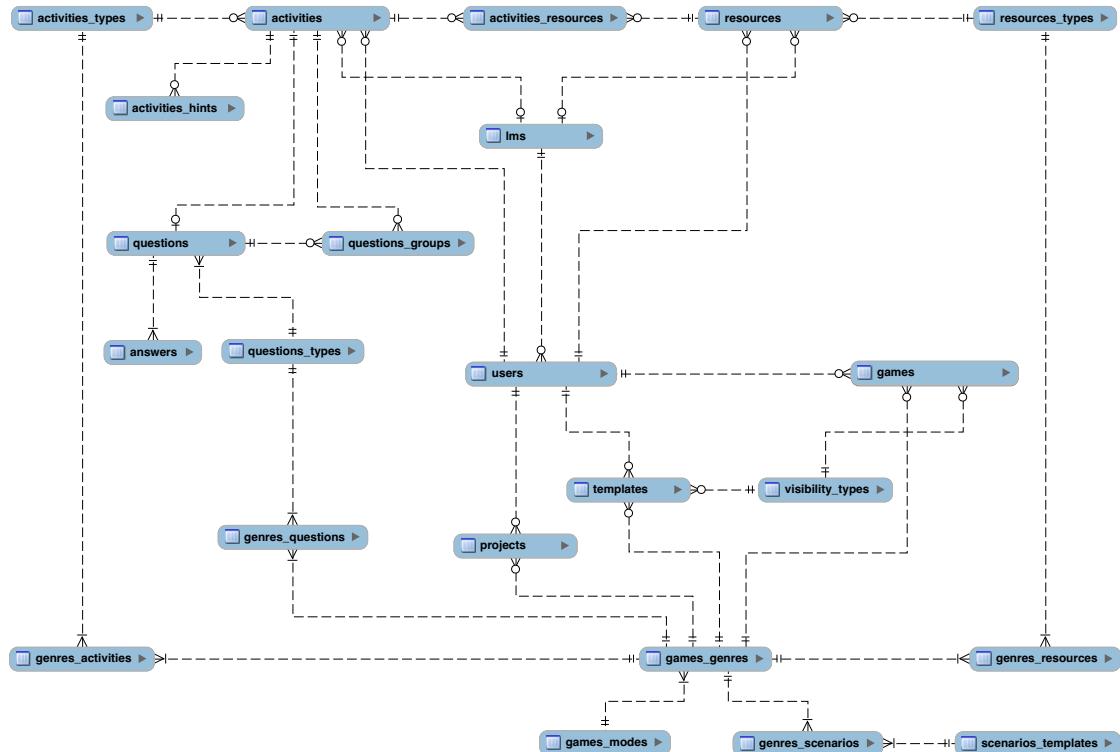


Figura 4.2: Esquema de tabelas da base de dados de utilizadores do Epik.

das tabelas de *activities_hints* e *activities_resources*, respetivamente. É a partir dessas relações que são obtidas as várias dicas e recursos que podem ser utilizados nas ajudas de dicas e consulta dos jogos de Questionário, respetivamente. Apesar de não ser visível no diagrama, as dicas e recursos só podem estar associadas a atividades do tipo questão, sendo esta gestão feita pelo componente de atividades da camada de lógica. Os dois tipos de atividades existentes no Epik são armazenados nas seguintes tabelas:

- *questions* (questões) – uma questão deverá estar sempre associada a uma atividade e deverá ser de um dos tipos existentes na tabela de *questions_types* (escolha múltipla, verdadeiro ou falso, ou resposta curta). Esta tabela de tipos define para cada um quantas respostas lhe podem estar associadas. Essas respostas são armazenadas na tabela de *answers* havendo sempre uma para cada questão que é identificada como correta;
- *questions_groups* (grupos de questões) – um grupo de questões é composto por um conjunto de associações entre a atividade correspondente e várias questões. O número de questões com que a atividade deverá estar relacionada deverá ser pelo menos duas. Quando uma questão que pertencia a um grupo é eliminada, este último poderá ser também eliminado caso o número de questões que lhe estão associadas fique menor que dois;

- *resources* (recursos) – um recurso está também sempre associado a um determinado utilizador, é sempre de um dos tipos existentes na tabela *resources_types* (áudio, imagem, vídeo e PDF) e poderá estar associado a um tipo de [LMS](#) pelas mesmas razões referidas para as atividades;
- *projects* (projetos) – um projeto, tal como nos casos anteriores, está sempre associado a um determinado utilizador e deverá estar também associado a um dos géneros/tipos de jogos existentes na tabela *games_genres* (Questionário Individual ou Colaborativo). Esta tabela de projetos não armazena qualquer informação acerca das propriedades gerais dos jogos ou dos seus cenários e conteúdos, todos esses dados são armazenados no ficheiro [XML](#) do projeto;
- *templates* de projeto – estes *templates* estão também sempre associados a um tipo de jogo, a um tipo de visibilidade e a um utilizador. A acessibilidade destes *templates* é determinada pelo tipo de visibilidade (público ou privado) que lhes está associado através da tabela de *visibility_types*. Estes *templates* têm também um ficheiro [XML](#) associado com o formato de um projeto que é copiado para os projetos criados a partir deste;
- *games* (jogos) – um jogo, tal como no caso anterior, está sempre associado a um tipo de jogo, a um tipo de visibilidade e a um utilizador.

Além das tabelas de projetos, *templates* e jogos, referidas acima, a tabela de *games_genres* tem também uma relação com as tabelas de tipos de atividades, recursos e questões. Estas relações definem para cada um desses tipos de conteúdos quais os que podem ser utilizados num determinado tipo de jogo. No caso dos jogos de Questionário todos os tipos são utilizados, porém isso poderá não fazer sentido para outros jogos que surjam no futuro. Estas relações são necessárias para o correto carregamento ou armazenamento dos dados de um projeto, geração de um jogo e criação da barra de ferramentas do ambiente de desenvolvimento.

Um tipo de jogo está também associado a um modo de jogo através da tabela de *games_modes* e a vários *templates* de cenários. Estes últimos são armazenados na tabela de *scenarios_templates* e podem ter vários tipos de jogos associados, pois um *template* deste género poderá adequar-se a mais do que um tipo de jogo. Estes *templates* têm ainda um ficheiro [JSON](#) associado com a informação do cenário estruturada da mesma forma que nos ficheiros dos projetos, sendo a única diferença o formato utilizado para armazenar esses dados. Escolhemos utilizar [JSON](#) para armazenar estes ficheiros porque é o formato utilizado para representar os dados no cliente e porque neste caso não é necessário garantir a validade desses dados, pois não podem ser alterados pelos utilizadores. Desta forma evitamos a necessidade de converter estes ficheiros de um formato para outro antes de serem enviados para o cliente.

4.1.3.2 Formato interno dos projetos e *templates*

Os ficheiros XML de cada projeto ou *template* de projeto armazenam informação acerca do estado corrente do projeto, das propriedades gerais do jogo, dos vários cenários que o compõem e respetivos conteúdos. A estrutura destes ficheiros é descrita e validada por uma linguagem definida em XML Schema. Na Figura 4.3 é representada uma versão reduzida dessa estrutura na forma de uma árvore de nós.

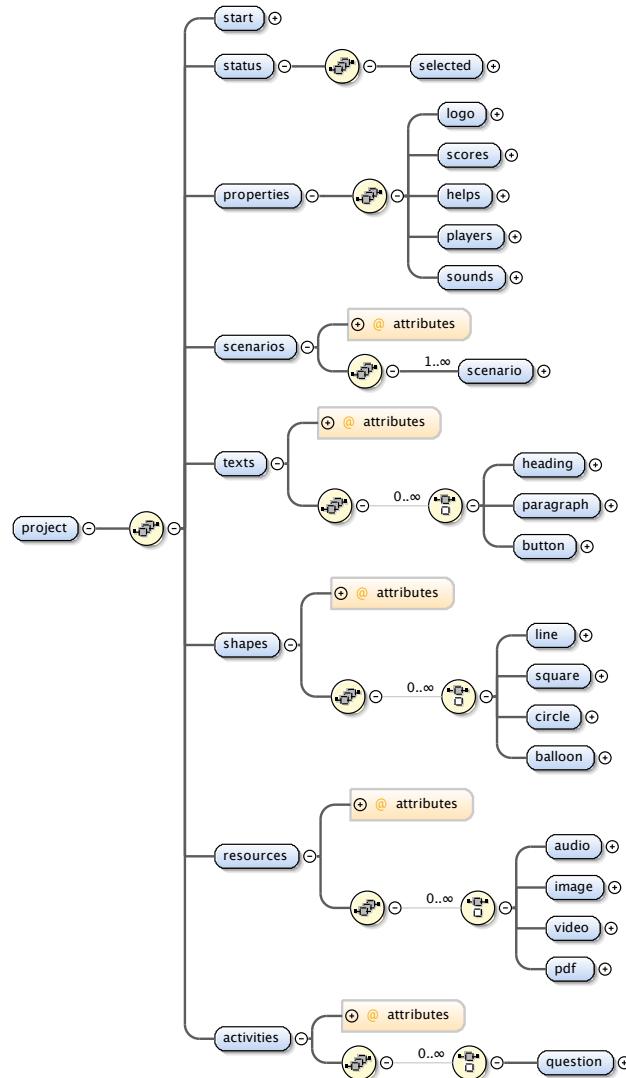


Figura 4.3: Representação em árvore da estrutura interna de um projeto do Epik.

Como é possível ver na figura, toda a informação armazenada nestes ficheiros encontra-se sempre sobre um nó denominado *project*. Dentro deste nó encontram-se vários outros nós que estão sempre organizados pela mesma ordem. Esses nós e os tipos de informação que armazenam são:

- **start** – guarda uma referência para o primeiro cenário do jogo;

- **status** – guarda o estado corrente do projeto, que de momento corresponde ao elemento que estava selecionado no painel de navegação do ambiente de desenvolvimento do [Epik](#) quando o utilizador guardou o projeto;
- **properties** – guarda os dados das várias propriedades gerais divididos nos seguintes nós:
 - **logo** – guarda a posição do logótipo no ecrã de jogo;
 - **scores** – guarda os nomes dos vários tipos de pontuações e se estas devem fazer parte dos registo de sessões;
 - **helps** – guarda o nome das ajudas, o seu valor e se devem fazer parte dos registo de sessões;
 - **players** – guarda o número de jogadores e a posição e estilos do painel de jogadores;
 - **sounds** – para cada tipo de som permitido, guarda uma referência para o recurso de áudio existente na base de dados de utilizadores. Se o utilizador mantiver os sons utilizados por omissão este nó fica vazio.
- **scenarios** – guarda os dados de todos os cenários de corpo que compõem o projeto pela ordem que são apresentados no painel de navegação do ambiente de desenvolvimento. Para cada cenário são armazenados um identificador único, o seu tipo, nome, referências e posição de cada conteúdo (*texts*, *shapes*, *resources*, ou *activities*), limite de ajudas, bónus, saltos e estilos;
- **texts** – guarda todos os elementos do tipo texto utilizados nos cenários de um projeto. Para cada um são armazenados um identificador único, o seu nome, conteúdo textual e estilos;
- **shapes** – guarda todas as formas geométricas usadas nos cenários de um projeto. Para cada uma são armazenados um identificador único, o seu nome, conteúdo textual e estilos;
- **resources** – guarda todos os recursos presentes nos cenários de um projeto. Para cada um são armazenados um identificador único, uma referência para os seus dados na base de dados, o seu nome e estilos;
- **activities** – guarda todas as atividades presentes nos cenários de um projeto. Para cada uma são armazenados um identificador único, uma referência para os seus dados na base de dados, o seu nome, pontuações, ajudas e estilos.

Além da estrutura de árvore apresentada acima, esta linguagem define ainda vários tipos de dados que permitem garantir que os ficheiros [XML](#) contêm dados válidos. Alguns exemplos são os vários tipos de bónus, saltos, ajudas, e também os valores mínimos e máximos de pontuações, ajudas, tamanhos, entre outros.

4.2 Servidor de execução de jogos

O servidor de execução de jogos do [Epik](#) tem como principal responsabilidade gerir todas as sessões de jogos em espera e a decorrer, no que diz respeito à validação de dados e colaboração entre jogadores. Este servidor funciona com base no protocolo WebSocket [\[Wor13a\]](#), o qual é já suportado pelas versões mais recentes dos *browsers* mais utilizados. Este protocolo permite estabelecer um canal de comunicação *full-duplex* entre clientes e servidores, permitindo que tanto o cliente como o servidor iniciem um processo de comunicação sem ser necessário restabelecer um canal por cada pedido. Antes da existência deste protocolo a única forma de um *browser* interagir com um servidor era utilizando técnicas de *polling*. Porém, a utilização dessas técnicas em aplicações como os jogos *multiplayer* ou outras que necessitem de informação de outros clientes para se manterem atualizadas, poderá resultar numa elevada latência de rede tornando-se desapropriadas para esse efeito [\[Sha12\]](#).

Para a implementação deste servidor foi utilizado o NodeJS 0.8.16 [\[Joy13\]](#), uma plataforma recente, que permite a implementação de servidores que suportam vários tipos de protocolos e que é já bastante utilizada para a implementação de servidores de jogos Web [\[Sei11, KSG⁺12, Sha12\]](#). As razões são principalmente por ser leve, eficiente e rápida, e por permitir o desenvolvimento da lógica do servidor na mesma linguagem que no cliente, ou seja, em JavaScript. Em conjunto com esta plataforma, como já foi referido, foi também utilizado o módulo de servidor da biblioteca Socket.IO 9.11 [\[Rau13\]](#). Esse módulo além de tratar de toda a gestão do canal de comunicação entre servidor e clientes, permite a utilização de um ou mais protocolos de comunicação de forma transparente, sendo possível definir a preferência entre estes. Isto significa que apesar de no nosso caso o protocolo preferencial ser o WebSocket, caso o *browser* do cliente não o suporte poderão ser utilizados outros meios de comunicação, como o XHR-polling, sem qualquer esforço adicional.

A arquitetura deste servidor é apresentada na Figura 4.4 como uma arquitetura de três camadas. A camada de comunicação é responsável pelo tratamento de pedidos realizados pelos clientes, no que diz respeito à sua receção e envio da respetiva resposta. Estes pedidos, após serem recebidos, são processados na camada de lógica, que em certos casos poderá necessitar de dados existentes na camada de dados. Depois de processados, os dados resultantes são devolvidos à camada de comunicação para serem enviados para um ou mais clientes.

4.2.1 Camada de comunicação

Os pedidos recebidos ou enviados pela camada de comunicação são sempre compostos por um nome e dados. O nome serve para identificar o pedido, e os dados contêm a informação necessária para o processar. O formato utilizado para estruturar estes pedidos é o [JSON](#) [\[Cro13\]](#).

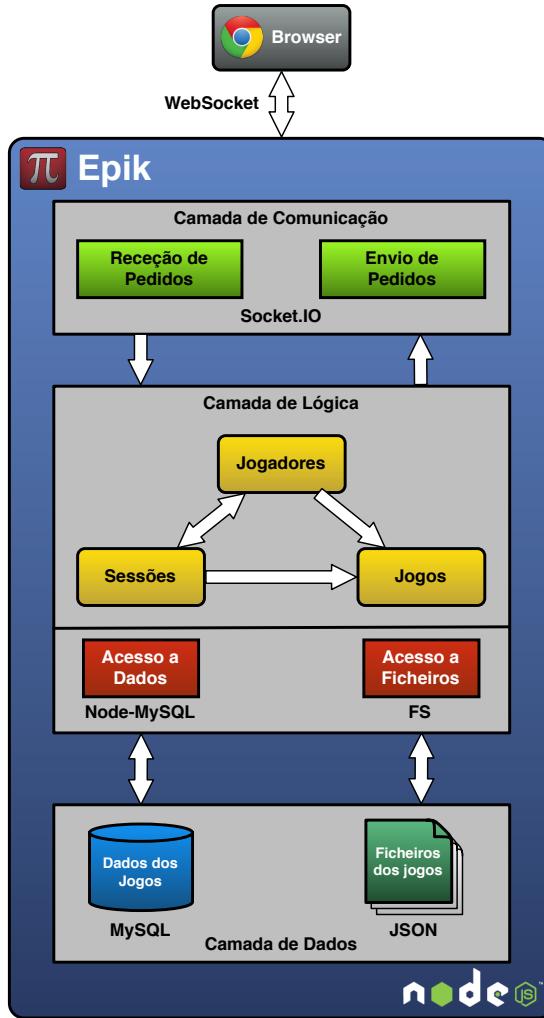


Figura 4.4: Arquitetura do servidor de execução de jogos.

Nesta camada, os processos de receção, envio, e identificação de pedidos são tratados pela biblioteca Socket.IO. Isto significa que foi apenas necessário definir os pedidos que podem ser realizados ou enviados por este servidor. Além disso, o formato utilizado para estruturar esses pedidos é também o imposto por esta biblioteca.

Os pedidos que podem ser realizados a este servidor serão apresentados na secção 4.2.4, juntamente com uma descrição detalhada do comportamento do servidor por cada pedido.

4.2.2 Camada de lógica

A camada de lógica, como é possível observar na Figura 4.4, é composta por vários módulos. Os que tratam de grande parte do processamento de pedidos são:

- **Jogadores** – gere todos os dados dos jogadores independentemente da sessão em que se encontram;
- **Sessões** – gere os dados de todas as sessões em espera e a decorrer;

- **Jogos** – gere os dados dos vários jogos que têm de momento sessões em espera, ou a decorrer;

Estes módulos, em certos casos, podem ter de comunicar entre si para processar determinados pedidos. Um pedido é sempre recebido no contexto de um cliente, o qual está associado a um jogador de uma dada sessão de um determinado jogo. Esta estrutura implica que através do jogador é possível obter a sessão, e que através desta é possível obter o jogo correspondente. De notar que um jogador se refere à relação cliente-pessoa e não à pessoa apenas, isto porque uma mesma pessoa pode estar a jogar várias instâncias do mesmo jogo ou de jogos diferentes e para cada instância conta como um jogador diferente.

Cada um desses módulos pode ainda necessitar de acesso a dados existentes na camada de dados para processar um determinado pedido. O acesso à base de dados de jogos pode ser realizado através do módulo de acesso a dados que utiliza como base o Node-MySQL [Gei13], um módulo externo do NodeJS para realizar *queries* a bases de dados MySQL [Ora13]. Do mesmo modo, o módulo de jogos pode também utilizar o de acesso ao sistema de ficheiros (disponibilizado pelo NodeJS), para aceder aos dados de um jogo gerados aquando da sua criação.

4.2.3 Camada de dados

A camada de dados, como referido acima, é composta pela base de dados de jogos e pelos ficheiros dos jogos gerados aquando da sua criação. Essa base de dados é a mesma que foi apresentada na camada de dados do servidor de gestão e desenvolvimento do [Epik](#). Este servidor pode aceder a essa base de dados para obter as várias atividades e recursos de um jogo, as dicas ou recursos associados a uma atividade, a pergunta correta de uma questão, entre outros. Além disso, no final do jogo os dados acerca do desempenho dos jogadores são armazenados nesta base de dados.

4.2.3.1 Base de dados de jogos

O esquema de tabelas da base de dados de jogos é apresentado na Figura 4.5. Nesta base de dados, os jogos são armazenados na tabela *games*, a qual referência a sua homónima na base de dados de utilizadores do servidor de desenvolvimento do [Epik](#). Nesta última as atividades e recursos estavam associadas a um utilizador, mas neste caso passam a estar associadas a um jogo.

As atividades e recursos armazenados nesta base de dados são nada mais que réplicas das que eram referenciadas nos projetos que deram origem aos jogos existentes. A forma como esses dados são armazenados e as relações entre essas tabelas são idênticas ao apresentado sobre a base de dados de utilizadores na secção 4.1.3.1. A grande diferença é que neste caso não se revelou necessário armazenar os grupos de questões, pois a informação necessária para os identificar está nos ficheiros de dados dos jogos. Tal como

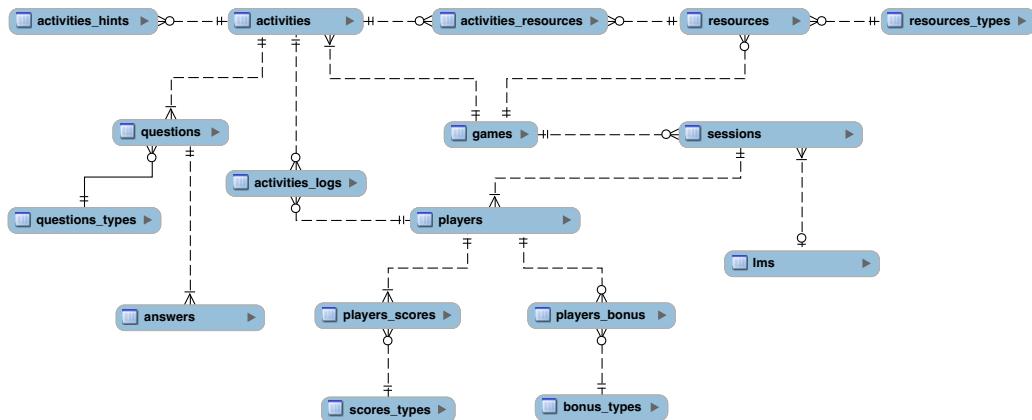


Figura 4.5: Esquema de tabelas da base de dados de jogos do Epik.

na base de dados de utilizadores, esta tem também tabelas que armazenam os tipos de questões e de recursos.

Relacionada com a tabela *games* está também a tabela *sessions* que armazena os dados das várias sessões de um jogo que foram terminadas com sucesso. Uma sessão poderá estar associada a um *lms* e a vários *players* (jogadores). A tabela de *lms* armazena a informação dos vários LMS que já comunicaram com o Epik através de pedidos IMS LTI. Entre essa informação encontra-se o URL necessário para enviar uma nota de volta para o LMS, o qual, segundo a especificação do standard IMS LTI, deverá ser sempre igual para um determinado sistema.

No que diz respeito à tabela de jogadores, esta armazena a informação necessária para identificar o jogador, o seu respetivo utilizador no LMS e o número de pedidos de ajuda a que respondeu e que realizou. Esta tabela relaciona-se com as de (i) *players_scores* (pontuações do jogador) – armazena os pontos totais recebidos por cada tipo de pontuação permitido no jogo; (ii) *players_bonus* (bónus dos jogadores) – armazena o valor total recebido por cada tipo de bónus em cada cenário do jogo; (iii) *activities_logs* (registos de atividades) – armazena os pontos de recompensa e penalização recebidos em cada atividade, bem como o número de tentativas que realizou até acertar. Todos esses dados, incluindo a informação acerca das ajudas, só são armazenados se o utilizador que criou o jogo definiu que deveriam fazer parte do registo de sessões.

4.2.3.2 Formato interno dos jogos

Além dos dados armazenados na base de dados dos jogos, tal como para os projetos no servidor de desenvolvimento do Epik, a informação acerca das propriedades, cenários e conteúdos dos jogos são armazenados em ficheiros. Para cada jogo são sempre criados dois ficheiros com os dados a serem carregados para o cliente e servidor, respetivamente. No entanto, o formato utilizado para estruturar estes ficheiros é JSON e não XML, isto porque estes ficheiros contêm informação que já não pode ser alterada pelo utilizador, não sendo necessário garantir a sua validade. Além disso o JSON é um formato bastante

mais leve e compacto que o [XML \[NPRI09\]](#).

Os dados armazenados em ambos os ficheiros estão organizados de uma forma bastante idêntica à dos projetos. No entanto, neste caso não existe um nó geral com toda a informação armazenada no seu interior, nem um nó com o estado do projeto.

O ficheiro do cliente contém sempre os nós para as propriedades gerais, cenários, textos, formas geométricas, recursos e atividades. Para cada um desses elementos é armazenada neste ficheiro a informação necessária para visualização, como o nome das pontuações e ajudas, os sons, as posições dos conteúdos de um cenário e os estilos.

Por outro lado, o ficheiro do servidor contém sempre os nós com o cenário de início, propriedades gerais, cenários e atividades. Para cada um deles é guardada a informação necessária para validar os pedidos do cliente e para executar o jogo. Alguns exemplos são o valor mínimo e máximo de jogadores, o número de ajudas, os bónus e saltos de um cenário, as pontuações e ajudas de uma atividade, entre outros.

4.2.4 Execução de um jogo

Enquanto um jogo está a decorrer o servidor mantém em memória principal toda a informação que diz respeito ao estado da sessão e de cada jogador. Ao longo de um jogo os clientes poderão fazer vários pedidos que servem em alguns casos apenas para obter dados e noutras casos para atualizar dados e notificar os outros jogadores sobre essas alterações. Esses pedidos nunca enviam para o servidor informação acerca de pontuações, ajudas, ou fluxo do jogo, pois essa informação é sempre gerada pelo servidor com base em ações dos jogadores. Optámos por esta abordagem de forma a garantir a fiabilidade dos dados a guardar nos registos de sessões, pois assim é o servidor que os gera e mantém não sendo possível manipulá-los a partir de uma entidade externa.

Um pedido apenas informa o servidor de que um jogador clicou num determinado botão, que fez uma tentativa de resposta a uma pergunta, que utilizou uma ajuda, entre outros. Com base na informação recebida por cada um desses tipos de pedidos, se esta for válida no contexto do cenário corrente, o servidor irá ordenar uma mudança de cenário, calcular pontuações, ou decrementar ajudas, respetivamente. Quando o jogo termina os dados da sessão e dos jogadores são armazenados na base de dados de jogos para que um utilizador os possa consultar mais tarde e são depois removidos de memória principal.

De forma a descrever o comportamento deste servidor, são apresentados na lista abaixo uma breve descrição e um diagrama de atividades para os vários pedidos que um cliente pode enviar ao servidor. Alguns dos diagramas foram colocados em anexo por serem pouco relevantes e/ou complexos.

- **Registo de jogador** (Figura 4.6) – quando um jogador inicia um jogo, a sua informação é enviada para o servidor. Essa informação é composta pelo nome, avatar, e LMS do jogador. Os dados do LMS serão usados para encontrar uma sessão para o jogador. Depois de lhe ser atribuída uma sessão são enviados para o cliente os dados das atividades e recursos do jogo;

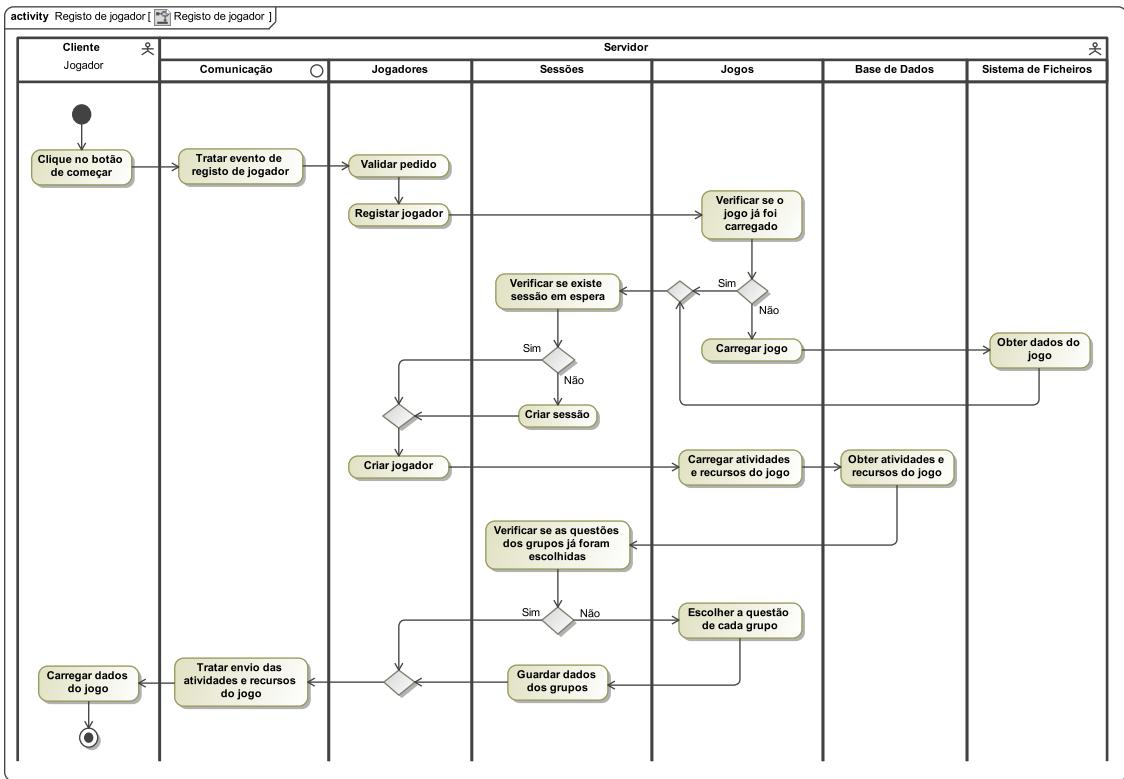


Figura 4.6: Diagrama de atividades para o pedido de registo de jogador.

- **Iniciar o jogo** (Figura 4.7) – quando o cliente termina de carregar os dados do jogo (enviados no pedido anterior), notifica o servidor de que está pronto para começar o jogo. Ao receber esta informação o servidor vai verificar se já há jogadores suficientes para iniciar o jogo, e se sim inicia-o;
- **Clique em botão** (Figura B.1) – quando um jogador clica num botão de continuar ou saltar, o servidor é notificado e verifica se o número de jogadores que clicou é suficiente para mudar para o próximo cenário (Figura 4.11), se não for informa os outros acerca do atual estado do jogador;
- **Fim de tempo** (Figura B.2) – quando o tempo limite de permanência no cenário corrente é atingido, o servidor é notificado e verifica se o número de jogadores que atingiu esse tempo é suficiente para mudar para o próximo cenário (Figura 4.11), se não for informa os outros acerca do atual estado do jogador;
- **Resposta a pergunta** (Figura 4.8) – quando um jogador responde a uma pergunta, a resposta escolhida é enviada para o servidor e este irá tratar de a validar. Independentemente de a resposta estar certa ou errada todos os jogadores são notificados acerca das novas pontuações. Se a resposta estiver correta e já todos terminaram as atividades do cenário corrente pode ainda ocorrer uma mudança de cenário (Figura 4.11);

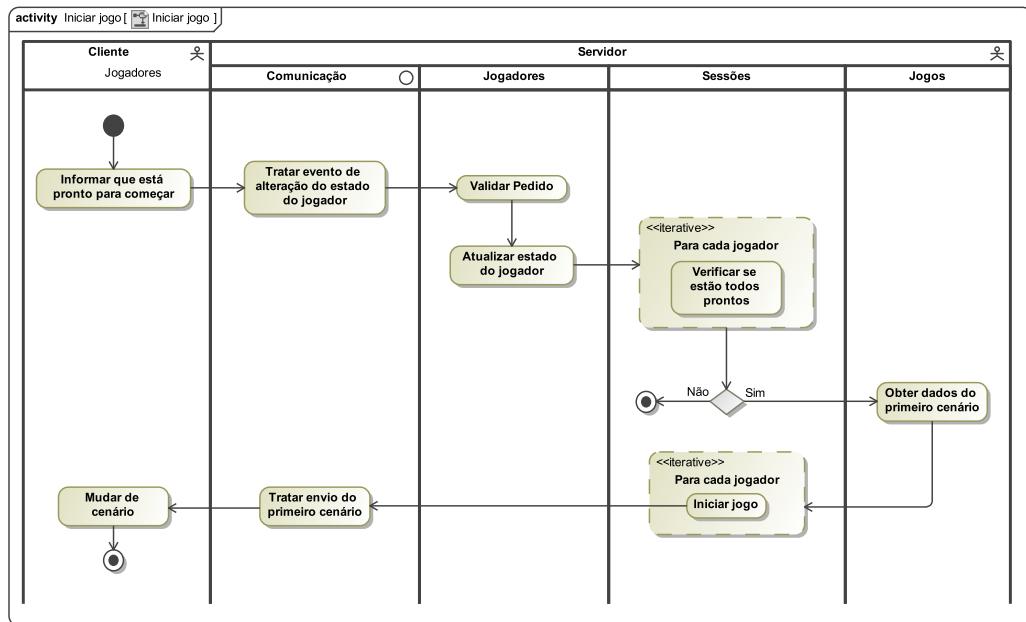


Figura 4.7: Diagrama de atividades para o pedido de iniciação de jogo.

- **Uso de ajuda** (Figura 4.9) – quando um jogador utiliza uma ajuda é enviado um pedido de ajuda para o servidor. Este pedido é tratado de diferentes formas consoante o tipo de ajuda e o modo de jogo. Em jogos *singleplayer* o servidor escolhe os recursos, dicas, ou respostas erradas aleatoriamente e envia-os para o jogador (Figura 4.10). Por outro lado, nos jogos *multiplayer*, no caso das ajudas de dicas e remoção de respostas, o servidor escolhe um jogador que já tenha resolvido a atividade e envia-lhe todas as dicas ou respostas erradas, respetivamente;
- **Resposta a pedido de ajuda** (Figura 4.10) – ao receber os dados de resposta a um pedido de ajuda, o servidor termina a colaboração entre os dois jogadores, caso esta exista, e envia os dados da ajuda ao jogador que fez o pedido.

Como foi descrito anteriormente, quando o servidor trata um pedido de clique em botão, fim de tempo, ou resposta a pergunta, dependendo do estado do jogo, poderá ocorrer uma mudança de cenário. Essa operação ocorre quando o número de jogadores é suficiente para realizar um determinado salto. Ao mudar de cenário todos os estados relacionados com o cenário corrente são reiniciados e é enviada para cada cliente a informação do próximo cenário. O diagrama de atividades deste processo é apresentado na Figura 4.11.

Quando os jogadores atingem o último cenário, a operação de mudança de cenário irá executar a de término de jogo (Figura B.3). Nesse caso a sessão é guardada e depois é apresentado o cenário de classificações aos jogadores. No entanto, o jogo pode também ser terminado porque não existem jogadores suficientes para continuar. Nessa situação a sessão é eliminada e o cenário de fim de jogo é apresentado aos jogadores.

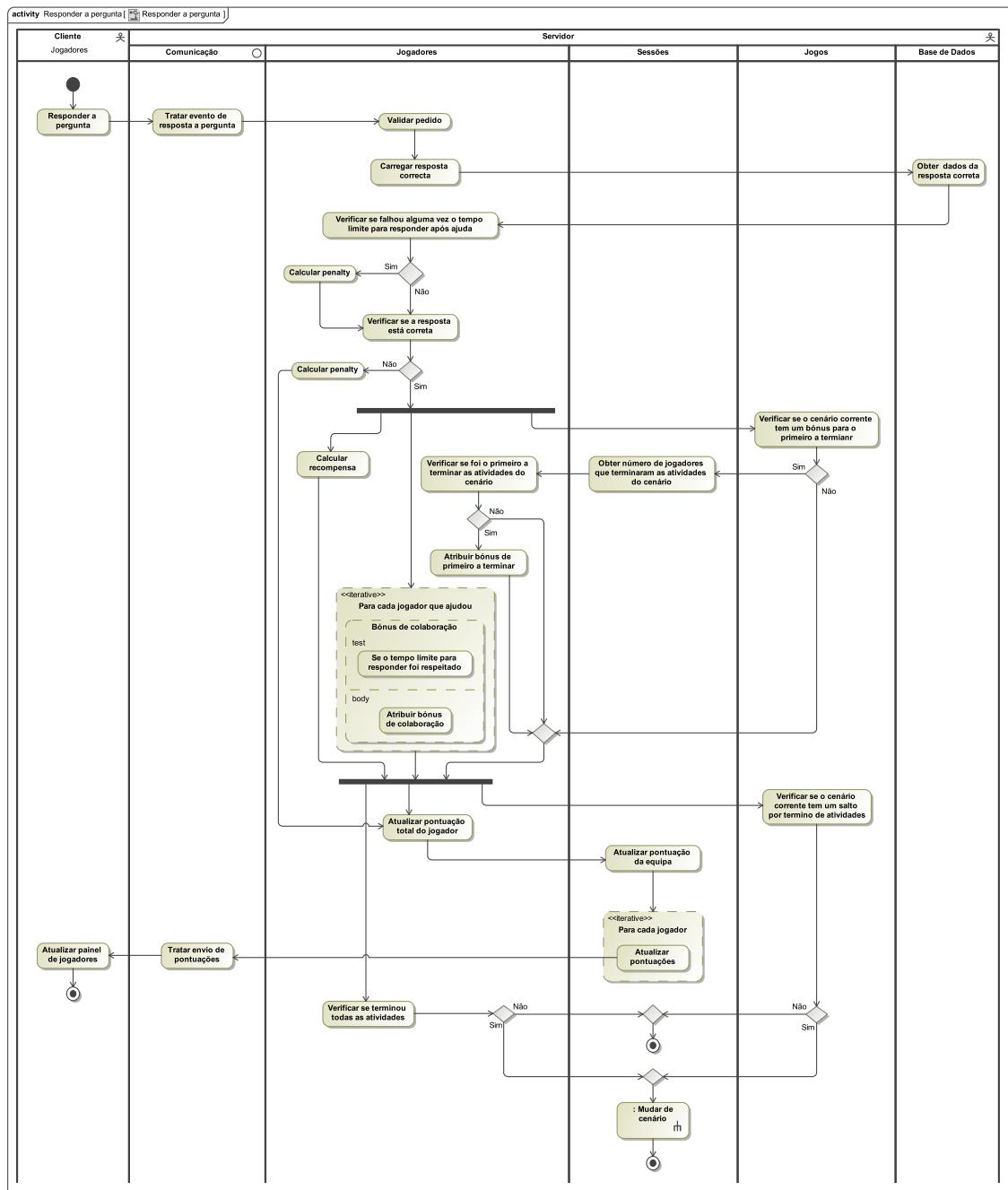


Figura 4.8: Diagrama de atividades da notificação de resposta a pergunta.

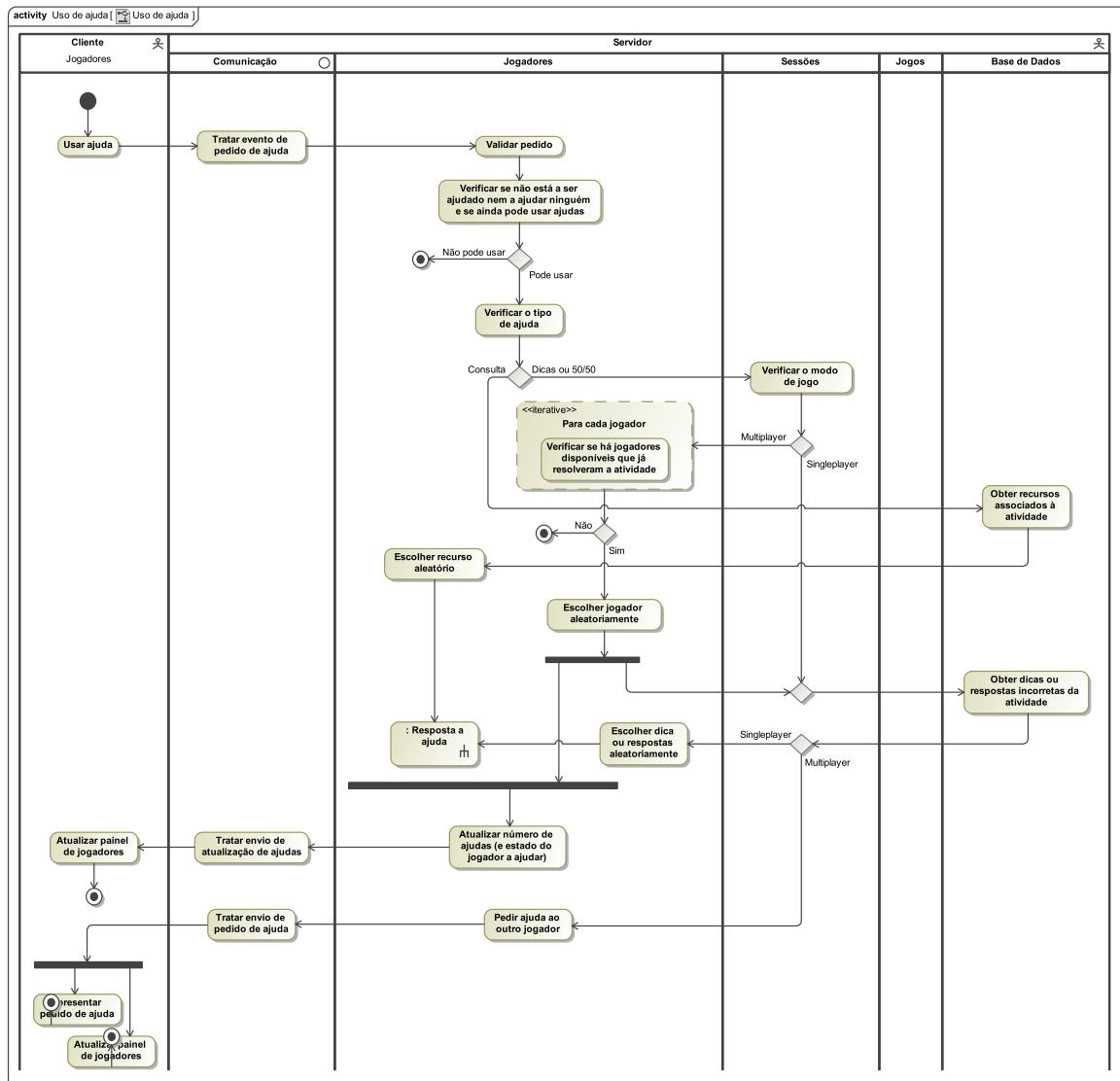


Figura 4.9: Diagrama de atividades da notificação de uso de ajuda.

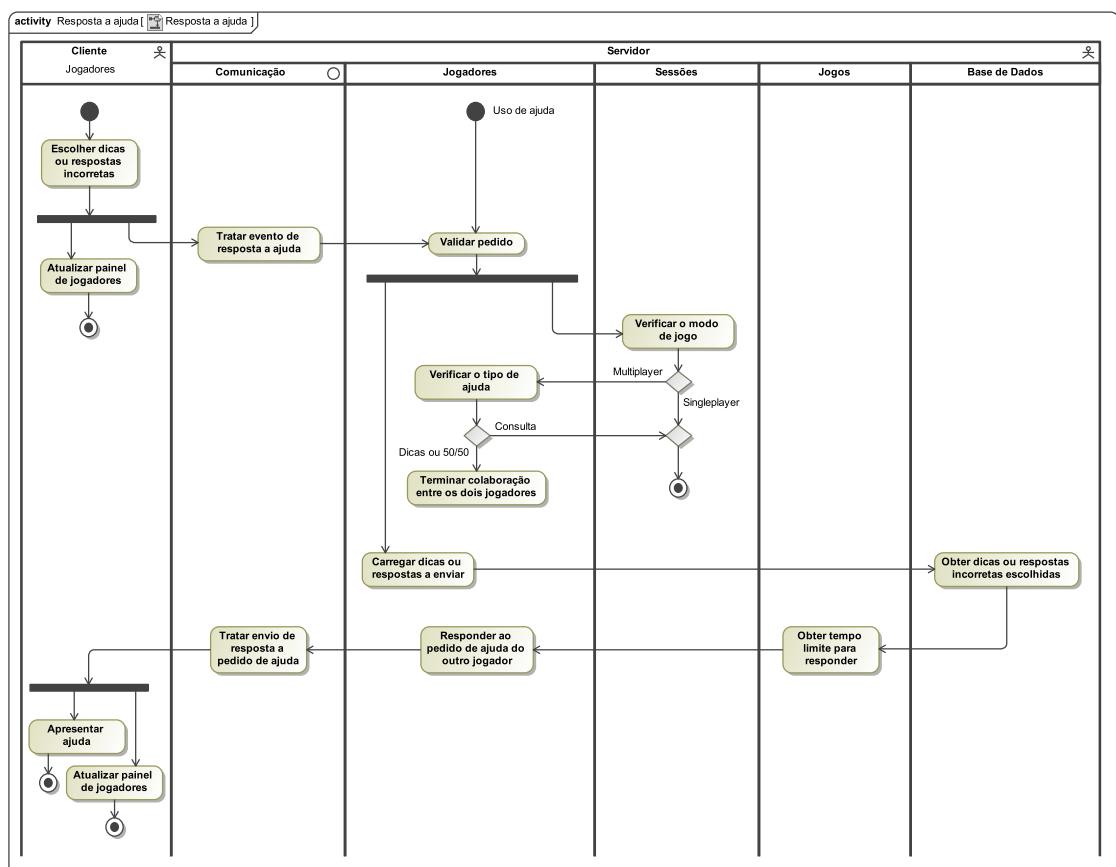


Figura 4.10: Diagrama de atividades da notificação de resposta a um pedido de ajuda.

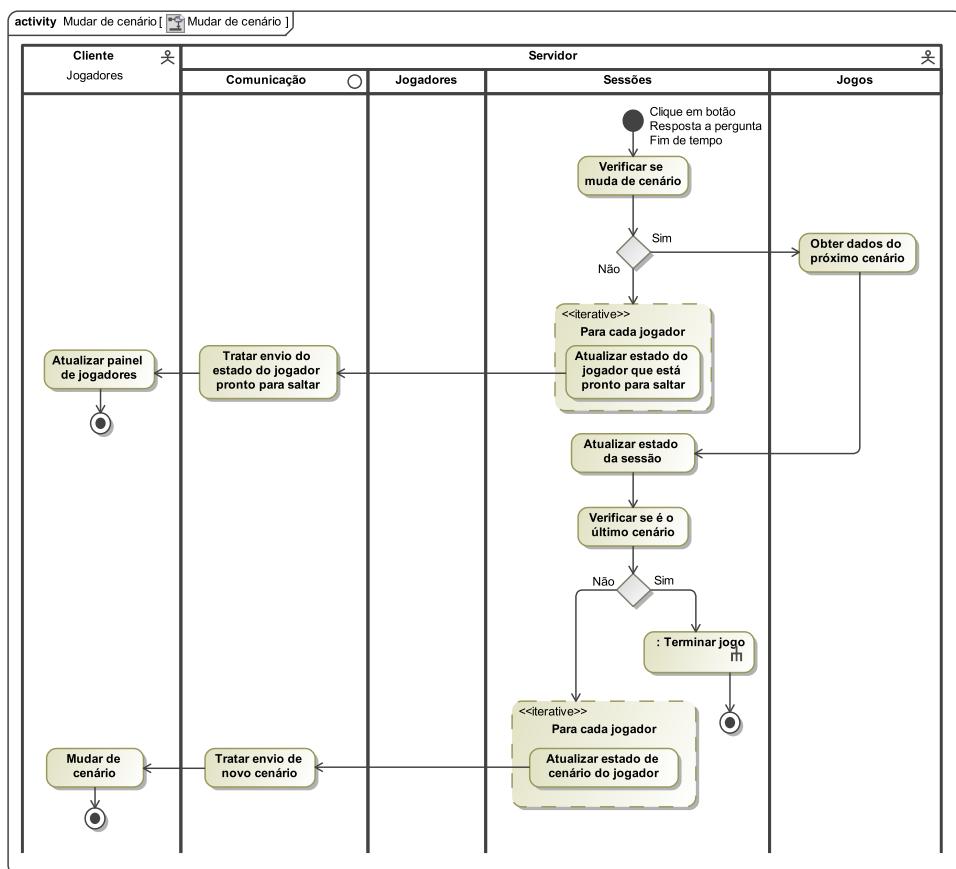


Figura 4.11: Diagrama de atividades de uma mudança de cenário.

O servidor deteta que um jogador abandonou o jogo quando o canal de comunicação que existe entre ambos é terminado. Quando isso acontece o jogador é removido da sessão em que se encontrava e depois uma das seguintes situações pode ocorrer: (i) se a sessão estava em espera e ficou sem jogadores esta é eliminada; (ii) se a sessão estava a decorrer e o número de jogadores ficou abaixo do mínimo permitido é terminada e é invocada a operação de mudança de cenário; (iii) caso contrário, se a sessão já estivesse a decorrer, os outros jogadores seriam notificados de que um jogador abandonou o jogo. Este processo é apresentado no diagrama de atividades na Figura B.4.

Qualquer uma das operações que este servidor suporta podem em alguns casos levar a uma situação de erro inesperado. Quando isso acontece é enviada uma mensagem para o cliente que fez o pedido com o código do erro. Exemplos de situações que podem levar a um erro deste tipo são: (i) os dados do jogador enviados ao inicio não estão completos; (ii) o jogador tentou pedir ajuda enquanto está ainda à espera da resposta a outro pedido de ajuda; (iii) ou o jogador perdeu a ligação com o servidor. Neste último caso, para além do erro é também apresentado o cenário de fim de jogo ao jogador.

4.3 Extensões para o Moodle

De forma a permitir que o [Epik](#) importe conteúdos didáticos existente no [Moodle](#) e de o integrar neste último, como foi referido anteriormente, revelou-se necessário criar várias extensões para esse [LMS](#). Essas extensões são novos *plugins*, atualizações de *plugins* já existentes e dois novos pacotes de lógica para o [Moodle](#). Estes últimos introduzem dois novos conceitos neste [LMS](#). A criação de pacotes para o [Moodle](#) não é uma prática muito comum, pois geralmente uma extensão para este [LMS](#) encaixa-se sempre num dos pacotes de *plugins* existentes no sistema. No entanto, não encontrámos nenhum tipo de *plugin* que se adequa-se ao que pretendíamos.

As extensões criadas são as seguintes:

- **Pacote de Aplicações** – este pacote permite aos utilizadores adicionar, remover, ou aceder a aplicações que utilizam o *standard IMS LTI*. A informação dessas aplicações é armazenada numa nova tabela da base de dados do [Moodle](#), designada *applications*. Esta tabela relaciona-se com as de *user* e *lti_types* já existentes no [Moodle](#);
- **Bloco de Aplicações** – é um novo *plugin* do tipo bloco, dependente do pacote de aplicações referido acima, que permite ao utilizador gerir e aceder às suas aplicações em qualquer momento independentemente da página em que se encontra;
- **Bibliotecas de serviços externos** – estas bibliotecas estendem a [API](#) de serviços externos do [Moodle](#), adicionando novas funções aos *plugins* das atividades de questionário (*quiz*) e recurso (*resource*) já existentes no [Moodle](#). Estas funções permitem que uma aplicação externa acesse à informação de questionários ou recursos de um determinado utilizador do [Moodle](#);

- **Pacote de Instituições** – este pacote permite ao administrador de um sistema [Moodle](#) adicionar ou remover páginas institucionais externas ao sistema que podem depois ser acedidas por qualquer utilizador. O objetivo deste pacote é apenas fornecer uma nova forma de aceder a este tipo de páginas sem que seja necessário abandonar o contexto do [LMS](#). Este pacote não é relevante para a integração do [Epik](#) no [Moodle](#) é apenas um extra que considerámos interessante. Tal como para o pacote de aplicações, este introduz uma nova tabela na base de dados do [Moodle](#), designada *institutions*, que não se relaciona com nenhuma outra;
- **Chat Global** – é um novo *plugin* do tipo bloco que fornece aos utilizadores do [Moodle](#) uma nova forma de comunicação. Este *plugin* não é relevante para a integração do [Epik](#) no [Moodle](#), mas contribuí como um outro tipo de atividade colaborativa que poderá ser utilizada para tirar dúvidas e realizar trabalhos ou outras atividades em grupo. Para a implementação deste *plugin* foi criado um servidor que funciona com base no protocolo WebSocket [Wor13a] utilizando a biblioteca Ratchet [Bod13]. Esta biblioteca é similar ao Socket.IO para NodeJS, mas está escrita em PHP e apenas implementa o protocolo WebSocket sem disponibilizar qualquer forma de gestão dos vários canais de comunicação. Escolhemos utilizar esta biblioteca porque este servidor necessita de acesso às API's do [Moodle](#), as quais estão escritas em PHP. O servidor desenvolvido é responsável pela gestão dos vários utilizadores *online* e da comunicação entre estes. Para essa gestão foram criadas novas tabelas para a base de dados que armazenam os utilizadores *online*, as sessões de *chat* abertas, os participantes nessas sessões e as mensagens trocadas entre si. Este *plugin* disponibiliza também uma página de configuração que permite ao administrador uma forma de alterar o URL e porta usados para comunicar com o servidor, além de outras opções de estilos;
- **Cubic** – é um novo *plugin* do tipo tema, que adiciona novas características à interface de utilizador de um sistema [Moodle](#), como a barra de utilizador, os sistemas de notificações, uma nova forma de organização dos conteúdos de um curso, entre outros. Para implementar este tema foi utilizado o [HTML5](#) [Wor13c], o [CSS 3](#) [Wor13b] e o [jQuery 1.7](#) [The13a]. Este tema disponibiliza uma página de configuração que permite ao administrador alterar vários aspectos de estilos, como o logótipo na barra de utilizador, as cores da barra, as cores dos botões, entre outros.

O Cubic conta também como uma forma de integrar todas as extensões desenvolvidas. No entanto, todas elas podem ser usadas de forma independente, com exceção do bloco de aplicações.

Com estes novos pacotes e *plugins*, é possível aceder ao [Epik](#) através do [Moodle](#) sem perder o contexto deste último devido às funcionalidades que o tema Cubic oferece. Para além disso, com os novos serviços adicionados à API de serviços do [Moodle](#), é possível importar para o [Epik](#) perguntas e conteúdos didáticos que podem ser utilizados nos jogos

desenvolvidos neste último.

4.4 Conclusões

De um modo geral, como foi visto, a plataforma [Epik](#) é composta por dois servidores, um responsável pela gestão de conteúdos e desenvolvimento de jogos e outro pela execução destes últimos. Esses servidores utilizam o protocolo [HTTP](#) e [WebSocket](#) para comunicar com os clientes, respetivamente. O servidor de gestão e desenvolvimento lida maioritariamente com dados que estão associados a um utilizador do sistema, os quais são armazenados numa base de dados ou em ficheiros. Enquanto que o de execução de jogos lida com os dados de todos os jogos executáveis e é responsável pela sua execução. Além disso, ambos os servidores partilham a gestão da base de dados de jogos.

Para a integração do [Epik](#) no [Moodle](#) e importação de dados deste último, foram ainda desenvolvidas várias extensões para esse [LMS](#). Entre essas extensões, as mais relevantes são as aplicações, os novos serviços externos e o [Cubic](#), as quais permitem aos utilizadores gerir aplicações, aceder-lhes sem abandonar o contexto do [LMS](#) e importar perguntas e conteúdos didáticos deste último, respetivamente. Permitindo assim que o [Epik](#) seja integrado de uma forma harmoniosa num ambiente de ensino, bem como os jogos nele criados.

5

Avaliação

Neste capítulo são apresentados 4 estudos por nós realizados que tinham como objetivo avaliar: o desenvolvimento de jogos no [Epik](#), a jogabilidade dos jogos de Questionário do [Epik](#), a utilização desses jogos no ensino e a utilização do tema Cubic para o [Moodle](#). Para a avaliação desses quatro aspetos foram criados três inquéritos de satisfação e qualidade a serem realizados por grupos de utilizadores distintos.

Neste capítulo são apresentados os resultados obtidos nesses inquéritos, o seu significado dentro do contexto deste trabalho e as conclusões que deles se podem retirar.

5.1 Descrição dos inquéritos e grupos de utilizadores

Neste processo de avaliação, que decorreu durante duas semanas, participaram vários utilizadores com diferentes papéis no ensino (professores e estudantes), e de diferentes áreas e níveis de ensino. O nosso principal objetivo era obter opiniões variadas que nos permitissem avaliar os vários aspetos em causa, com base nas necessidades de diferentes áreas de ensino. Para a realização desta avaliação, a plataforma [Epik](#)¹ e o tema Cubic² foram instalados numa máquina do Departamento de Informática da FCT/UNL e podem ser acedidos a partir do URL em nota de rodapé.

Os vários inquéritos utilizados para a realização deste processo de avaliação foram divididos em várias secções de perguntas relacionadas com diferentes aspetos. Na sua maioria, essas perguntas eram de resposta rápida de onde deveria ser escolhido um valor numa escala de 1 (negativo) a 5 (positivo), ou escolhida uma ou mais hipóteses de resposta das disponibilizadas. Nas perguntas de escala o valor intermédio (3) significa

¹<http://epik.di.fct.unl.pt/epik> consultado em 22-03-2013

²<http://epik.di.fct.unl.pt/moodle> consultado em 22-03-2013

indiferença ou neutralidade. A última pergunta desses inquéritos era sempre de resposta aberta onde os inquiridos poderiam deixar comentários, críticas ou sugestões.

Para avaliar o desenvolvimento de jogos no [Epik](#), e a utilização de jogos no ensino por parte de quem ensina, foi pedido a um grupo de 35 professores ou instrutores que realizassem várias tarefas no [Epik](#) e que no final respondessem ao inquérito apresentado na secção A.1. Das 45 perguntas que compõem esse inquérito, as perguntas de 5 a 19 dizem respeito à utilização de jogos no ensino, e as de 20 a 45 ao processo de desenvolvimento de jogos. Neste processo de avaliação obtivemos 13 resultados de pessoas entre os 21 e 55 anos de idade, onde 62% eram do sexo masculino e 38% do sexo feminino. Além disso, como apresentado no gráfico 5.1, 85% dos inquiridos tinham um curso superior e os restantes 15% tinham o ensino secundário. Os resultados obtidos acerca da utilização de jogos no ensino por parte de professores serão apresentados na secção 5.2.1, e os acerca do processo de desenvolvimento na secção 5.3.

3. Qual o seu grau de instrução?

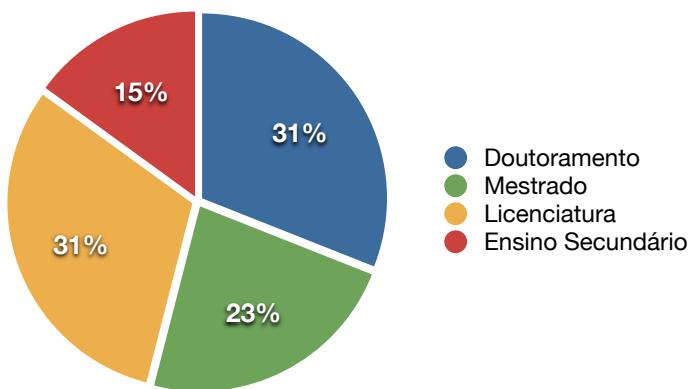


Figura 5.1: Graus de instrução dos participantes na avaliação de utilização e desenvolvimento de jogos.

Relativamente à avaliação da jogabilidade dos jogos do [Epik](#), e da utilização de jogos no ensino por parte de quem aprende, foi pedido a um grupo de aproximadamente 100 estudantes ou ex-estudantes que jogassem pelo menos um Questionário Individual e um Questionário Colaborativo por nós desenvolvidos. Depois disso foi-lhes pedido que respondessem ao inquérito apresentado na secção A.2. Das 29 perguntas que compõem esse inquérito, as perguntas de 5 a 15 dizem respeito à utilização de jogos no ensino, e as de 16 a 29 à jogabilidade dos jogos. Neste inquérito obtivemos 37 respostas de pessoas entre os 16 e os 55 anos de idade, das quais 65% eram do sexo masculino e 35% do sexo feminino. Além disso, como apresentado no gráfico 5.2, 73% dos inquiridos tinham um curso superior e os restantes 24% e 3% eram estudantes com o ensino secundário e o 3º ciclo de escolaridade, respetivamente. Os resultados obtidos acerca da utilização de jogos por parte dos estudantes serão apresentados na secção 5.2.2, e os acerca da jogabilidade dos jogos na secção 5.4.

No que diz respeito à avaliação do tema Cubic para o [Moodle](#), foi pedido a um grupo

3. Qual o seu grau de instrução?

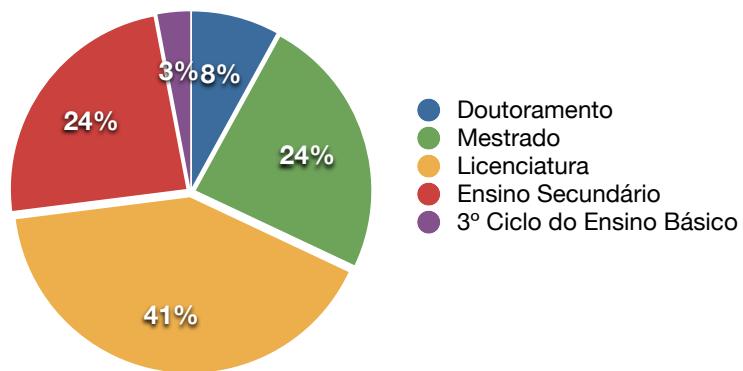


Figura 5.2: Graus de instrução dos participantes na avaliação de utilização e jogabilidade dos jogos.

de 50 professores e estudantes, dos quais alguns já tinham participado nas outras avaliações, que realizassem várias tarefas nesse LMS, incluindo experimentar a integração do [Epik](#) com o [Moodle](#). De seguida, foi-lhes solicitado que respondessem ao inquérito apresentado na secção A.3. Desse grupo de pessoas, apenas 10 entre os 16 e os 55 anos de idade responderam ao inquérito, das quais 60% eram do sexo masculino e 40% do sexo feminino. Além disso, como é possível ver no gráfico 5.3, 90% dos inquiridos tinham um curso superior e os restantes 10% tinham o 3º ciclo. Os resultados desta avaliação são apresentados na secção 5.5.

3. Qual o seu grau de instrução?

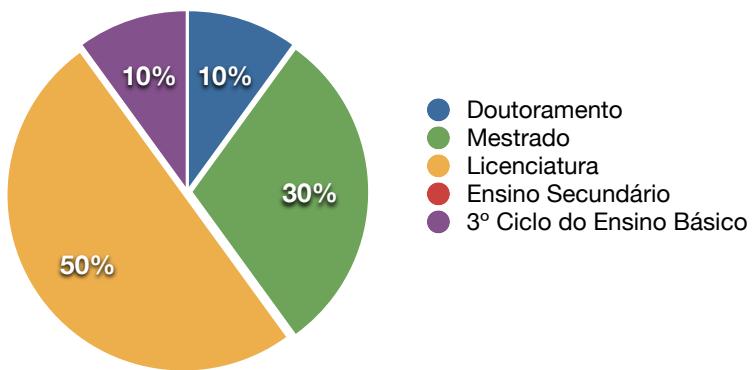


Figura 5.3: Graus de instrução dos participantes na avaliação do tema Cubic.

5.2 Utilização dos jogos do Epik no Ensino

A utilização dos jogos do [Epik](#) no ensino foi avaliada, como referido anteriormente, pelos mesmos grupos de professores e estudantes que avaliaram o desenvolvimento e jogabilidade desses jogos, respetivamente. As perguntas que foram realizadas a ambos os grupos pretendiam obter opiniões acerca da utilização de jogos no ensino, de questionários como

atividades educativas e dos jogos de questionário do [Epik](#). Em relação a estes últimos foram realizadas questões acerca da sua utilização no ensino como forma de avaliação e a vários aspectos relacionados com as ajudas, pontuações, colaboração, competição, entre outros.

As opiniões de cada um dos grupos de utilizadores foram, de um modo geral, positivas e são apresentadas em separado nas próximas subsecções.

5.2.1 Opiniões dos professores

No que diz respeito à utilização de jogos no ensino em geral, como é possível ver no gráfico 5.4, 54% dos professores avaliaram a ideia como muito interessante, 31% como interessante e os restantes 15% foram neutros (pergunta 5). Este é de forma geral um resultado bastante positivo, dado que 85% desses professores nunca utilizou jogos no ensino e que os restantes 15% continuam a considerar a ideia de certo modo interessante (pergunta 6).

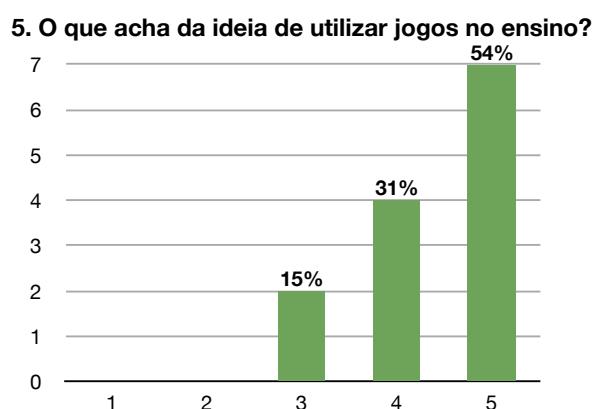


Figura 5.4: Interesse dos professores na utilização de jogos no ensino.

Por outro lado, a maioria desses professores (69%) referiu já ter utilizado questionários no passado como forma de atividade educativa (pergunta 8). Alguns desses questionários foram inclusivamente utilizados como forma de avaliação e foram criados utilizando na maioria dos casos o [Moodle](#) (pergunta 9).

Relativamente à utilização dos jogos de questionário do [Epik](#) (pergunta 10), como apresentado no gráfico 5.5, os Questionários Individuais foram considerados pela maioria (93%) mais interessantes que os Questionários Colaborativos (77%). Apesar de este resultado ser um pouco inesperado, é para ambos os casos bastante positivo, dado que a maioria nunca tinha utilizado jogos no ensino e porque todos referiram que utilizariam estes jogos no ensino, com exceção de um em relação aos Questionários Colaborativos (pergunta 11).

A preferência pelos Questionários Individuais, na nossa opinião, deve-se ao facto de os Questionários Colaborativos serem menos apropriados como atividades para avaliação, por não ser possível garantir em certos contextos (fora de aula) que os estudantes

10. O que pensa do uso de cada tipo de questionário no ensino?

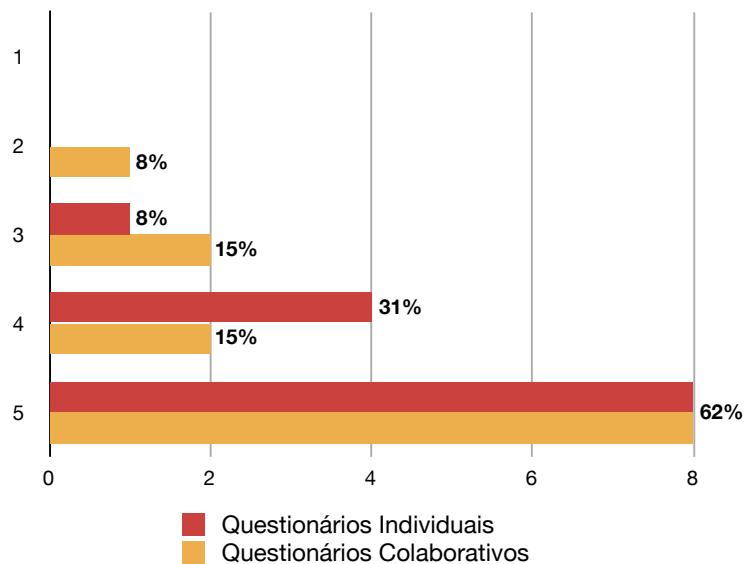


Figura 5.5: Interesse dos professores na utilização dos jogos de Questionário do Epik no ensino.

realizaram a atividade da forma esperada. Essa análise está de certo modo de acordo com os resultados obtidos acerca da utilização destes jogos como forma de avaliação. A maioria dos professores, neste caso 69%, referiu que utilizaria ocasionalmente os Questionários Individuais como forma de avaliação e os restantes 31% usariam sempre. Em comparação com os Questionários Colaborativos, 62% dos professores referiram que os utilizariam ocasionalmente como forma de avaliação, contudo houve apenas um que referiu que os utilizaria sempre e houve ainda 23% que afirmaram que nunca utilizariam (pergunta 12). Relativamente ao contexto, em média 85% indicaram que utilizariam esses jogos como atividades a realizar dentro de aula, e para ambos os casos 69% indicaram que poderiam também ser atividades a realizar fora de aula (perguntas 13 e 14). De um modo geral, esses são resultados positivos, no entanto segundo a nossa análise os Questionários Colaborativos seriam menos utilizados como forma de avaliação e na maioria dos casos em que fossem, provavelmente seria no contexto de uma aula.

Quanto aos vários tipos de ajudas que estes jogos disponibilizam (pergunta 15), as ajudas de consulta e de dicas foram consideradas por 84% e 69% dos professores como as mais interessantes, respetivamente. A ajuda de remoção de respostas, como apresentado no gráfico 5.6, foi a considerada menos interessante por 30% dos professores. A nosso ver, esses resultados devem-se ao facto de esta ajuda ser menos interessante no contexto do ensino porque não fornece nova informação aos estudantes, apenas lhes reduz o número de alternativas sem explicar o porquê de as respostas removidas estarem erradas.

Relativamente às características específicas dos Questionários Colaborativos (pergunta 16), em média 46% dos professores foram neutros em relação à colaboração nas ajudas de dicas e remoção, 61% consideraram interessante a coordenação entre jogadores, e 72%

15. Como se adaptam os seguintes tipos de ajudas dos Questionários Individuais ao ensino? - Ajuda 50/50

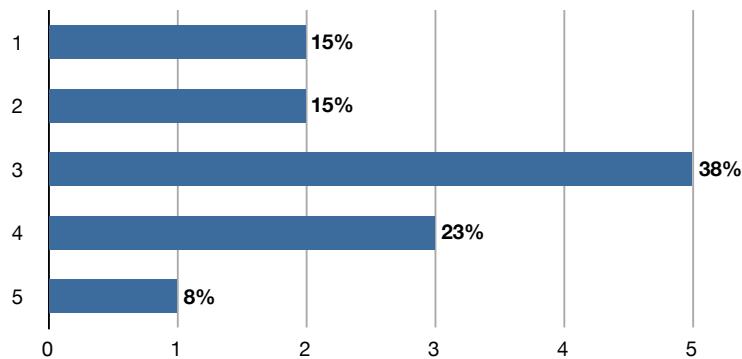


Figura 5.6: Interesse dos professores na ajuda de remoção de respostas.

consideraram também interessante a competição entre jogadores por atribuição de recompensas e penalizações. No caso particular da colaboração nas ajudas, além dos valores neutros, a de dicas foi considerada interessante por 46% e a de remoção foi considerada por 23% como desinteressante. Este último valor condiz com a forma como esta ajuda foi avaliada em termos do seu interesse no ensino, não sendo portanto de estranhar. Para os restantes casos houve ainda uma percentagem significativa de professores que foram neutros (em média 25%). Destes resultados, o mais inesperado, mas talvez compreensível, foi o facto de a competição ter sido considerada mais interessante que a colaboração. Isso poderá estar relacionado com o facto de a maioria destes professores não terem experimentado jogar os jogos do [Epik](#), o que os levou na maioria dos casos a dar respostas neutras ou negativas, por não compreenderem exatamente como este processo de colaboração se realiza. Por outro lado, o conceito de competição introduzido é-lhes já familiar tendo sido mais fácil de o avaliar sem experimentar.

17/18. Em geral, como avalia a utilização dos jogos de Questionário como uma atividade educativa?

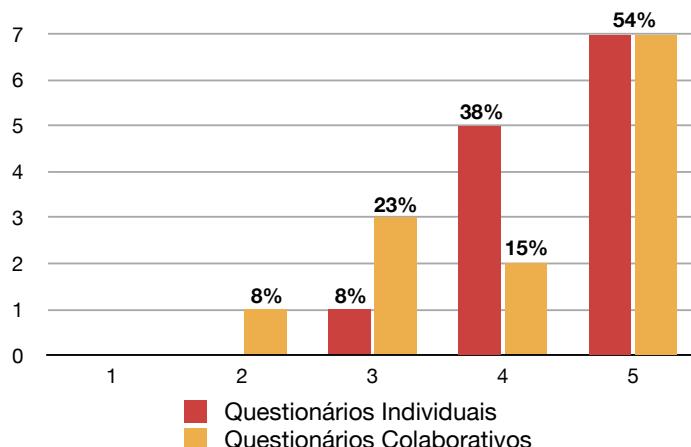


Figura 5.7: Interesse dos professores nos jogos de Questionário do Epik.

Independentemente dos resultados menos positivos obtidos para os Questionários Colaborativos, como apresentado no gráfico 5.7, estes foram considerados por 69% dos professores como interessantes, 23% foram neutros e 8% consideraram-nos desinteressantes (pergunta 18). Por outro lado, os Questionários Individuais foram considerados por 92% dos inquiridos como os mais interessantes e os restantes 8% foram neutros (pergunta 17).

5.2.2 Opiniões dos estudantes

No que diz respeito à utilização de jogos no ensino em geral (pergunta 5), como é possível ver no gráfico 5.8, a grande maioria dos estudantes mostrou-se bastante mais interessado que os professores, sendo que 76% avaliaram a ideia com nota máxima. No entanto, a maioria referiu nunca ter jogado jogos educativos anteriormente (pergunta 6), o que demonstra uma vez mais a falta de divulgação destes meios de aprendizagem, que dado o interesse demonstrado poderiam contribuir para um grande aumento da motivação para aprender. Dos que afirmaram já ter jogado jogos educativos, apenas alguns referiram quais, e entre eles encontravam-se outros jogos de questionário e os jogos geralmente disponibilizados em manuais escolares. Estes últimos necessitam geralmente que seja feito um investimento por parte dos estudantes para os obter dificultando portanto a sua distribuição.

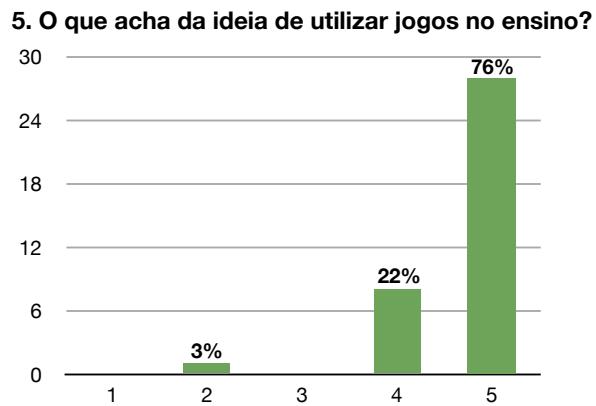


Figura 5.8: Interesse dos estudantes pela utilização de jogos no ensino.

Em contra partida, grande parte dos inquiridos, mais precisamente 78%, já tinham realizado questionários como atividade educativa no passado (pergunta 8) e, como apresentado no gráfico 5.9, avaliaram essa experiência de um modo geral como interessante (pergunta 9). No entanto, é de notar que poucos a consideraram como muito interessante comparativamente aos resultados obtidos acerca do uso dos jogos.

Relativamente à utilização dos jogos de questionário do Epik no ensino (pergunta 10), como apresentado no gráfico 5.10, 89% dos estudantes considerou os Questionários Colaborativos mais interessantes que os Questionários Individuais. Apesar de com menos destaque, os Questionários Individuais também foram considerados interessantes, mas



Figura 5.9: Interesse dos estudantes pela utilização de atividades de questionário no ensino.

apenas por 73% e os restantes 27% foram neutros. Esta preferência pelos Questionários Colaborativos em comparação com os Questionários Individuais é um resultado bastante positivo dado que o nosso objetivo era precisamente fornecer formas de aprendizagem aos estudantes que fossem mais interessantes por oferecerem meios de interação e colaboração. No entanto, este resultado está em desacordo com o obtido para os professores acerca destes dois tipos de jogos.

10. O que pensa do uso de cada um desses tipos de jogos no ensino?

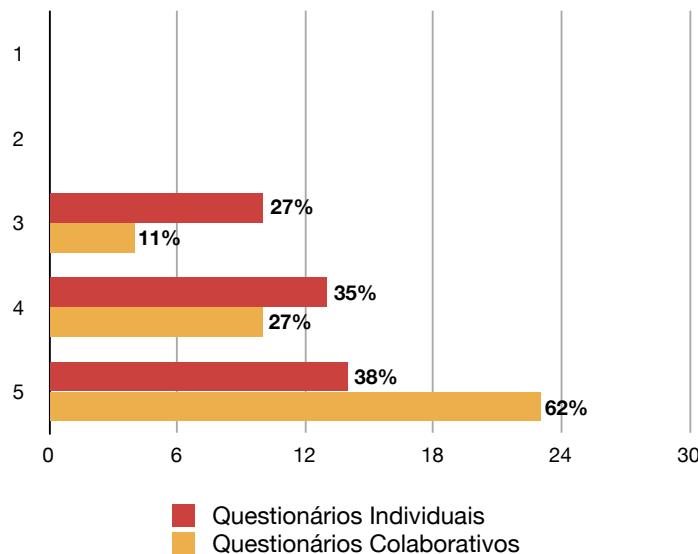


Figura 5.10: Interesse dos estudantes pela utilização dos jogos de Questionário do Epik no ensino.

Ainda relativamente à utilização desses jogos mas como forma de avaliação (pergunta 11), houve para ambos os casos um número significativo de estudantes que se opuseram à ideia. No entanto, de um modo geral a maioria considerou a ideia interessante. A maior percentagem de valores negativos em relação a esta questão foram obtidos para

os Questionários Colaborativos, o que poderá fazer sentido dado que são uma atividade em que os estudantes tanto podem ser avaliados individualmente ou como grupo, e em que o seu desempenho poderá estar dependente das ajudas fornecidas por outros, algo que já não acontece nos Questionários Individuais. Este resultado está de certo modo de acordo com o obtido por parte dos professores, visto que alguns destes últimos referiram que nunca os usariam como forma de avaliação.

Quanto às características dos jogos de questionário do [Epik](#), a sua avaliação foi também de um modo geral positiva. A todos os tipos de ajudas foi atribuída nota máxima (pergunta 12), sendo a que mais se destaca a de dicas, considerada por 51% dos inquiridos como muito interessante, 38% como interessante e os restantes 11% foram neutros. Para as outras ajudas os resultados foram idênticos apesar de se terem verificado alguns resultados menos positivos, que no nosso entender se devem ao facto de essas ajudas em comparação com a de dicas serem menos interessantes a nível de aprendizagem. Isto porque a ajuda de remoção apenas contribui para uma redução do número de hipóteses sem fornecer nova informação ao jogador, e a de consulta poderá por vezes não ser tão direta como a de dicas dada a quantidade de informação que pode estar presente nos conteúdos apresentados. De um modo geral, estes resultados condizem com os obtidos para os professores, dado que estes também consideraram a ajuda de remoção de respostas a menos interessante.

Os meios de colaboração presentes nos Questionários Colaborativos, como a colaboração nas ajudas e a coordenação entre jogadores, foram também considerados interessantes (pergunta 13). Em particular a colaboração nas ajudas foi a que obteve resultados mais positivos, pois 43% dos inquiridos avaliaram-na como interessante e 38% como muito interessante. Além disso, a componente competitiva destes jogos foi também considerada interessante pelos inquiridos (pergunta 14). Neste caso os resultados divergem já um pouco dos obtidos para os professores, principalmente no que diz respeito à colaboração, o que como referimos poderá estar relacionado com o facto de estes na sua maioria não terem experimentado os jogos.

De um modo geral, como ilustrado no gráfico 5.11, a maioria dos estudantes avaliaram ainda estes jogos como adequados para o processo de aprendizagem (pergunta 15). O que é um resultado de algum modo semelhante com o obtido para os professores.

5.3 Desenvolvimento de jogos no Epik

Para o processo de desenvolvimento de jogos no [Epik](#), foi solicitado aos professores ou instrutores que realizassem várias tarefas como criar ou importar atividades e recursos, criar um projeto, criar cenários, inserir conteúdos nesses cenários, parametrizar as propriedades gerais e o fluxo de um jogo, e gerar um jogo. De forma a facilitar a realização das tarefas de importação, foram fornecidas as credências de um utilizador de um sistema [Moodle](#) com acesso a várias atividades e recursos. Do grupo de inquiridos, 54% utilizou o Google Chrome para aceder ao [Epik](#) e realizar essas tarefas, e os restantes 46%

15. De modo geral, considera estes jogos bons para o processo de aprendizagem?

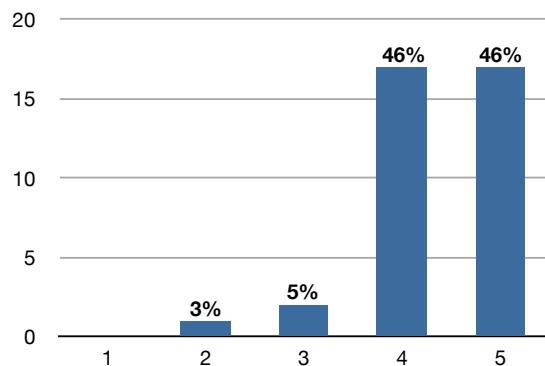


Figura 5.11: Avaliação geral dos jogos do Epik como atividade educativa por parte dos estudantes.

utilizaram o Mozilla Firefox.

Nesta avaliação começámos por averiguar a experiência dos utilizadores com outras ferramentas de desenvolvimento de jogos (perguntas 20 e 21). Segundo os resultados obtidos, 77% dos inquiridos nunca tinha utilizado ferramentas deste género. Os restantes referiram já ter utilizado ferramentas como o Alice [Car12], WebQuest³, Hot Potatoes⁴, e linguagens como o HTML5, PHP e ASP. Em termos de satisfação com essas ferramentas a experiência com o Alice foi considerada como muito boa, com as linguagens boa e com as restantes neutra. No entanto, essas ferramentas não possibilitam a criação de atividades educativas muito interativas nem com meios de colaboração. Além disso, as linguagens referidas, apesar de possibilitarem a criação de jogos, requerem obviamente conhecimentos de programação, não sendo portanto adequadas para a maioria dos professores.

De seguida, foram realizadas várias questões relacionadas com a facilidade de utilização das áreas de interface de utilizador do *Epik* (*dashboard* e ambiente de desenvolvimento) e da plataforma *Epik* em geral. Os resultados obtidos sobre cada um desses aspectos foram, de um modo geral, positivos e são apresentados nas próximas subsecções, respetivamente.

5.3.1 Satisfação com a *dashboard*

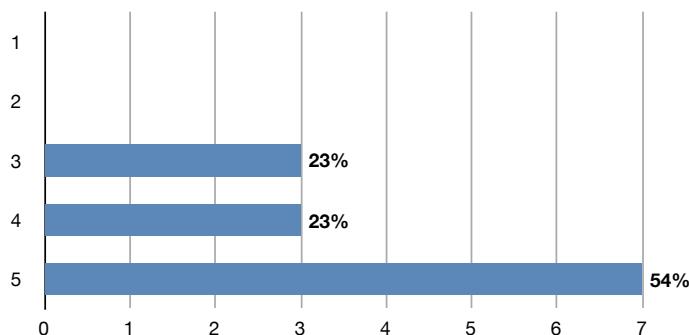
Na generalidade, 62% dos utilizadores consideraram a nomenclatura utilizada na *dashboard* como intuitiva e os restantes foram neutros (pergunta 23). Relativamente à realização de tarefas nessa área, como a criação ou importação de atividades e recursos, criação de projetos e associação de dicas e recursos a atividades, os resultados obtidos foram de modo geral positivos (pergunta 24). As tarefas que mais se destacaram, por ordem de preferência, foram a criação de projetos (Figura 5.12a), recursos (Figura 5.12b) e atividades (Figura 5.12c), as quais foram consideradas por aproximadamente 60% dos inquiridos

³<http://webquest.org/> consultado em 20-03-2013.

⁴http://hotpot.uvic.ca/v6_faq.php consultado em 20-03-2013.

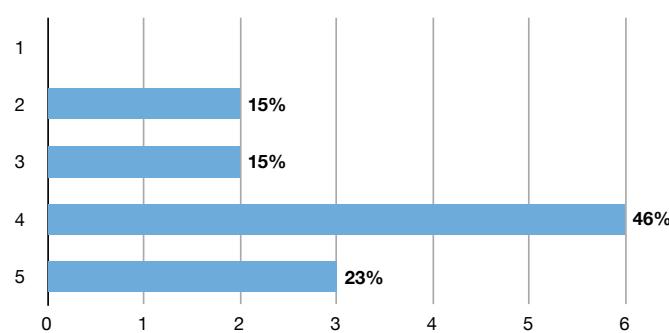
como muito fáceis. No entanto, houve ainda uma percentagem significativa de utilizadores que atribuiu nota neutra a essas tarefas (em média 23%), o que a nosso ver significa que consideraram os passos a realizar normais, ou que não experimentaram.

24. Como classifica a realização das seguintes tarefas? - Criar projetos



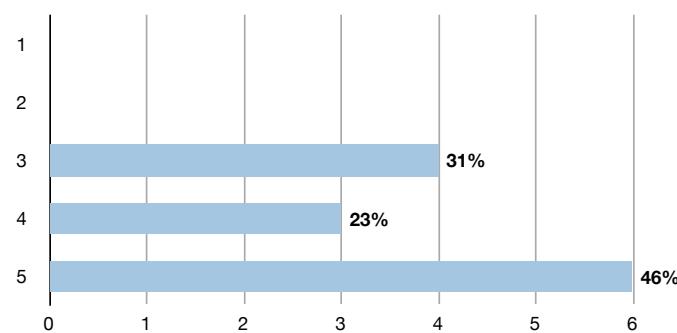
(a) Criação de projeto

24. Como classifica a realização das seguintes tarefas? - Criar recursos



(b) Criação de recurso

24. Como classifica a realização das seguintes tarefas? - Criar atividades



(c) Criação de atividade

Figura 5.12: Facilidade na realização de tarefas de criação no Epik.

No que diz respeito à possibilidade de importar conteúdos do [Moodle](#), como apresentado no gráfico 5.13, praticamente todos os inquiridos consideraram a ideia interessante, ou muito interessante (pergunta 25). Do mesmo modo, os *templates* de projetos foram também considerados úteis ou muito úteis pela quase totalidade dos inquiridos (92%),

como pode ser observado no gráfico 5.14 (pergunta 26).

25. Considera útil a possibilidade de importar atividades ou recursos de um sistema Moodle?

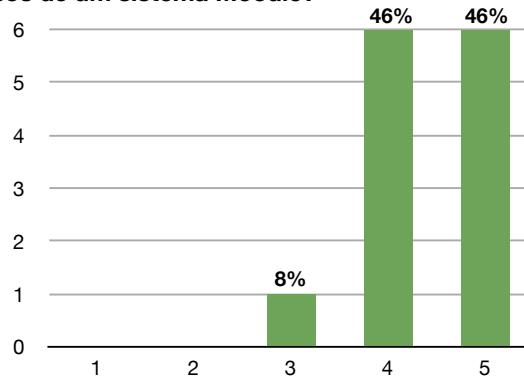


Figura 5.13: Interesse da possibilidade de importar conteúdos do Moodle.

26. Considera os templates de projetos úteis para o desenvolvimento de novos jogos?

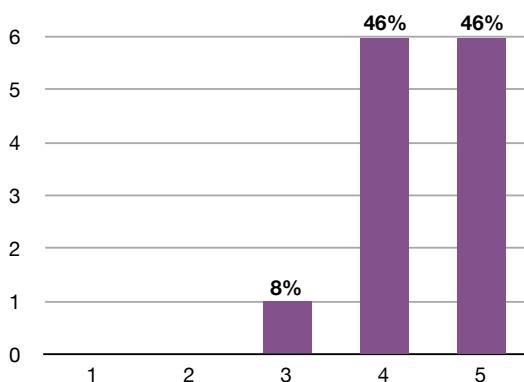


Figura 5.14: Utilidade dos *templates* de projetos do Epik.

Relativamente à descrição dos jogos de Questionário do Epik apresentada aquando da criação de um projeto, 54% dos inquiridos considerou-a fácil de entender e os restantes foram neutros, tendo havido ainda um que considerou difícil (pergunta 27). Segundo a nossa análise, pensamos que quem atribuiu um valor neutro a esta questão foi porque não reparou na descrição, seja como for, seria importante ter resultados mais positivos nesta questão dado que é importante que a descrição dos jogos seja percepível. Para isso, visto que os inquiridos não deixaram comentários, será necessário averiguar as dificuldades que encontraram ao ler essas descrições.

No geral, como apresentado no gráfico 5.15, a *dashboard* em termos de utilização foi considerada pela maioria dos inquiridos como prática ou muito prática (pergunta 28). Contudo, houve ainda alguns que deram nota neutra a esta questão, o que se relaciona de certo modo com os resultados obtidos acerca desta área da interface de utilizador.

28. De uma forma geral, como classifica a dashboard em termos de utilização?

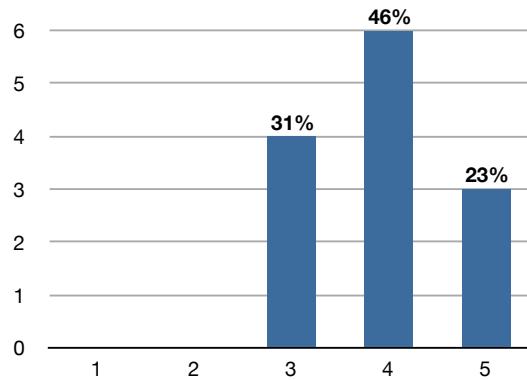


Figura 5.15: Nível de satisfação com a *dashboard* de um modo geral.

5.3.2 Satisfação com o ambiente de desenvolvimento

Depois das questões relacionadas com a *dashboard*, foram então realizadas várias perguntas sobre a utilização do ambiente de desenvolvimento para a criação de um jogo. Para esse processo foi primeiro avaliado o nível de satisfação dos utilizadores com a facilidade de realização de tarefas de criação de cenários e de inserção de textos, formas geométricas, recursos e atividades (pergunta 29). Os resultados obtidos foram de um modo geral positivos (em média 71%) dado que para todas essas operações a maioria considerou-as fáceis ou muito fáceis de realizar. Quanto à criação de cenários a partir de um *template*, como pode ser visto no gráfico 5.16, 77% dos inquiridos considerou a existência desses *templates* muito útil (pergunta 30).

30. Considera a existência de templates de cenários útil para o desenvolvimento de um jogo?

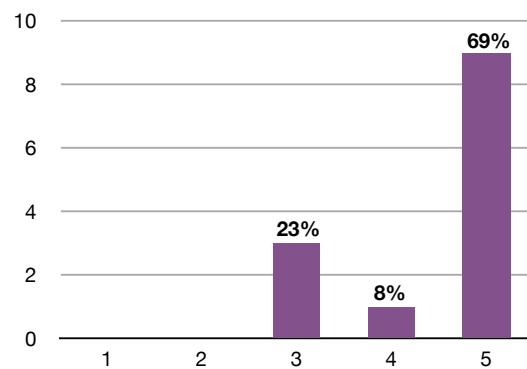


Figura 5.16: Utilidade dos *templates* de cenários.

Relativamente à utilização do painel de propriedades do ambiente de desenvolvimento, foi avaliado o nível de facilidade da realização das ações de parametrização de estilos, fluxo do jogo, bónus de cenários, pontos e ajudas de atividades e registos. No

geral, todas essas ações foram avaliadas pela maioria (em média 66%) como uma tarefa fácil ou muito fácil (perguntas 31, 32, 33, 34 e 35). Porém, para todas elas houve ainda um número significativo de utilizadores que deu nota neutra (em média 29%), o que poderá significar que ou não realizaram a ação em questão ou consideraram-na banal.

De um modo geral, como apresentado no gráfico 5.17, a organização do ambiente de desenvolvimento foi avaliada por 69% dos inquiridos como intuitiva (pergunta 36). Este resultado está de certo modo de acordo com os obtidos para as restantes perguntas relacionadas com este ambiente, dado que uma vez mais existe uma percentagem de utilizadores que deram avaliação neutra.

36. No geral, como avalia a organização deste ambiente de desenvolvimento de jogos?

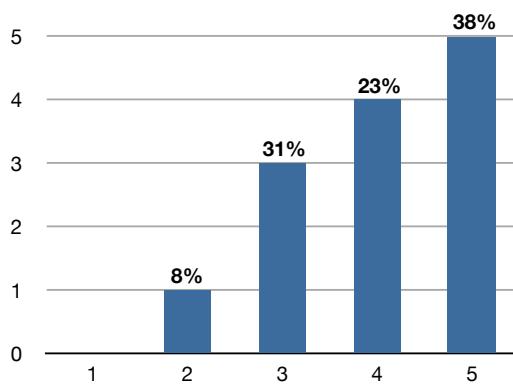


Figura 5.17: Organização do ambiente de desenvolvimento do Epik.

Do mesmo modo, como apresentado no gráfico 5.18, a facilidade de desenvolvimento de um jogo no Epik foi avaliada por 46% dos inquiridos como muito fácil e por 31% como fácil (pergunta 37). Este é um resultado bastante positivo tendo em conta que houve um número significativo de utilizadores a avaliar a realização de várias ações de forma neutra e que grande parte deles nunca tinha desenvolvido um jogo. Além disso, dois dos utilizadores que já tinham utilizado ferramentas deste género foram uns dos que contribuíram para esse resultado.

No que diz respeito à facilidade do processo de geração de um jogo os resultados obtidos foram os mesmos que para o caso apresentado no gráfico 5.17. Além disso, 77% dos inquiridos referiu não ter tido problemas na geração de um jogo e os restantes 23% consideraram esclarecedora a descrição dos erros que obtiveram no final desse processo de geração.

5.3.3 Satisfação com a plataforma Epik

Relativamente à caracterização do Epik (pergunta 42), como apresentado no gráfico 5.19, a maioria dos inquiridos escolheu palavras como agradável, inovador, útil, prático e simples. Além dessas, alguns caracterizaram-no ainda como intuitivo, motivante, confuso,

37. No geral, como classifica o desenvolvimento de um questionário no Epik?

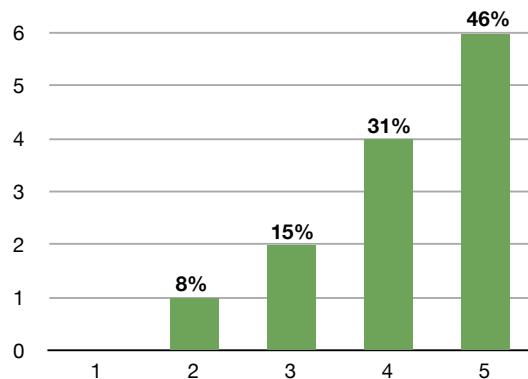


Figura 5.18: Facilidade do processo de desenvolvimento de um jogo no Epik.

frustrante e pouco intuitivo. Estes últimos indicam que é ainda necessário realizar algumas melhorias que permitam aos utilizadores sentir-se mais familiarizados com a interface de utilizador. Segundo os resultados obtidos na pergunta 43, essas melhorias deverão incidir principalmente na terminologia utilizada e na simplicidade de utilização, este último aspeto, como é possível ver no gráfico 5.20, foi dos três avaliados nesta questão o que teve resultados mais negativos. O que obteve resultados mais positivos foi o aspeto gráfico da interface de utilizador, com o qual 76% dos inquiridos ficaram satisfeitos.

Por fim, como apresentado no gráfico 5.21, 92% dos inquiridos referiu que recomendaria o [Epik](#) a outros amigos ou colegas (pergunta 44), o que corresponde basicamente a todos os que participaram nesta avaliação com exceção de um. A nosso ver isto significa que apesar das melhorias necessárias a maioria dos inquiridos ficou bastante interessada neste conceito e gostou da experiência.

5.4 Jogabilidade dos jogos do Epik

Para avaliar a jogabilidade dos jogos do [Epik](#) foram por nós desenvolvidos quatro jogos sobre diferentes temáticas que foram disponibilizados aos estudantes ou ex-estudantes. Desses jogos, como referido anteriormente, foi solicitado aos participantes que jogassem pelo menos um de cada tipo de questionário. Para o fazerem, 78% dos participantes utilizou o Google Chrome, 16% o Mozilla Firefox e os restantes 6% utilizaram o Safari ou o Internet Explorer. Os jogos disponibilizados foram os seguintes:

- **General Knowledge IQuiz**⁵ – é um Questionário Individual com perguntas de cultura geral sobre o sistema solar, a história da Terra, a história de Portugal e ciência em geral. O jogo é composto por 3 cenários de conceitos e 4 de atividades. Em termos de fluxo, os cenários de conceitos têm sempre um botão para continuar e os

⁵<http://epik.di.fct.unl.pt/epik/games/play/1> consultado em 22-03-2013.

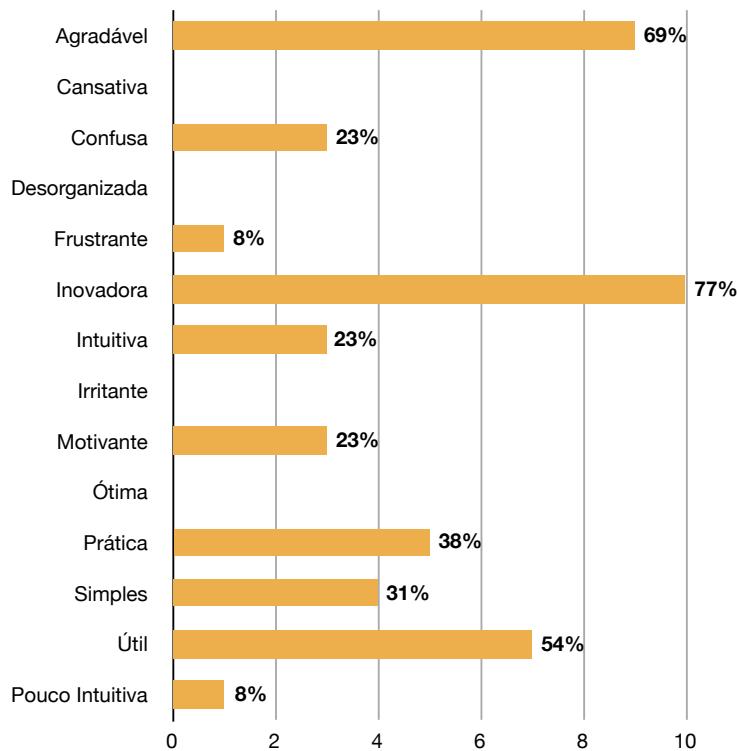
42. Das palavras abaixo quais na sua opinião melhor caracterizam o Epik?

Figura 5.19: Palavras caracterizadoras do Epik.

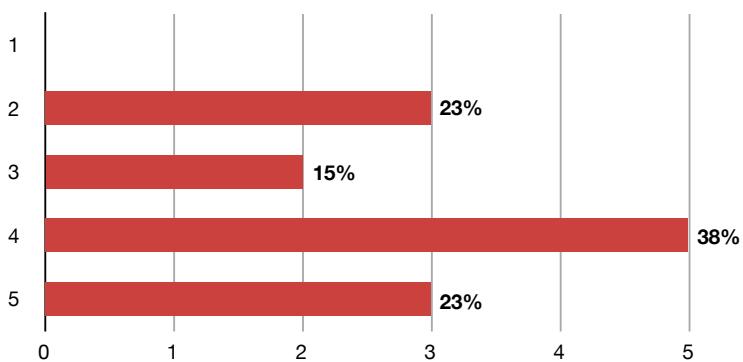
43. Indique o seu grau de satisfação relativamente aos seguintes aspetos do Epik. - Simplicidade de utilização

Figura 5.20: Grau de satisfação com a simplicidade de utilização do Epik.

44. Recomendaria o Epik a outros amigos/colegas?

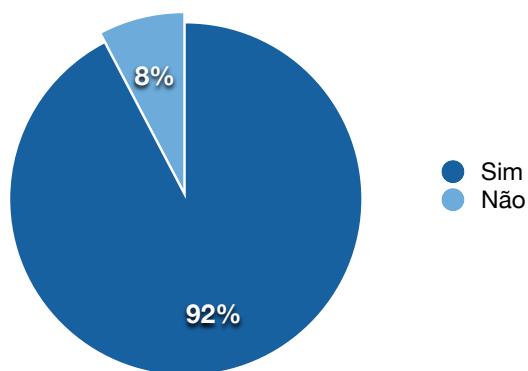


Figura 5.21: Quantidade de pessoas que recomendariam o Epik a outros amigos/colegas.

de atividades têm um tempo limite de permanência que nunca será atingido se as atividades forem todas terminadas. No total o jogo tem 10 atividades, todas elas questões de escolha múltipla com exceção de uma que é de verdadeiro ou falso. Todas essas atividades disponibilizam ajudas;

- **Abstract Data Types IQuiz⁶/CQuiz⁷** – são um Questionário Individual e Questionário Colaborativo, respectivamente, com questões dirigidas a estudantes de informática sobre tipos abstratos de dados (TAD). Os TAD abordados nestes jogos são a Lista, a Fila, a Pilha e a Árvore (mais precisamente as árvores binárias). Ambos são compostos por 4 cenários de conceitos e 6 de atividades. Em termos de fluxo, estes jogos disponibilizam sempre também um botão para continuar em cenários de conceitos e nos de atividades um tempo limite que poderá não ser atingido se as atividades forem todas terminadas. Num dos cenários de atividades do jogo existe ainda um botão para saltar para um outro cenário de atividades que só é acessível nessa situação. No total ambos os jogos são compostos por 17 atividades, encontrando-se entre elas alguns grupos de questões. O jogo contém questões de todos os tipos e para a maioria delas são disponibilizadas ajudas;
- **Math 12 CQuiz⁸** – é um Questionário Colaborativo com perguntas de matemática para estudantes a partir do 12º ano de escolaridade. Os principais temas abordados por este jogo são funções, derivadas e números complexos. O jogo é composto por 9 cenários de conceitos e 5 de atividades. O fluxo do jogo é idêntico ao dos anteriores e existe também um cenário de conceitos que é apenas acessível se algum jogador escolher saltar no último cenário de atividades. No total o jogo tem 23 atividades, encontrando-se entre elas alguns grupos de questões. O jogo contém questões de todos os tipos e para algumas delas são disponibilizadas ajudas;

⁶<http://epik.di.fct.unl.pt/epik/games/play/2>. consultado em 22-03-2013

⁷<http://epik.di.fct.unl.pt/epik/games/play/3>. consultado em 22-03-2013

⁸<http://epik.di.fct.unl.pt/epik/games/play/4>. consultado em 22-03-2013

O objetivo desta avaliação era obter a opinião dos jogadores relativamente à facilidade de entendimento do fluxo dos jogos, da qualidade da informação que estes disponibilizam e do seu aspetto gráfico, e também perceber se estes se divertiam e se sentiam motivados para aprender jogando os jogos do **Epik**. Os resultados obtidos foram de um modo geral bastante positivos e alguns dos jogadores deixaram ainda algumas sugestões. Esses resultados e sugestões são apresentados ao longo das próximas subsecções.

5.4.1 Satisfação com a aparência e fluidez dos jogos

Em termos da nomenclatura e dos ícones utilizados nos jogos de questionário do **Epik** a grande maioria dos inquiridos, em média 84%, considerou-os intuitivos (perguntas 16 e 17).

Relativamente ao painel de jogadores, como apresentado no gráfico 5.22, 81% dos inquiridos consideraram-no como uma forma interessante de percepção do estado do jogo (pergunta 18). Do mesmo modo, 79% considerou a informação apresentada por esse painel relativamente intuitiva (pergunta 19). Segundo os comentários de alguns jogadores a informação deste painel poderia ser mais intuitiva se fosse possível consultar durante o jogo o que significam os vários tipos de pontuações e o que significa o formato k/n para apresentação do número de ajudas disponíveis num cenário e no total. Além disso, ainda que tendo uma avaliação positiva no que diz respeito à competição nos jogos (68%), alguns consideraram os pontos pouco informativos e referiram que uma escala de desempenho poderia ser mais adequada.

18. Considera interessante o uso do painel de jogadores como uma forma de percepção do estado do jogo?

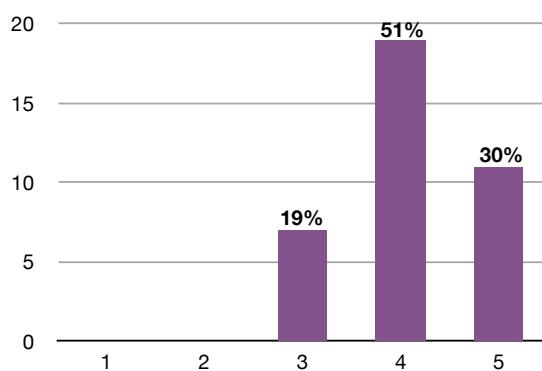


Figura 5.22: Interesse do painel de jogadores como forma de percepção do estado do jogo.

No que diz respeito aos sons utilizados como música de fundo e notificações de resposta correta, incorreta e pedido ou receção de ajuda, a maioria dos inquiridos considerou-os como adequados (pergunta 20). De um modo geral, o som de notificação de resposta correta foi o que obteve os resultados mais positivos, 93% consideraram-no adequado, e a música de fundo foi o que teve resultados mais negativos, 10% consideraram-no desadequado e 31% foram neutros. No entanto, esta última foi considerada ainda por 58% dos

inquiridos, como adequada por ser relaxante, convidativa e se enquadrar num ambiente de aprendizagem.

Relativamente aos métodos utilizados para responder a perguntas (pergunta 21), a maioria dos inquiridos considerou-os muito práticos por bastar escolher, ou fornecer a sua resposta, sem ser necessário confirmá-la antes de a enviar. Contudo, os resultados foram menos positivos para as perguntas de resposta curta porque os jogadores preferiam ser eles a confirmar a resposta, em vez de esta ser enviada ao fim de algum tempo.

O fluir do jogo, como é ilustrado no gráfico 5.23, foi também considerado simples de entender, neste caso por 95% dos participantes (pergunta 22). Os restantes 5% atribuíram uma nota menos positiva porque, tal como alguns da maioria, consideraram confuso que o jogo mudasse de cenários sem informar de que isso iria acontecer. Alguns sugeriram inclusivamente que o jogo informa-se alguns segundos antes de que o tempo para responder ia acabar, ou que fosse fornecido um botão para continuar depois de as atividades serem terminadas.

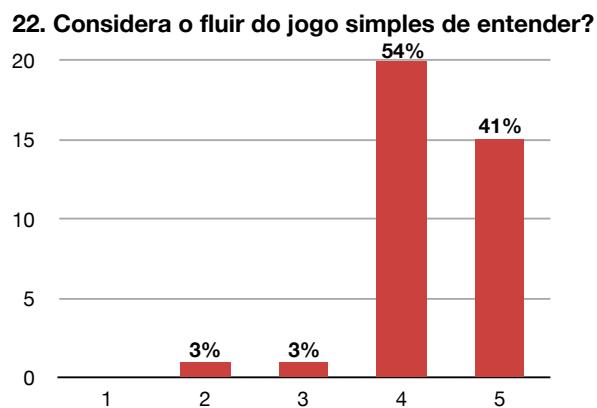


Figura 5.23: Simplicidade do fluir dos jogos de Questionário do Epik.

Quanto à informação apresentada aos jogadores na janela de pedido ou receção de ajuda, esta foi considerada como intuitiva em média por 84% dos inquiridos (pergunta 23). Porém, houve ainda alguns resultados menos positivos para ambos os casos por considerarem a informação apresentada nestas janelas pouco direta. Alguns dos inquiridos, mesmo os que atribuíram nota positiva, referiram que essa informação poderia ser mais clara se indicasse quem está a pedir ajuda ou quem respondeu ao pedido, e se a pergunta correspondente fosse identificada de uma forma mais visível. No que diz respeito ao método de resposta a um pedido de ajuda, para o caso dos Questionários Colaborativos (pergunta 24), 70% dos inquiridos considerou-o prático por bastar selecionar os valores a enviar sem ser necessário ainda confirmar.

De um modo geral, a grande maioria dos inquiridos, como apresentado no gráfico 5.24, consideraram a sua experiência com estes jogos interessante (54%) ou muito interessante (38%). Este é um resultado bastante positivo, pois apesar das notas menos positivas a algumas das características referidas anteriormente, a grande maioria dos jogadores

gostou da experiência.

25. No geral, como classifica a sua experiência com estes jogos?

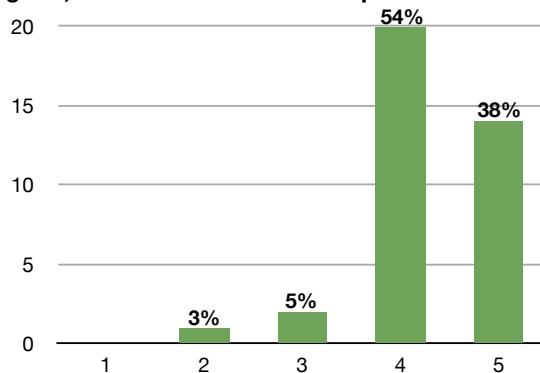


Figura 5.24: Satisfação com os jogos do Epik como atividades educativas.

5.4.2 Satisfação com os jogos do Epik

Os jogos do [Epik](#), como apresentado no gráfico 5.25, foram caracterizados pela maioria dos inquiridos como agradáveis, úteis, divertidos, inovadores, intuitivos e motivantes. Todos esses aspectos são bastante positivos e indicam que grande parte dos nossos objetivos foram atingidos. A diversão é talvez dos referidos aquele que mais se destaca dado que incluir essa componente em jogos educativos não é geralmente uma tarefa fácil. Apesar desses resultados, houve ainda uma minoria que considerou os jogos aborrecidos, cansativos e confusos, o que significa que há obviamente melhorias que têm de ser feitas, algumas das quais foram já referidas.

No que diz respeito à avaliação geral do aspetto gráfico, da terminologia utilizada e da fluidez dos jogos a grande maioria dos inquiridos (em média 90%), demonstrou estar uma vez mais satisfeita. Além disso, como é possível ver no gráfico 5.26, 97% dos inquiridos referiu que recomendaria estes jogos a outros amigos ou colegas (pergunta 28), o que é uma vez mais um resultado muito positivo e que indica a satisfação de todos com os jogos, independentemente dos problemas que encontraram.

5.4.3 Sugestões por parte dos jogadores

A maioria dos inquiridos, na pergunta 29, deixou ainda alguns comentários, críticas e sugestões acerca dos jogos do [Epik](#). Na lista abaixo são apresentados os que considerámos mais interessantes e relevantes para este trabalho:

- “Os ícones das ajudas deviam ser maiores e deviam estar mais visíveis”;
- “Quando termino as perguntas de um cenário gostaria de ser eu a decidir quando passo para o próximo”;
- “Nas questões de resposta curta deveria ser necessário submeter a resposta em vez de esta ser submetida ao fim de algum tempo”;

26. Das palavras abaixo quais na sua opinião melhor caracterizam os jogos de Questionário do Epik?

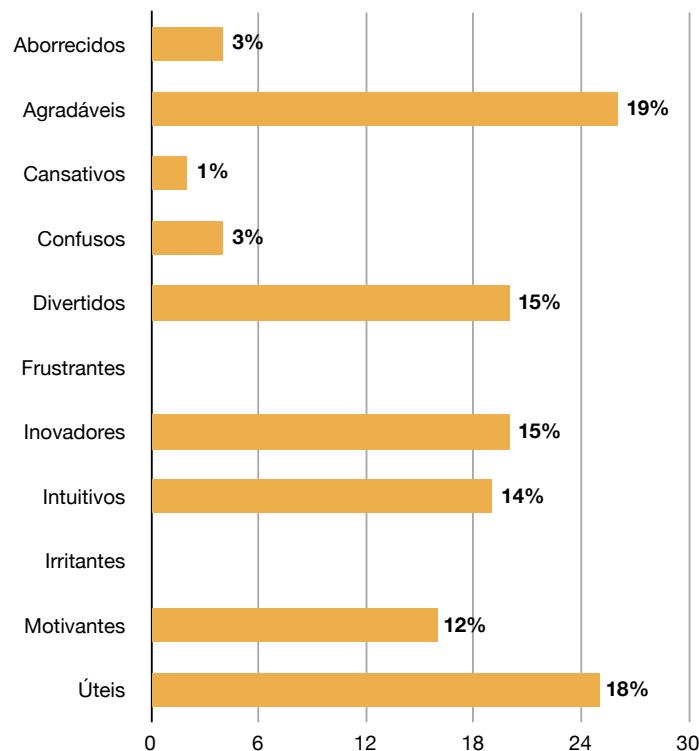


Figura 5.25: Palavras caracterizadoras dos jogos do Epik.

28. Recomendaria estes jogos a outros amigos/colegas?

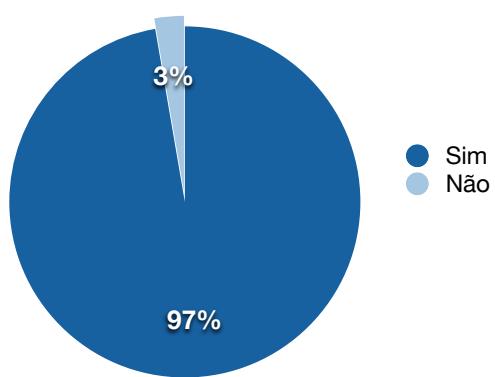


Figura 5.26: Estudantes que recomendariam os jogos do Epik a outros amigos/colegas.

- “A mensagem na janela de pedido de ajuda deveria ser mais clara indicando que o jogador x está a pedir ajuda”;
- “Gostaria de ter uma forma de saber quantos cenários faltam para terminar o jogo”;
- “A resolução do ecrã de jogo não se adequa à dimensão de monitores mais pequenos”;
- “Deveria existir a opção de pausa nos jogos de Questionário Individual”;
- “Penso que o jogador deveria ser avisado de que o tempo está quase a terminar em vez de mudar de cenário de uma forma brusca”;
- “Além das pontuações penso que seria ideal ter uma escala de pontuações que permita ao jogador saber se está ou não a ter um bom desempenho, pois os pontos apresentados não proporcionam essa informação”.

Além desses comentários, alguns jogadores referiram ainda que o *design* dos jogos necessita de ser melhorado, que gostariam de ter a possibilidade de desligar os sons e que estes últimos deveriam reduzir de volume quando se está a ver um vídeo do Youtube⁹ ou Vimeo¹⁰.

Todos esses comentários levantam questões pertinentes e válidas que pensamos que deverão ser consideradas em trabalho realizado no futuro, de forma a melhorar a experiência com estes jogos.

5.5 Utilização do tema Cubic

O tema Cubic para o [Moodle](#) foi avaliado em termos da satisfação com a interface de utilizador e das principais funcionalidades que oferece de novo, como os sistemas de notificações, aplicações, *chat* global, entre outros. Para a realização desta avaliação foi fornecido aos vários participantes um utilizador sem permissões de edição de um curso ou do sistema e foi-lhes pedido que realizassem várias tarefas como interagir com os menus da barra de utilizador, integrar o [Epik](#) no [Moodle](#) criando e acedendo a uma aplicação e utilizar o *chat* global. Para a realização deste teste, 70% dos inquiridos utilizou o Google Chrome para aceder ao nosso [Moodle](#), 30% utilizou o Mozilla Firefox e os restantes 20% utilizaram o Safari ou Internet Explorer. Além disso, 90% dos inquiridos já tinham usado [Moodle](#) anteriormente, havendo apenas um que o estava a utilizar pela primeira vez.

Os resultados obtidos neste processo de avaliação, apesar de baseados nas respostas de uma pequena população de utilizadores, foram, de um modo geral, bastante positivos. Esses resultados são apresentados em maior detalhe nas próximas subsecções.

⁹<http://www.youtube.com/> consultado em 22-03-2013

¹⁰<https://vimeo.com/> consultado em 22-03-2013

5.5.1 Satisfação com as funcionalidades do tema e do *chat*

O primeiro conjunto de perguntas realizadas aos utilizadores neste processo de avaliação diziam respeito à barra de utilizador do Cubic e às funcionalidades que esta oferece. De um modo geral, como pode ser visto no gráfico 5.27, 50% dos utilizadores consideraram a informação disponibilizada por esta barra útil e 40% muito útil (pergunta 7). No que diz respeito à utilidade dos menus para consulta das instituições, atividades, notificações e eventos do utilizador (perguntas 8, 9, 10, 11), a maioria considerou-os muito úteis, práticos e intuitivos (em média 87%). Relativamente à possibilidade de integrar aplicações no Moodle, como pode ser visto no gráfico 5.28, 80% dos inquiridos consideraram a ideia muito útil e os restantes 20% como útil (perguntas 14 e 15).

7. Considera útil a informação disponibilizada pelos menus da barra de utilizador?

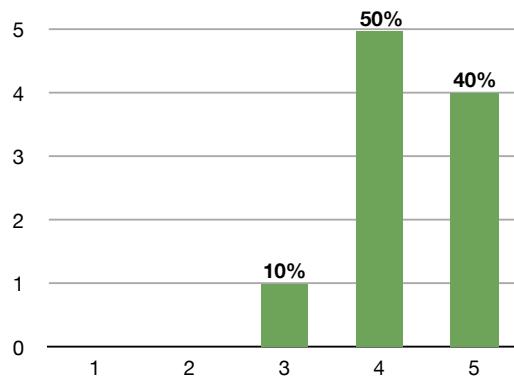


Figura 5.27: Utilidade da barra de utilizador do Cubic.

14. Considera a possibilidade de integrar aplicações no Moodle útil?

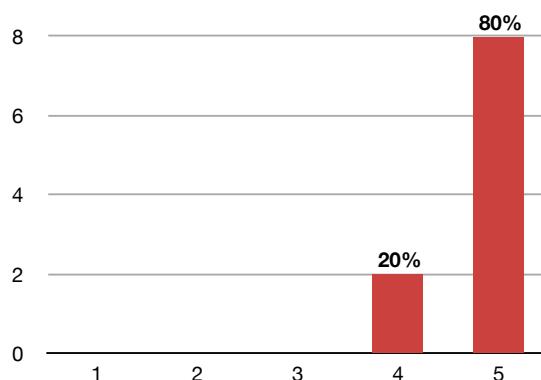


Figura 5.28: Utilidade do conceito de aplicações no Moodle.

Quanto ao *chat* global, tal como para o caso anterior, todos os utilizadores o consideraram útil ou muito útil e indicaram que este oferece uma forma mais prática, simples e familiar de comunicar ou colaborar com outros colegas (perguntas 18 e 20). Esse é um

resultado esperado e que revela a necessidade deste tipo de ferramentas nos LMS. No entanto, a maioria (90%) considerou que é necessário ter a possibilidade de definir quando é que este *chat* está disponível aos estudantes (pergunta 19). Por exemplo, só estar acessível fora do período de aulas ou fora do contexto de uma atividade individual.

5.5.2 Satisfação geral

Este tema, como apresentado no gráfico 5.29, foi caracterizado pela maioria dos inquiridos como agradável, simples, útil, prático e intuitivo (pergunta 21). Apesar de menos significativos, alguns classificaram-no ainda como inovador e confuso. No entanto, de um modo geral, consideramos que foi uma caracterização bastante positiva dado que pretendíamos precisamente fornecer aos utilizadores deste LMS uma forma mais agradável, prática, simples e útil de o utilizar.

21. Das palavras abaixo quais é que na sua opinião melhor caracterizam o tema Cubic?

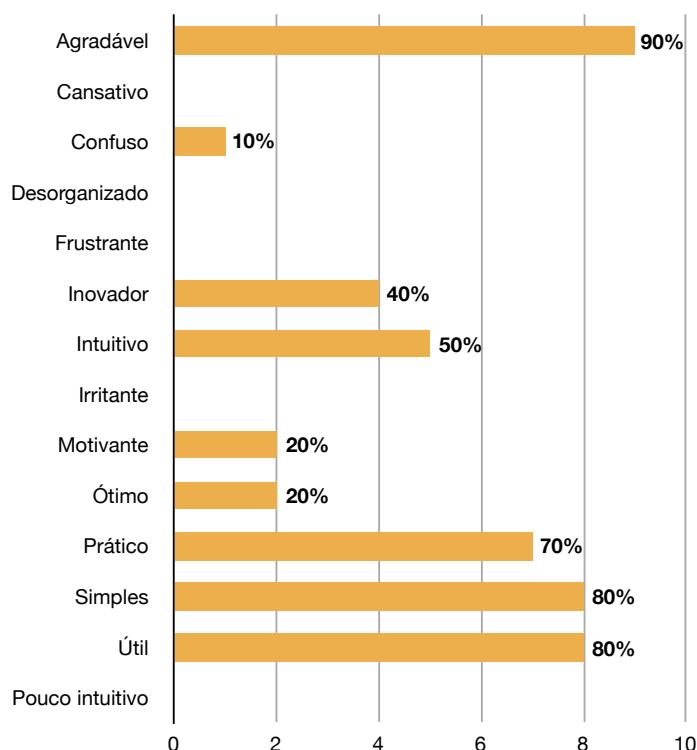


Figura 5.29: Palavras caracterizadoras do tema Cubic.

Em termos do aspeto gráfico do tema, da terminologia utilizada e da simplicidade de utilização, em média 90% mostrou-se satisfeito (pergunta 22), tendo sido a terminologia utilizada a que teve mais resultados neutros (20%). Além disso, todos os inquiridos indicaram que recomendariam este tema a outro colegas ou amigos (pergunta 23) e avaliaram o tema, de um modo geral, como bom ou muito bom (pergunta 24).

5.6 Conclusões

Os resultados obtidos neste processo de avaliação foram, de um modo geral, muito positivos e vão de um certo modo ao encontro dos nossos objetivos. Como foi visto, os grupos de professores e estudantes demonstraram ter interesse nos jogos de questionário do [Epik](#). No entanto, consideraram que estes jogos, principalmente os Questionários Colaborativos, não são muito apropriados como forma de avaliação. Do mesmo modo, ambos os grupos consideraram que a ajuda de remoção de respostas não é muito interessante no contexto do ensino. Os professores mostraram também preferir os Questionários Individuais, ao contrário do obtido para os estudantes, contudo pela nossa análise consideramos que talvez tivesse sido mais adequado distribuir os jogos desenvolvidos também pelos professores para que estes os pudessem experimentar e avaliar.

Relativamente ao desenvolvimento de jogos por parte dos professores, a avaliação de ambas as áreas da interface de utilizador que pretendem facilitar esse processo foi positiva. Houve ainda assim, um número significativo de utilizadores que deram respostas neutras a alguns dos aspetos a avaliar, porém, como referimos, esses valores puderam estar apenas relacionados com o facto de não terem explorado todas as funcionalidades que pretendíamos que avaliassem. Por outro lado, o aspeto considerado mais negativo foi a simplicidade de utilização, o qual deverá ser tido em conta como trabalho futuro. Apesar desses aspetos menos positivos, o processo de desenvolvimento foi, de um modo geral, considerado fácil e os utilizadores mostraram-se agrados com a plataforma e interessados em a utilizar.

No que diz respeito aos jogos, os estudantes classificaram a sua experiência com estes como interessante e divertida. Além disso, demonstraram interesse pelos meios de colaboração disponibilizados e gostaram do conceito da maioria das ajudas. Porém, identificaram alguns problemas a nível do fluxo do jogo, das pontuações e da informação apresentada, para os quais sugeriram em alguns casos alternativas ou melhorias que poderão ser também consideradas como trabalho futuro.

Finalmente, o tema [Cubic](#) para o [Moodle](#) foi também avaliado como interessante pelo grupo de utilizadores que o experimentaram, tal como o *chat* e o conceito de aplicações. Durante a realização dos testes, verificaram-se alguns problemas no entendimento do conceito de aplicações, contudo quando este se tornava claro todos se mostravam bastante interessados.

6

Conclusões e Trabalho Futuro

6.1 Conclusões

Nesta dissertação foi proposta uma plataforma, denominada [Epik](#), para o desenvolvimento, distribuição e execução de jogos educativos. Esta plataforma fornece aos professores uma forma simples e fácil de criar jogos, e aos estudantes uma forma mais divertida de aprender por meios de colaboração e interação. Os jogos que podem ser criados nesta plataforma são do tipo Questionário Individual ou Questionário Colaborativo. Esses jogos são compostos por um conjunto de cenários ligados entre si que podem conter texto, formas geométricas, recursos didáticos, e atividades. Para o desenvolvimento desses jogos foram criadas várias áreas da interface de utilizador, que permitem aos utilizadores criar e parametrizar os vários conteúdos e propriedades de um jogo.

Para além disso, esta plataforma permite a importação de conteúdos didáticos e de atividades educativas existentes num curso de um sistema [Moodle](#), e pode ainda ser integrada neste último. Além dessa plataforma, foram ainda propostas várias extensões para o [Moodle](#) 2.3, entre elas o tema [Cubic](#) que introduz o conceito de instituições, aplicações, sistemas de notificações e um *chat* como uma nova forma de comunicação e colaboração.

Antes de iniciar o processo de implementação da plataforma e das extensões para o [Moodle](#), existiu uma primeira fase de estudo, com o objetivo de fazer um levantamento do estado da arte. Esse estudo incidiu principalmente nas características necessárias aos ambientes de aprendizagem e na utilização de *software colaborativo* como atividades educativas, dando especial ênfase aos jogos educativos. Com base na informação recolhida foi possível identificar as características a garantir na plataforma [Epik](#) e as tecnologias a utilizar para a sua implementação. Além disso, foram identificados vários problemas nos [LMS](#) existentes na atualidade, o que nos levou a implementar um tema e um *chat*

para o [Moodle](#) como forma de demonstrar uma solução para alguns dos problemas identificados a nível da utilização desses sistemas. Depois de terminada essa fase de estudo, procedeu-se a uma segunda fase dedicada à elaboração da plataforma proposta e das extensões para o [Moodle](#).

Logo no início da fase de elaboração decidiu-se que dado o tempo que tínhamos para desenvolver a plataforma proposta teríamos de optar por decisões menos ótimas a nível de desempenho, de modo a que conseguíssemos atingir os nossos objetivos. Durante essa fase, foram identificados vários problemas que nos levaram a tomar decisões diferentes das que tínhamos inicialmente planeado, tais como: (i) a replicação dos dados de um conteúdo do [Moodle](#) para o [Epik](#), dado que estes só podem ser acedidos no contexto de um utilizador; (ii) a utilização de uma resolução fixa para os jogos por não se ter revelado simples torná-la adaptável ao ecrã do utilizador; (iii) o envio de uma nota com um valor de 100% para o [LMS](#) no final de um jogo, por o *standard IMS LTI* não suportar o envio de outro tipo de valores numéricos.

Além desses problemas, o processo de geração de um jogo foi o que tivemos mais dificuldade em definir numa fase inicial, mas que se tornou mais claro à medida que a plataforma foi evoluindo. Inicialmente a nossa abordagem envolvia a geração do código do jogo com todos os dados das atividades e recursos neste contidos. Contudo, concluímos que essa solução era desnecessária dado que o que varia nestes jogos é apenas o seu conteúdo e não o código. Além disso, essa solução levaria a que os dados desses conteúdos fossem carregados para o cliente e que este último fosse o responsável pela validação de respostas, pontuações e fluxo do jogo. Por essas razões, optámos antes pela criação da base de dados de jogos e por centralizar a lógica do jogo no servidor.

Após terminar todo o processo de implementação, encontrámos ainda alguns problemas na instalação das tecnologias escolhidas num servidor o que atrasou o inicio da fase de avaliação. Por essa razão, esta última ficou reduzida a um período de duas semanas, no qual foi pedido a diferentes grupos de utilizadores que testassem o processo de desenvolvimento de jogos no [Epik](#), a jogabilidade dos jogos de questionário e o tema [Cubic](#) desenvolvido para o [Moodle](#). Com base nos resultados obtidos, foi possível concluir que a plataforma [Epik](#) facilita o processo de desenvolvimento de jogos e que os estudantes se sentem interessados por estes últimos. Porém, nem todos os resultados foram positivos o que significa que há ainda melhorias a fazer de futuro, principalmente a nível da simplicidade de utilização do [Epik](#) e da qualidade da informação apresentada nos jogos.

Ainda no âmbito da elaboração deste trabalho, foi submetido e aceite um artigo na conferência internacional da especialidade, *International Conference on Cognition and Exploratory Learning in Digital Age* (CELDA 2012) [SMB12].

Além desse artigo, foram submetidos como novos *plugins* para o [Moodle](#), o tema [Cubic](#)¹ em conjunto com os pacotes de instituições e aplicações, e o *chat global*², os quais estão de momento a aguardar aprovação.

¹https://moodle.org/plugins/view.php?plugin=theme_cubic consultado em 21-03-2013.

²https://moodle.org/plugins/view.php?plugin=block_gchat consultado em 21-03-2013.

Por último, tendo em conta que este trabalho se trata de um protótipo que pretendia avaliar a viabilidade de uma plataforma de desenvolvimento e distribuição de jogos, são apresentados na próxima secção melhorias, modificações e adições que podem ser realizadas como trabalho futuro.

6.2 Trabalho Futuro

Esta secção contempla algumas ideias sobre o trabalho futuro que poderá ser desenvolvido no âmbito do tema desta dissertação. Existem alguns ajustes e melhorias que podem ser feitos, tanto a nível da implementação da plataforma [Epik](#) como a nível da sua utilização e desempenho. Além disso, podem ser criados novos tipos de jogos, ou podem ser feitas melhorias aos jogos de questionário do [Epik](#) e algumas das regras destes últimos poderão ter que se adaptar aos problemas identificados pelos jogadores durante a avaliação da plataforma.

Numa primeira fase seria interessante realizar testes de desempenho à plataforma [Epik](#) que permitissem avaliar vários aspetos a nível dos processos de desenvolvimento e geração de um jogo. Algumas sugestões são:

- Avaliar o comportamento do servidor de gestão e desenvolvimento durante o desenvolvimento de vários jogos por parte de diferentes utilizadores. Para este caso seria interessante avaliar o comportamento das ações de carregamento do projeto, armazenamento das alterações a um projeto e geração do jogo;
- Avaliar o comportamento do cliente (*browser*) durante o carregamento de um projeto e desenvolvimento de jogos mais complexos que, por exemplo, contenham muitos cenários ou cenários com um elevado número de conteúdos. Independentemente dos resultados obtidos talvez fosse interessante definir um limite de cenários por jogo e um limite de conteúdos por cenário de forma a melhorar o desempenho no cliente e servidor, e a impedir práticas incorretas de desenvolvimento de jogos;
- Avaliar o comportamento do servidor de execução quando existe um elevado número de sessões a decorrer. No caso de o seu comportamento não ser o esperado uma solução seria a utilização de várias máquinas pelas quais seriam distribuídas as várias sessões a decorrer.

Além dessas avaliações será necessário investigar soluções para alguns dos problemas que se levantaram ao longo da implementação desta plataforma, como os seguintes:

- Apesar de o [Epik](#) funcionar corretamente nas versões mais recentes dos *browsers* mais utilizados na atualidade, existem algumas funcionalidades que não funcionam de igual forma entre eles ou que não são disponibilizadas por nem sempre funcionarem como seria esperado. Por essa razão, talvez fosse interessante encontrar soluções para os problemas que se verificam em *browsers* que não sejam o Google Chrome.

- A linguagem que define a estrutura dos ficheiros XML, apesar de ter aspetos que se adaptam a alguns tipos de jogos, poderá não ser a mais apropriada para novos jogos que surjam de futuro. Por essa razão, será necessário avaliar com base nesses novos jogos se fará sentido criar várias linguagens em XML Schema, havendo uma que define a estrutura geral de um projeto e outras que definem aspetos mais específicos de cada tipo de jogo.
- A quantidade de ficheiros JavaScript a carregar durante o carregamento do ambiente de desenvolvimento poderá revelar-se um problema quando muitos utilizadores estão a aceder ao sistema, além de que isso poderá levar a elevados tempos de espera no cliente. Por essa razão, propõe-se a investigação de métodos que permitam, por exemplo, reduzir o tamanho desses ficheiros e ao mesmo tempo fundi-los num só ficheiro (idêntico à técnica usada pelo [Moodle](#)), ou em vários ficheiros mais pequenos. Idealmente esses ficheiros seriam ainda armazenados em *cache* no servidor não sendo assim necessário gerá-los por cada vez que um utilizador acede ao ambiente de desenvolvimento;
- O servidor de execução está neste momento preparado para a execução de jogos do tipo questionário. No entanto, será necessário investigar se fará sentido que este servidor seja o responsável pela execução de todos os tipos de jogos que o [Epik](#) venha a suportar, ou se será mais adequado utilizar um servidor de execução distinto para cada tipo de jogo.

Os resultados obtidos nas avaliações ao sistema e nesse processo de investigação poderão influenciar o caminho a seguir no que diz respeito ao suporte de novos jogos, quer seja relativamente ao seu desenvolvimento como à sua execução. Os novos jogos a suportar poderão ser variantes dos questionários já desenvolvidos, ou novos tipos de jogos. Um outro tópico que poderá também ser interessante considerar é o desenvolvimento de jogos para dispositivos móveis, visto que começam a haver cada vez mais trabalhos de investigação sobre a utilização desses dispositivos no processo de aprendizagem.

No que diz respeito à integração da plataforma [Epik](#) com os [LMS](#) seria interessante investigar a possibilidade de importar conteúdos de outros [LMS](#), pois o sistema já se encontra preparado para isso. Por outro lado, seria também interessante investigar para esses mesmos [LMS](#) se suportam o [IMS LTI](#), de forma a testar o trabalho realizado com esse *standard* noutros sistemas que não o [Moodle](#).

A nível do desenvolvimento de jogos poderiam ser adicionadas novas funcionalidades aos projetos e ao ambiente de desenvolvimento do [Epik](#) que tornam esse processo mais fácil e simples para os utilizadores. Alguns exemplos seriam:

- Um modo tutorial que ensine a um novo utilizador como utilizar o ambiente de desenvolvimento do [Epik](#);
- Ferramentas para desfazer ou refazer ações, ou seja, voltar atrás ou à frente, e uma

outra que permita testar um jogo em desenvolvimento sem que seja necessário gerar o jogo final. As ferramentas para esses efeitos são inclusivamente já disponibilizadas na barra de ferramentas do ambiente de desenvolvimento necessitando apenas de ser implementadas;

- Uma nova área no ambiente de desenvolvimento que permita a visualização e edição do fluxo de um jogo de forma gráfica;
- Opções que estão associadas ao utilizador e que permitem definir, por exemplo, os valores a associar às pontuações de uma atividade quando esta é criada, os sons a utilizar em todos os jogos, entre outros aspetos que não possam ser definidos nos *templates* de projetos;
- Uma ferramenta para criação de *templates* de projetos, a qual é já disponibilizada pelo sistema necessitando apenas de ser implementada. Esta ferramenta permitiria aos utilizadores reutilizar a estrutura definida num projeto para a criação de outros jogos idênticos;
- Uma nova ferramenta para duplicação de um projeto, permitindo assim que o utilizador reutilize a estrutura desse projeto e todos os seus conteúdos sem que seja necessário refazer tudo. Além disso, esta deverá permitir a conversão do projeto para outro tipo de jogo se o utilizador assim o desejar;
- Um novo parâmetro nos projetos que permite definir a resolução do ecrã de jogo, algo que poderia inclusivamente ser útil para a criação de jogos para dispositivos móveis;

Quanto aos jogos do [Epic](#) poderiam ser feitas algumas melhorias a nível das suas regras e da informação que disponibilizam, tais como:

- Um menu de opções que disponibilize aos jogadores, uma forma de desligar o som do jogo, de pausar o jogo (apenas para jogos *singleplayer*), de sair do jogo, entre outros;
- Notificar os jogadores quando recebem um bónus em vez de este ser simplesmente adicionado aos seus pontos sem que estes percebam o que aconteceu;
- Alterar o método de resposta a perguntas de resposta curta, passando a ser necessário submeter a resposta para que esta seja validada pelo servidor de execução de jogos;

Finalmente, há ainda outras melhorias e funcionalidades menos relevantes mas também interessantes, que poderiam ser consideradas, como por exemplo: (i) a possibilidade de organizar atividades, recursos, projetos e jogos por pastas ou categorias; (ii) a utilização de gráficos para visualização dos registos de sessões; (iii) a criação de uma área a

partir da qual poderiam ser pesquisados e acedidos todos os jogos públicos existentes no [Ezik](#) organizados por categorias (áreas de ensino).

Bibliografia

- [Ama12] Amanita Design. Questionaut. <http://amanita-design.net/games/questionaut.html>, Consultado em 04/12/2012.
- [Arc12] Academics. Tractor multiplication. <http://www.academickskillbuilders.com/games/tractor-multiplication>, Consultado em 28/06/2012.
- [Ash09] P. Ashley. A teaching with technology white paper - collaboration tools. *Teaching with Technology*, 2009.
- [Att07] G. Attwell. Personal learning environments-the future of eLearning? *eLearning Papers*, 2(1):1–7, 2007.
- [Bla12] Blackboard Inc. Blackboard. <http://www.blackboard.com/>, Consultado em 15/03/2012.
- [Bod13] Chris Boden. Ratchet - PHP websockets. <http://socketo.me/>, Consultado em 07/02/2013.
- [BTMGM09] A. Blanco, J. Torrente, P. Moreno-Ger, e B. Fernandez-Manjon. A general architecture for the integration of educational videogames in standards-compliant virtual learning environments. In *Advanced Learning Technologies, 2009. ICALT 2009. Ninth IEEE International Conference on*, pág. 53 –55, Julho 2009.
- [Cak13] Cake Software Foundation, Inc. CakePHP. <http://cakephp.org/>, Consultado em 18/02/2013.
- [Car12] Carnegie Mellon University. Alice. <http://www.alice.org/>, Consultado em 24/05/2012.
- [Cha10] Charles Severance. IMS basic learning tools interoperability. <http://vimeo.com/14100773>, Consultado em 12/08/2010.

- [CL96] Seth Chaiklin e Jean Lave. Developmental studies of work as a test-bench of activity theory: The case of primary care medical practice. In *Understanding Practice: Perspectives on Activity and Context*, pág. 64–103. Cambridge University Press, Maio 1996.
- [CMH08] Thibault Carron, Jean-Charles Marty, e Jean-Mathias Heraud. Teaching with game-based learning management systems: Exploring a pedagogical dungeon. *Simul. Gaming*, 39(3):353–378, Setembro 2008.
- [Cou12] Coursera. Coursera courses. <https://www.coursera.org/courses>, Consultado em 28/06/2012.
- [CRE12] CREO. Quest4K. <http://demo.quest4k.com/>, Consultado em 28/06/2012.
- [Cro13] Douglas Crockford. JSON. <http://json.org/>, Consultado em 13/02/2013.
- [Des12] Desire2Learn Incorporated. Desire2Learn. <http://desire2learn.com/>, Consultado em 15/03/2012.
- [Don07] Mary Dondlinger. Educational video game design: A review of the literature. *Journal of Applied Educational Technology*, 4:21–31, 2007.
- [e-U12] e-UCM. e-adventure. <http://e-adventure.e-ucm.es/>, Consultado em 24/05/2012.
- [eLe12a] eLearningBrothers. Moodle eLearningBrothers. <http://moodle.elearningbrothers.com/>, Consultado em 29/06/2012.
- [Ele12b] Electricity. Incredibots. <http://incredibots.com/if/game.php>, Consultado em 28/06/2012.
- [Ele12c] Electronic Arts. SimCity. http://www.simcity.com/en_US, Consultado em 28/06/2012.
- [Emi12] Emily Navarro. SimSE. <http://www.ics.uci.edu/~emilyo/SimSE/>, Consultado em 15/04/2012.
- [Emm12] Emmett Nicholas. Minesweeper. <http://minesweeperonline.com/>, Consultado em 28/06/2012.
- [For12a] Forfás. NanoQuest. <http://www.nanoquest.ie/>, Consultado em 03/06/2012.
- [For12b] Forsvarsmakten. The swedish armed forces. <http://team.forsvarsmakten.se/english/>, Consultado em 03/10/2012.

- [Gav12] Gavin Henrick. Ireland and UK moodlemoot 2012 - IMS LTI demo. <http://www.somerandomthoughts.com/blog/2012/04/11/ireland-and-uk-moodlemoot-2012-ims-lti-demo/>, Consultado em 29/06/2012.
- [Gei13] Felix Geisendorfer. Node-MySQL. <https://github.com/felixge/node-mysql>, Consultado em 13/02/2013.
- [GHS⁺09] Charlotte N. Gunawardena, Mary Beth Hermans, Damien Sanchez, Carol Richmond, Maribeth Bohley, e Rebekah Tuttle. A theoretical framework for building online communities of practice with social networking tools. *Educational Media International*, 46(1):3–16, 2009.
- [Goo13] Google. Google drive. <https://drive.google.com>, Consultado em 17/03/2013.
- [Gra09] D. Graham. The evolution of a cooperative work framework for e-learning. *The Journal of eWorking*, 3(2):1–16, 2009.
- [HBL⁺09] A. Hemmi, S. Bayne, R. Land, A. Hemmi, S. Bayne, e R. Land. The appropriation and repurposing of social technologies in higher education. *Journal of Computer Assisted Learning*, 25, 25(1, 1):19, 19–30, 30, Fevereiro 2009.
- [IG12] Ali Hakan Işık e İnan Güler. Comprehensive comparison of traditional and distance learning master programs. *Procedia - Social and Behavioral Sciences*, 31(0):120–123, 2012.
- [IMS12] IMS Global. IMS learning tools interoperability. <http://www.imsglobal.org/toolsinteroperability2.cfm>, Consultado em 24/06/2012.
- [JED⁺11] Jacob George, Eric de Araujo, Desiree Dorsey, D. Scott McCrickard, e Greg Wilson. Multitouch tables for collaborative object-based learning. *Design, User Experience, and Usability. Theory, Methods, Tools and Practice*, pág. 237–246, 2011.
- [Joy13] Joyent, Inc. NodeJS. <http://nodejs.org/>, Consultado em 07/02/2013.
- [JT10] H. Jiang e M. X Tang. Web-based learning platforms integrating social networking for design education at high schools in china. In *2010 International Conference on Computational Intelligence and Software Engineering (CiSE)*, pág. 1–3. IEEE Xplore, 2010.
- [KCC⁺02] C. Kelleher, D. Cosgrove, D. Culyba, C. Forlines, J. Pratt, e R. Pausch. Alice2: programming without syntax errors. In *User Interface Software and Technology*, 2002.

- [KKJ02] K. Kreijns, P.A. Kirschner, e W. Jochems. The sociability of computer-supported collaborative learning environments. *Educational Technology & Society*, 5(1):8–22, 2002.
- [Koc08] Ned F. Kock. *Encyclopedia of E-Collaboration*. Idea Group Inc (IGI), 2008.
- [KP88] Glenn E. Krasner e Stephen T. Pope. A cookbook for using the model-view controller user interface paradigm in smalltalk-80. *J. Object Oriented Program.*, 1(3):26–49, Agosto 1988.
- [KRCB10] Kathryn Kane, Janine Robinson-Combre, e Zane L. Berge. Tapping into social networking: Collaborating enhances both knowledge management and e-learning. *VINE*, 40(1):62–70, Fevereiro 2010.
- [KSG⁺12] Egor Kuryanovich, Shy Shalom, Russell Goldenberg, Mathias Paumgarten, David Strauß, Seb Lee-Delisle, Gaëtan Renaudeau, Jonas Wagner, Jonathan Bergknoff, Brian Danchilla, e Rob Hawkes. A real-time multiplayer game using WebSockets. In *HTML5 Games Most Wanted*, pág. 213–238. Apress, Janeiro 2012.
- [LaP03] Maria LaPadula. A comprehensive look at online student support services for distance learners. *American Journal of Distance Education*, 17(2):119–128, 2003.
- [LP08] L. Lockyer e J. Patterson. Integrating social networking technologies in education: Case study of a formal learning environment. In *Advanced Learning Technologies, 2008. ICALT '08. Eighth IEEE International Conference on*, pág. 529 –533. IEEE Xplore, Julho 2008.
- [LT08] April Lynn Luehmann e Liz Tinelli. Teacher professional identity development with social networking technologies: learning reform through blogging. *Educational Media International*, 45(4):323–333, 2008.
- [Mas12] Massachusetts Institute of Technology. MIT open courseware. <http://ocw.mit.edu/courses/>, Consultado em 28/06/2012.
- [McG11] Jane McGonigal. *Reality Is Broken: Why Games Make Us Better and How They Can Change the World*, volume 22. Penguin Press HC, 2011.
- [MD07] Riccardo Mazza e Vania Dimitrova. CourseVis: a graphical student monitoring tool for supporting instructors in web-based distance courses. *International Journal of Human-Computer Studies*, 65(2):125–139, Fevereiro 2007.
- [Mec08] Maram Meccawy. *A service-orientated architecture for adaptive and collaborative e-learning systems*. PhD, University of Nottingham, Julho 2008.

- [MGBMO⁺08] Pablo Moreno-Ger, Daniel Burgos, Iván Martínez-Ortiz, José Luis Sierra, e Baltasar Fernández-Manjón. Educational game design for online education. *Computers in Human Behavior*, 24(6):2530–2540, Setembro 2008.
- [Mic13] Microsoft. OAuth protocol. <http://oauth.net/>, Consultado em 17/03/2013.
- [Min12] Miniclip. Bloxorz. <http://miniclip.sapo.pt/games/bloxorz/pt/>, Consultado em 28/06/2012.
- [Moo12a] Moodle. Moodle. <http://moodle.org/>, Consultado em 15/03/2012.
- [Moo12b] Moodle. Moodle game module. http://docs.moodle.org/22/en/Game_module, Consultado em 23/06/2012.
- [Moo12c] Moodle. Registered moodle sites. <http://moodle.org/sites/>, Consultado em 28/06/2012.
- [MTJV09] Mathieu Muratet, Patrice Torguet, Jean-Pierre Jessel, e Fabienne Viallet. Towards a serious game to help students learn computer programming. *International Journal of Computer Games Technology*, 2009:1–12, 2009.
- [Nin12] Nintendo. Wii sports. <http://www.wiisportsresort.com/>, Consultado em 28/06/2012.
- [NPRI09] Nurzhan Nurseitov, Michael Paulson, Randall Reynolds, e Clemente Izurieta. Comparison of JSON and XML data interchange formats: A case study. *Computer Applications in Industry and Engineering*, 2009.
- [Ora13] Oracle. MySQL. <http://www.mysql.com/>, Consultado em 15/02/2013.
- [PB07] Marilícia Witzler Antunes Palmieri e Angela Uchoa Branco. Educação infantil, cooperação e competição: análise microgenética sob uma perspectiva sociocultural. *Psicologia Escolar e Educacional (Impresso)*, 11(2), 2007.
- [PMP10] Fotini Paraskeva, Sofia Mysirlaki, e Aikaterini Papagianni. Multiplayer online games as educational tools: Facing new challenges in learning. *Computers & Education*, 54(2):498–505, Fevereiro 2010.
- [Rau13] Guillermo Rauch. Socket.IO. <http://socket.io/>, Consultado em 07/02/2013.
- [Rel12] Relentless Software. Buzz!: The schools quiz. <http://www.buzzthegame.com/>, Consultado em 03/03/2012.

- [RF11] P. Ractham e D. Firpo. Using social networking technology to enhance learning in higher education: A case study using facebook. In *System Sciences (HICSS), 2011 44th Hawaii International Conference on*, pág. 1–10. IEEE Xplore, Janeiro 2011.
- [Riv12] Rivermind. CyberCiege. <http://cisl.nps.edu/cyberciege/>, Consultado em 03/06/2012.
- [Row13] Eric Rowell. KineticJS. <http://kineticjs.com/>, Consultado em 18/02/2013.
- [RR09] A. Ravenscroft e A. Ravenscroft. Social software, web 2.0 and learning: status and implications of an evolving paradigm. *Journal of Computer Assisted Learning*, 25, 25(1, 1):1, 1–5, 5, Fevereiro 2009.
- [Rus12] Rustici Software. SCORM. <http://scorm.com/>, Consultado em 24/05/2012.
- [Sab11] Sabrina Leone. Characterisation of a personal learning environment as a lifelong learning tool. *Doctoral School on Engineering Sciences*, Dezembro 2011.
- [Sak12] Sakai Foundation. Sakai project. <http://www.sakaiproject.org/>, Consultado em 15/03/2012.
- [SBS99] F.M. Santoro, M.R.S. Borges, e N. Santos. Um framework para estudo de ambientes de suporte à aprendizagem cooperativa. *Revista Brasileira de Informática na Educação*, 4:51–68, 1999.
- [SC09] Sahar Shabanah e Jim X. Chen. Simplifying algorithm learning using serious games. In *Proceedings of the 14th Western Canadian Conference on Computing Education*, WCCCE '09, pág. 34–41, New York, NY, USA, 2009. ACM.
- [Sch93] J. B.M Schick. The decision to use a computer simulation. *History Teacher*, pág. 27–36, 1993.
- [Scr12] Screenlife. Scene it? <http://www.screenlifegames.com/scene-it>, Consultado em 28/06/2012.
- [Sei11] Jacob Seidelin. *HTML5 Games: Creating Fun with HTML5, CSS3, and WebGL*. Wiley, 1 edition, Dezembro 2011.
- [Sha12] Aditya Ravi Shankar. Multiplayer with WebSockets. In *Pro HTML5 Games*, pág. 293–317. Apress, Janeiro 2012.

- [SMB12] Bruno Sampaio, Carmen Morgado, e Fernanda Barbosa. Collaborative quiz game developed with epik. *IADIS International Conference on Cognition and Exploratory Learning in Digital Age*, pág. 308–301, 2012.
- [SNJ09] G. Sindre, L. Natvig, e M. Jahre. Experimental validation of the learning effect for a pedagogical game on computer fundamentals. *Education, IEEE Transactions on*, 52(1):10–18, 2009.
- [Tak12] Take-Two Interactive Software. Civilization. <http://www.civilization.com/>, Consultado em 15/03/2012.
- [The13a] The jQuery Foundation. jQuery. <http://jquery.com/>, Consultado em 18/02/2013.
- [The13b] The PHP Group. PHP. <http://www.php.net/>, Consultado em 15/02/2013.
- [TMGMOFM09] J. Torrente, P. Moreno-Ger, I. Martínez-Ortiz, e B. Fernandez-Manjon. Integration and deployment of educational games in e-learning environments: the learning object model meets educational gaming. *Educational Technology & Society*, 12(4):359–371, 2009.
- [VC03] Jaan Valsiner e Kevin J. Connolly. *Handbook of Developmental Psychology*. SAGE, Fevereiro 2003.
- [Wiz12] WizIQ. Wiziq. <http://www.wiziq.com/>, Consultado em 28/06/2012.
- [WOMS09] A.I. Wang, T. Ofsdahl, e O.K. Mørch-Storstein. Collaborative learning through Games–Characteristics, model, and taxonomy. *Challenges*, 2009.
- [Wor13a] World Wide Web Consortium. The websocket API. <http://www.w3.org/TR/2011/WD-websockets-20110419/>, Consultado em 13/02/2013.
- [Wor13b] World Wide Web Consortium. CSS specification. <http://www.w3.org/TR/css-style-attr/>, Consultado em 15/02/2013.
- [Wor13c] World Wide Web Consortium. HTML5 specification. <http://www.w3.org/TR/html51/>, Consultado em 15/02/2013.
- [ZSG⁺09] Natalia Padilla Zea, José Luís González Sánchez, Francisco L. Gutiérrez, Marcelino J. Cabrera, e P. Paderewski. Design of educational multiplayer videogames: A vision from collaborative learning. *Advances in Engineering Software*, 40(12):1251–1260, Dezembro 2009.



Inquéritos de Testes de Utilizadores

A.1 Utilização e desenvolvimento de jogos

As questões do inquérito de satisfação e qualidade utilizado para avaliar o desenvolvimento de jogos no [Epik](#) são apresentadas na lista abaixo pela mesma ordem que foram realizadas aos utilizadores:

- **Tipo de utilizador**

1. Qual a sua faixa etária?
2. Qual o seu sexo?
3. Qual o seu grau de instrução?
4. Qual o(s) browser(s) que utilizou durante a utilização do Epik?

- **Utilização de jogos no ensino**

5. O que acha da ideia de utilizar jogos no ensino?
6. Já utilizou jogos no ensino?
7. Se sim, qual ou quais?
8. Alguma vez utilizou questionários no ensino com perguntas de escolha múltipla, resposta curta e verdadeiro ou falso?
9. Se sim, que ferramentas utilizou para os criar?
10. O que pensa do uso de cada tipo de questionários no ensino?
11. Usaria este tipo de jogos no ensino?

12. Se sim, essa atividade contaria para a avaliação do aluno?
13. No caso de Questionários Individuais, em que tipo(s) de atividade(s) usaria?
14. No caso de Questionários Colaborativos, em que tipo(s) de atividade(s) usaria?
15. Como se adaptam os seguintes tipos de ajudas dos Questionários Individuais ao ensino?
16. Como se adaptam as seguintes características dos Questionários Colaborativos ao ensino?
17. Em geral, como avalia a utilização dos Questionários Individuais como uma atividade individual no ensino?
18. Em geral, como avalia a utilização dos Questionários Colaborativos como uma atividade de grupo colaborativa no ensino?
19. Caso tenha algumas sugestões para novas formas de colaboração, coordenação e competição, indique-as abaixo.

• **Desenvolvimento de jogos educativos**

20. Alguma vez tinha utilizado ferramentas para desenvolver jogos educativos?
21. Se sim, quais?
22. Como classifica a sua experiência com essa(s) ferramenta(s) em termos de utilização e facilidade de desenvolvimento?
23. Como classifica a nomenclatura utilizada para identificar cada tipo de conteúdo?
24. Como classifica a realização das seguintes tarefas?
25. Considera útil a possibilidade de importar atividades ou recursos de um sistema Moodle?
26. Considera os templates de projetos úteis para o desenvolvimento de novos jogos?
27. Considera a descrição de cada tipo de jogo apresentada durante a criação de um projeto fácil de entender?
28. De uma forma geral, como classifica a *dashboard* em termos de utilização?
29. Como classifica a realização das seguintes ações de criação e inserção?
30. Considera a existência de templates de cenários útil para o desenvolvimento de um jogo?
31. Como avalia a parametrização das propriedades gerais (estilos) dos vários tipos de conteúdos?
32. Como avalia a parametrização do fluxo (ligação entre cenários) de um jogo?

33. Como classifica a parametrização de pontuações associadas a: (Cenários, Perguntas)?
34. Como classifica a parametrização das seguintes ajudas nas perguntas: (Consulta de conteúdos, Pedido de dicas, Remoção de respostas)?
35. Como classifica a parametrização dos valores a registar (guardar) no final do jogo?
36. No geral, como avalia a organização deste ambiente de desenvolvimento de jogos?
37. No geral, como classifica o desenvolvimento de um questionário no Epik?
38. Caso tenha algumas sugestões ou críticas acerca deste ambiente, indique-as abaixo.
39. Como classifica o processo de geração de um jogo?
40. Alguma vez terminou o processo de geração de um jogo sem sucesso?
41. Se sim, considerou a descrição do erro esclarecedora?

- **Satisfação com o Epik**

42. Das palavras abaixo quais na sua opinião melhor caracterizam o Epik?
43. Indique o seu grau de satisfação relativamente aos seguintes aspetos do Epik.
44. Recomendaria o Epik a outros amigos/colegas?
45. Caso tenha algumas sugestões, comentários, ou críticas por favor coloque-os abaixo.

O inquérito completo pode ser encontrado em <https://docs.google.com/spreadsheets/viewform?formkey=dDhuNnNnSUdqY1BqZnNNZWNJdGhNR2c6MA#gid=0>.

A.2 Utilização e jogabilidade dos jogos

As questões do inquérito de satisfação e qualidade utilizado para avaliar a utilização dos jogos do Epik são apresentadas na lista abaixo pela mesma ordem que foram realizadas aos utilizadores:

- **Tipo de utilizador**

1. Qual a sua faixa etária?
2. Qual o seu sexo?
3. Qual o seu grau de instrução?
4. Qual o(s) browser(s) que utilizou para jogar os jogos do Epik?

- **Utilização de jogos no ensino**

5. O que acha da ideia de utilizar jogos no ensino?
6. Já tinha jogado jogos educativos anteriormente?
7. Se sim, quais?
8. Alguma vez realizou um questionário como atividade educativa?
9. Se sim, como avalia essa experiência?
10. O que pensa do uso de cada um desses tipos de jogos no ensino?
11. O que pensa da utilização destes jogos como uma forma de avaliação?
12. Como avalia as ajudas disponibilizadas nestes jogos?
13. Como avalia as seguintes características presentes nos Questionários Colaborativos?
14. Em geral, como avalia a componente competitiva destes jogos?
15. De modo geral, considera estes jogos bons para o processo de aprendizagem?

• **Jogabilidade dos jogos do Epik**

16. Como classifica a nomenclatura utilizada nestes jogos?
17. Como classifica os ícones utilizados nestes jogos?
18. Considera interessante o uso do painel de jogadores como uma forma de percepção do estado do jogo?
19. Como classifica a informação apresentada no painel de jogadores?
20. Caso tenha jogado com som, como avalia os sons utilizados?
21. Como classifica o método de resposta a perguntas de: (Escolha múltipla, Verdadeiro ou falso e Resposta Curta)?
22. Considera o fluir do jogo simples de entender?
23. Como classifica a informação apresentada nas janelas de:
24. Nos Questionários Colaborativos, como classifica a forma de resposta a um pedido de ajuda?
25. No geral, como classifica a sua experiência com estes jogos?

• **Satisfação com os jogos do Epik**

26. Das palavras abaixo quais na sua opinião melhor caracterizam os jogos de Questionário do Epik?
27. Indique o seu grau de satisfação relativamente aos seguintes aspetos dos jogos do Epik.
28. Recomendaria estes jogos a outros amigos/colegas?
29. Caso tenha algumas sugestões, comentários, ou críticas por favor coloque-os abaixo.

O inquérito completo pode ser encontrado em <https://docs.google.com/forms/d/1RXRAo8jz4VCC56b9b2dNXbqvhC-OGVlxQUAmm2vhic/viewform>.

A.3 Utilização do tema Cubic para o Moodle

As questões do inquérito de satisfação e qualidade utilizado para avaliar o tema Cubic são apresentadas na lista abaixo pela mesma ordem que foram realizadas aos utilizadores:

- **Tipo de utilizador**

1. Qual a sua faixa etária?
2. Qual o seu sexo?
3. Qual o seu grau de instrução?
4. À quantos anos trabalha com o Moodle?
5. Qual o(s) browser(s) que utilizou para testar o tema Cubic?
6. Qual ou quais os papéis que desempenhou como utilizador do Cubic?

- **Barra de utilizador**

7. Considera útil a informação disponibilizada pelos menus da barra de utilizador?
8. O menu de atividades (último menu no lado esquerdo da barra) disponibiliza uma forma prática e rápida de consultar as atividades dos vários cursos que gere ou em que está inscrito. Concorda?
9. O menu de notificações (primeiro menu no lado direito da barra, a contar da esquerda) ajuda-o a manter-se informado sobre as novidades dos vários cursos que gere ou em que está inscrito. Concorda?
10. O menu de eventos (segundo menu no lado direito da barra, a contar da esquerda) ajuda-o a manter-se informado sobre as atividades que tem por realizar. Concorda?
11. Considera esta forma de organizar a informação das instituições mais prática e intuitiva que a geralmente utilizada?
12. Considera a informação apresentada por cada instituição suficiente? (Caso a resposta seja negativa indique por favor no final deste inquérito o que gostaria de encontrar neste menu)
13. No caso de a resposta à pergunta anterior ter sido negativa, indique que outros tipos de informação gostaria de encontrar neste menu.
14. Considera a possibilidade de integrar aplicações no Moodle útil?

15. Considera esta forma de integração de aplicações no Moodle útil?

• **Organização dos conteúdos de um curso**

16. Como avalia esta forma de organização?

17. De uma forma geral, como classifica os icons utilizados para identificar os diferentes tipos de atividades e recursos?

• ***Chat global***

18. O chat oferece uma forma mais prática, simples e familiar de comunicar e colaborar com os seus colegas/alunos. Concorda?

19. Considera que seria importante ter a possibilidade de definir quando é que o chat está disponível?

20. Como avalia a utilidade deste chat?

• **Satisfação com o Cubic**

21. Das palavras abaixo quais é que na sua opinião melhor caracterizam o tema Cubic?

22. Numa escala de 1 a 5, indique o seu grau de satisfação com os seguintes aspetos do Cubic.

23. Recomendaria o Cubic a outros amigos/colegas?

24. Numa escala de 1 a 5, como avalia o tema Cubic?

25. Caso tenha algumas sugestões, comentários, ou críticas por favor coloque-os abaixo.

O inquérito completo pode ser encontrado em <https://docs.google.com/spreadsheets/viewform?formkey=dGZFeG12eWJvelEzVlpDUzN1aUN3QXc6MQ#gid=0>.



Diagramas de Atividades

Neste capítulo são apresentados os diagramas de atividades de algumas das ações que podem ser realizadas pelo servidor de execução de jogos. As ações correspondentes a estes diagramas são descritas na secção 4.2.4.

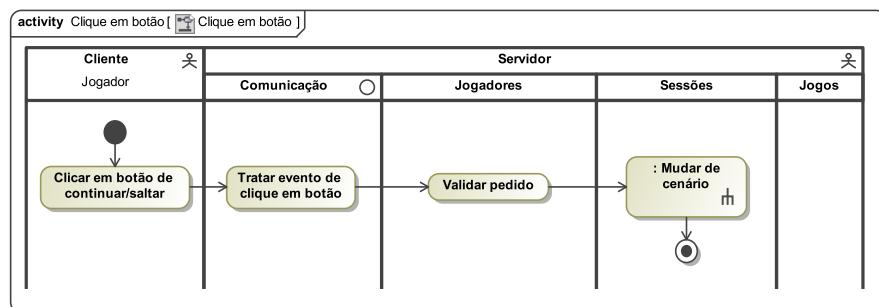


Figura B.1: Diagrama de atividades da notificação de clique em botão de continuar ou saltar.

B. DIAGRAMAS DE ATIVIDADES

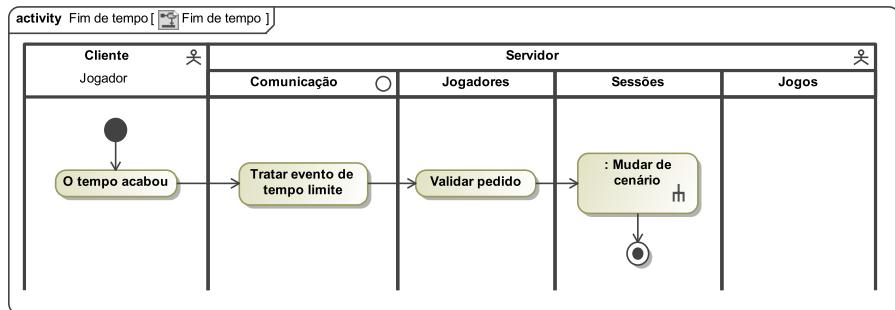


Figura B.2: Diagrama de atividades da notificação de fim do tempo de permanência no cenário.

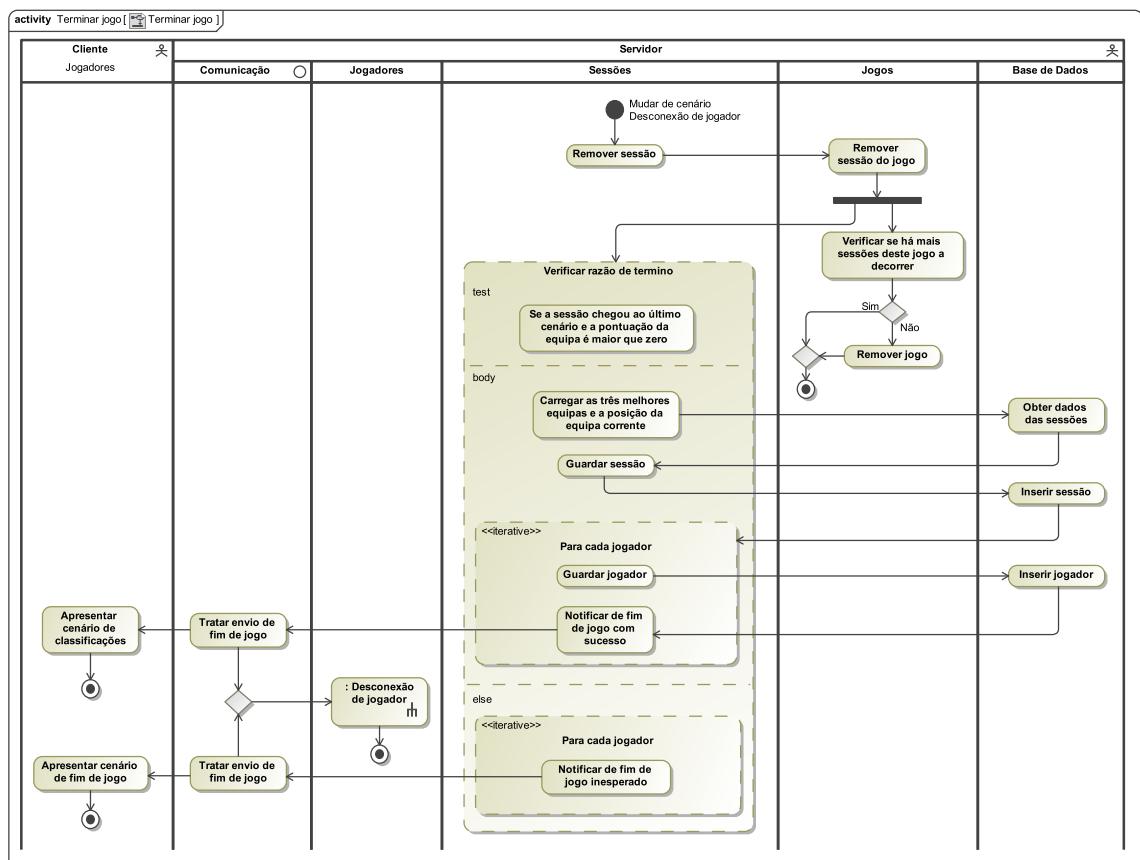


Figura B.3: Diagrama de atividades da notificação de término de jogo.

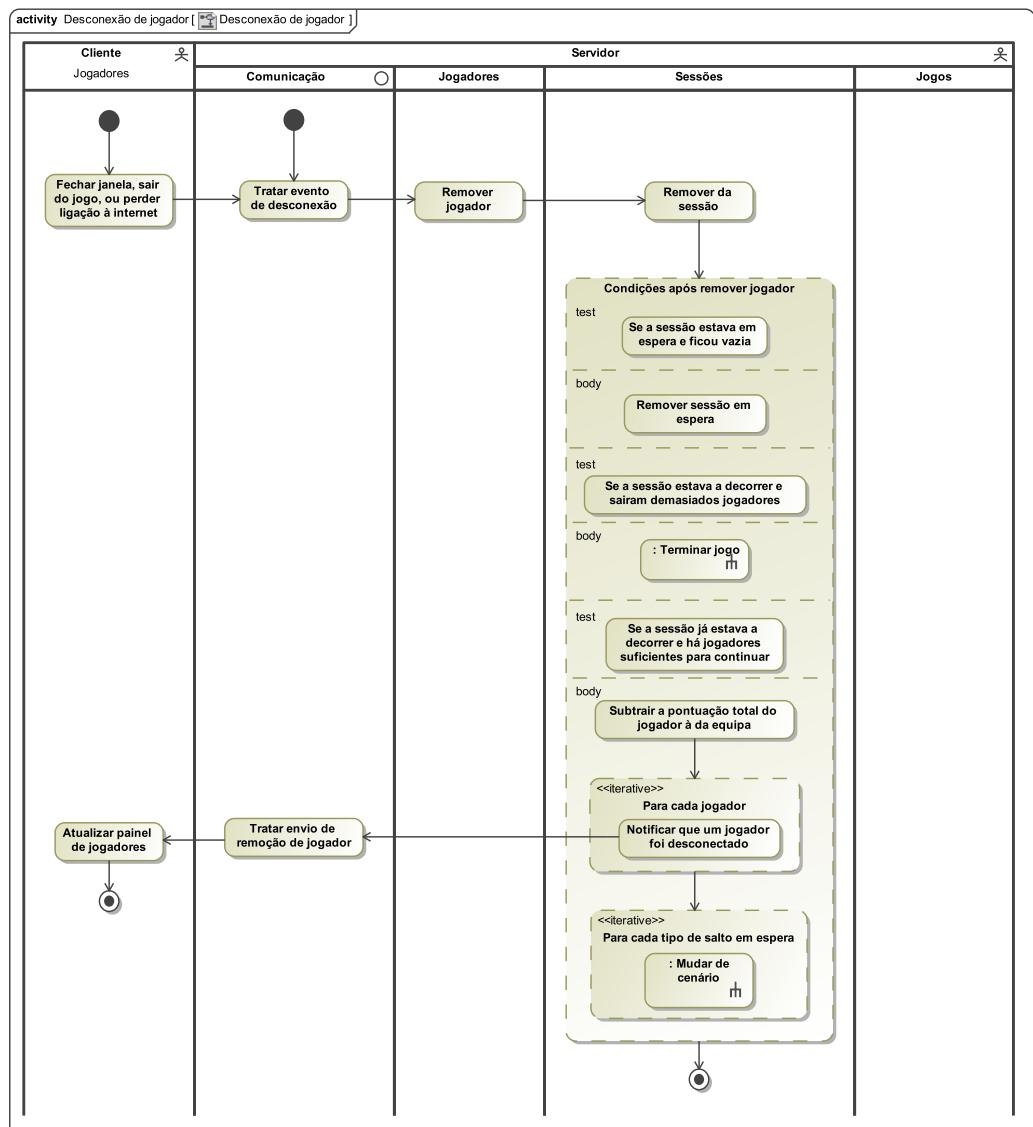


Figura B.4: Diagrama de atividades da notificação de desconexão de um jogador.