



Universidade de Évora - Escola de Ciências e Tecnologia

Mestrado em Engenharia Informática

Dissertação

**Realidade Virtual e e-Aprendizagem: Especificação de
Perguntas de Escolha Múltipla**

Tony Babej Chaveiro

Orientador(es) | Francisco Manuel Coelho

Évora 2022



Universidade de Évora - Escola de Ciências e Tecnologia

Mestrado em Engenharia Informática

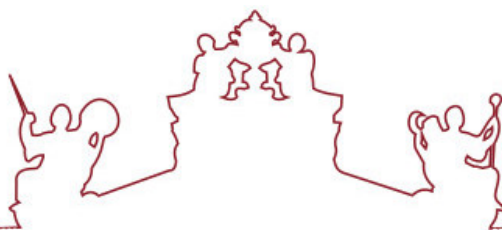
Dissertação

**Realidade Virtual e e-Aprendizagem: Especificação de
Perguntas de Escolha Múltipla**

Tony Babej Chaveiro

Orientador(es) | Francisco Manuel Coelho

Évora 2022



A dissertação foi objeto de apreciação e discussão pública pelo seguinte júri nomeado pelo Diretor da Escola de Ciências e Tecnologia:

Presidente | Lígia Maria Ferreira (Universidade de Évora)

Vogais | António Anjos (Universidade de Évora) (Arguente)
Francisco Manuel Coelho (Universidade de Évora) (Orientador)

Nunca percas 6 minutos a realizar uma tarefa manualmente quando podes perder 6 horas a falhar em automatizá-la.
- Zhuowei Zhang

Prefácio

A Realidade Virtual (RV) tem ganho imensa popularidade nos últimos anos. Por um lado, essa popularidade é muito impulsionada pela massificação e consequente descida dos custos de aquisição dos dispositivos físicos, por outro, o aparecimento de normas e especificações têm permitido às empresas fazer investimentos nesta área, o que tem beneficiado a comunidade no seu todo.

Desde finais de 2019, as ferramentas de ensino à distância têm ganho especial relevância dados os eventos pandémicos que desencadearam evoluções e adaptações "forçadas" na área do ensino/aprendizagem. Surgiram assim as condições perfeitas para acelerar o desenvolvimento e adoção de tecnologias emergentes, sendo a RV uma delas.

No entanto, existem ainda algumas resistências para que a RV se torne uma ferramenta *mainstream* no ensino. Uma das problemáticas centra-se no facto de não existirem muitas ferramentas intuitivas que permitam o desenvolvimento de atividades de RV sem conhecimentos técnicos aprofundados na temática.

Este trabalho pretende apresentar uma solução funcional, na área do ensino/aprendizagem, para o problema enunciado e demonstrar, através de uma Prova de Conceito (PdC), a geração de uma atividade, em RV, de um conjunto de questões, de resposta dicotómica e/ou múltipla, usando apenas um ficheiro de configuração simples e facilmente modificável por parte do utilizador/tutor. A solução poderá, futuramente, também ser integrada em sistemas de gestão de aprendizagem.

Agradecimentos

Esta dissertação é o culminar de mais uma fase na minha vida que, por motivos diversos, levou muito mais tempo a concluir do que alguma vez imaginei. No entanto, confesso que o percurso se revelou extremamente enriquecedor, quer a nível académico, quer a nível pessoal. Não posso deixar de agradecer a todas as pessoas que me ajudaram, de forma direta ou indireta, no desenvolvimento deste trabalho.

Em primeiro lugar, gostaria de agradecer ao meu orientador, Professor Doutor Francisco Coelho, pela infundável paciência e disponibilidade que demonstrou, ao longo de todo o percurso, muitas vezes atribulado, do desenvolvimento deste trabalho. Não desistiu de mim quando eu próprio, na sua posição, muito provavelmente o teria feito, em múltiplas ocasiões.

Em segundo lugar, quero agradecer a toda a minha família que me incentivou e motivou para concluir esta etapa. Quero agradecer em particular aos meus filhos, que se voluntariaram inúmeras vezes como *Beta Testers* da parte prática desta dissertação, o que, além de permitir aprimorar a aplicação, gerou motivação adicional para concluir esta tarefa.

Finalmente, mas não menos importante, um agradecimento especial a Emiliano Sauvisky, brilhante empreendedor e programador, que me introduziu ao desenvolvimento corporativo e cooperativo de aplicações, incluindo ao mal-amado *git*, que se revelou tão importante no decurso deste trabalho.

Conteúdo

Conteúdo	xiii
Lista de Figuras	xvii
Lista de Tabelas	xxi
Lista de Acrónimos	xxiii
Resumo	xxv
Abstract	xxvii
1 Introdução	1
1.1 Enquadramento	1
1.2 Objetivos	3
1.3 Análise S.W.O.T.	3
1.3.1 <i>Strengths</i>	3
1.3.2 <i>Weaknesses</i>	4
1.3.3 <i>Opportunities</i>	4
1.3.4 <i>Threats</i>	4
1.4 Estrutura	4
2 Estado da Arte	5
2.1 RV no Ensino e Educação	5
2.2 Conceção e Desenvolvimento de Atividades de RV	6
2.3 Arquiteturas e Especificações no Âmbito da RV	7

3	Especificação da Problemática	9
3.1	Linguagens, Arquiteturas e <i>Plug-ins</i>	10
3.1.1	A-Frame	10
3.1.2	H5P	10
3.1.3	JavaScript	11
3.2	Especificação do Problema	12
4	Metodologia	13
4.1	Ambiente de trabalho e Ferramentas	13
4.2	Fase de exploração	14
4.2.1	A-Frame	14
4.2.2	H5P	16
4.2.3	JavaScript	18
4.2.4	YAML	18
4.3	Fase de Desenvolvimento	18
4.3.1	Especificação da Estrutura <i>A-Frame</i>	18
4.3.2	Desenvolvimento do Código de Suporte à Prova de Conceito	20
4.3.3	Interatividade com Objetos	21
4.3.4	Processamento das Respostas de um Painel	22
4.3.5	Navegação no Ambiente	23
4.3.6	Construção do DOM	25
4.3.7	Especificação e Desenvolvimento da configuração YAML	26
5	Resultados Obtidos	29
5.1	Análise aos Resultados da PdC	29
5.2	Análise aos Resultados Globais	31
6	Conclusão e Trabalho Futuro	33
6.1	Integração com a tecnologia H5P	34
6.1.1	Biblioteca nativa de <i>A-Frame</i> para H5P	34
6.1.2	Adição de conectores <i>A-Frame</i> ou H5P	34
6.1.3	Desenvolvimento de um editor H5P	34
6.2	Refatoração do código <i>JavaScript</i> utilizado no PDC	35
6.3	Inclusão do cálculo de resultados	35
6.4	Desenvolvimento de um conector <i>xAPI</i>	35
6.5	Expansão do tipo de atividades	35

<i>CONTEÚDO</i>	xv
6.6 Desenvolvimento de múltiplos modos de execução da atividade	35
6.7 Diversificação do Ambiente RV	36
6.8 Biblioteca YAML	36
Bibliografia	37

Lista de Figuras

2.1	Liberdade de Movimentos 6-DOF num Ambiente Virtual [Com20]	7
4.1	Experiência com vários elementos <i>A-Frame</i>	16
4.2	Experiência com encapsulação de vários elementos e texto <i>A-Frame</i>	16
4.3	Diagrama de Classes da entidade <i>Painel</i>	19
4.4	Painel Base da Atividade <i>A-Frame</i>	20
4.5	Painel Completo da Atividade <i>A-Frame</i>	20
4.6	Fluxograma da função <i>select-on-click</i>	22
4.7	Fluxograma da função <i>end-panel</i>	23
4.8	Posicionamento dos Painéis na Atividade	24
4.9	Fluxograma da função <i>build-dom</i>	27

Listagens

3.1	Invocação da biblioteca <i>A-Frame</i>	10
4.2	Estrutura base de uma aplicação A-Frame	15
4.3	Encapsulação da biblioteca <i>A-Frame</i>	17
4.4	Invocação da biblioteca encapsulada <i>A-Frame</i> a partir de um projeto <i>H5P</i>	17
4.5	Registo de componente e inovação pela estrutura <i>A-Frame</i>	21
4.6	Definição da pose do painel para quatro instâncias	25
4.7	Definição do posicionamento do painel para quatro instâncias	26
4.8	Definição do posicionamento do utilizador para quatro instâncias do painel	26
4.9	Especificação do ficheiro de configuração YAML	28

Lista de Tabelas

3.1	Integrações e funcionalidades providenciadas pelo H5P [Mag21]	11
-----	-------------------------------------------------------------------------	----

Lista de Acrónimos

3D	Tridimensional
3-DOF	<i>Three Degrees of Freedom</i>
6-DOF	<i>Six Degrees of Freedom</i>
API	<i>Application Programming Interface</i>
ARSVR	<i>Adaptive Resolution Scheme for VR</i>
ASCII	<i>American Standard Code for Information Interchange</i>
DOM	<i>Document Object Model</i>
GC	<i>Google Cardboard</i>
H5P	<i>HTML5 Package</i>
HTML	<i>Hypertext Markup Language</i>
JS	<i>JavaScript</i>
JSON	<i>JavaScript Object Notation</i>
Moodle	<i>Modular Object Oriented Developmental Learning Environment</i>
PdC	Prova de Conceito
RA	Realidade Aumentada
RV	Realidade Virtual
SSL	<i>Secure Socks Layer</i>
Web	<i>World Wide Web</i>
WebVR	<i>Web-based Virtual Reality</i>
WebXR	<i>Web-based Extended Reality</i>
xAPI	<i>Experience API</i>
XML	<i>eXtensible Markup Language</i>
YAML	<i>YAML Ain't Markup Language</i>

Resumo

Com a proliferação de tecnologias de RV nos dias de hoje, a massificação das mesmas como instrumentos de ensino e aprendizagem é inevitável. Nesse cenário, levantam-se diversas questões e problemas relacionados, tanto com o seu uso como com a sua produção e desenvolvimento.

Esta dissertação foca-se na problemática da inclusão de tecnologias de RV no ensino à distância, por parte de utilizadores não-técnicos, com vista a enriquecer e motivar a transmissão de conhecimentos. É apresentada uma solução, suportada por uma Prova de Conceito, em que é demonstrada a possibilidade de, através da utilização de um conjunto de especificações claras e simples, gerar uma atividade RV com conteúdos dinâmicos e facilmente adaptáveis. É igualmente delineado um caminho à futura incorporação do trabalho, aqui apresentado, em sistemas de gestão de aprendizagem, através da combinação de diferentes arquiteturas e tecnologias de código livre e aberto.

Palavras chave: Realidade Virtual, e-Aprendizagem, A-Frame, H5P, Perguntas de Resposta Múltipla

Abstract

Virtual Reality and e-Learning

Specification of Multiple Choice Questions

As Virtual Reality (VR) technologies are currently spreading, it's rather inevitable that these become massively used as learning and teaching tools. In this scenario, many questions and related problems are emerging. Not only about how these technologies should be employed but also how they can be developed.

This thesis focuses on the problematic of giving non-technical users the ability of developing a feature-rich and motivating VR learning environment. A solution, backed by a working Proof of Concept, is presented. Through a set of clear and simple specifications, it's established how it's possible to generate a dynamic and adaptable VR activity. Additionally, a set of foundations are drawn out which will allow, through the combination of different open-source frameworks and technologies, incorporate the present work in Learning Management Systems.

Keywords: Virtual Reality, e-Learning, A-Frame, H5P, Multiple Choice Questions

1

Introdução

A aprendizagem é uma característica fundamental de qualquer ser vivo. É essa capacidade que lhe confere a habilidade de se adaptar a novos meios, aprender novas formas de sobrevivência e, naturalmente, transmitir os seus conhecimentos às gerações futuras, quer de forma genética, quer de forma interativa. Sem aprendizagem, o próprio ser humano não existiria.

1.1 Enquadramento

Ao longo do tempo, o homem procurou desenvolver, aperfeiçoar e diversificar as formas de aprendizagem. Na pré-história, o conhecimento era transmitido, oralmente, de geração em geração. Com a invenção da escrita, os livros e outros meios começaram a ser utilizados como suporte para o registo e transmissão de conhecimentos. Nos séculos XIX e XX, novas formas de comunicação, como a rádio, a televisão e, mais recentemente, as redes informáticas como a Internet, começaram a desempenhar um papel fundamental na aprendizagem e transmissão de conhecimentos. Esta evolução nos suportes e veículos transmissores de informação forçou, naturalmente, a própria evolução das

técnicas e formas de ensino tradicional e institucional, passando da mera aprendizagem passiva (expositiva) para um formato mais interativo, interrogativo e prático. Se há duas décadas, o fenômeno da Internet era, geralmente, considerado uma mera curiosidade, hoje em dia é um veículo fundamental de comunicação e uma ferramenta indispensável ao próprio funcionamento da sociedade. As metodologias de ensino começaram a abraçar estas novas tecnologias e temos assistido a uma evolução importante em todos os aspetos constituintes da aprendizagem. Os últimos desenvolvimentos alteraram, inclusive, as regras essenciais de aprendizagem coletiva como a partilha do mesmo espaço físico ou até mesmo a aprendizagem síncrona, por força das circunstâncias pandémicas nas quais temos vivido. As plataformas digitais de aprendizagem como, por exemplo, o *Modular Object Oriented Developmental Learning Environment* (Moodle) [Moo], *Blackboard* [Bla] ou *Claroline* [Cla], providenciam, neste contexto, um espaço de partilha e transmissão de conhecimentos unificado, permitindo aos seus intervenientes desenvolver sinergias e potenciar a aquisição de saberes.

Hoje em dia, as plataformas de aprendizagem assíncrona estão a proliferar, cada vez mais, em todos os níveis de ensino. Os intervenientes têm que se adaptar a estas novas realidades e atualizar as suas formas de interação com estes sistemas [PP15]. Esta adaptação nem sempre é fácil e existem aspetos que, por força dos meios utilizados, também têm que ser abordados de formas novas e diferentes. Os pressupostos tradicionais já nem sempre se aplicam e surgem novas dificuldades. Um claro exemplo é o distanciamento entre intervenientes que se verifica quando a aprendizagem é feita através de plataformas *online* como no caso do Moodle. Se, no ensino tradicional, o professor ou formador conseguia demonstrar e exemplificar determinados conceitos recorrendo a objetos físicos ou demonstrações reais que envolviam os estudantes na experiência, esta possibilidade é praticamente inexistente em plataformas de e-aprendizagem. É precisamente neste contexto que é preciso procurar soluções e recorrer a outros domínios tecnológicos que, de alguma forma, possam não só colmatar e superar estas limitações mas talvez até superar as experiências físicas realizadas em ambiente presencial.

A tecnologia de RV é capaz de simular um Ambiente Virtual Tridimensional (3D) através da combinação de sistemas gráficos com dispositivos de visualização adaptados. Esta tecnologia, no formato conceptual em que a reconhecemos hoje em dia, surgiu nos anos 60 com o desenvolvimento de protótipos com vista a criar ilusões tridimensionais [AF20]. Segundo os mesmos autores, o termo RV tornou-se popular na década de 80 sendo a responsabilidade maioritariamente atribuída ao fundador da empresa *VPL Research Inc*, Jaron Lanier, um pioneiro na comercialização de produtos para simular ambientes virtuais.

Nos últimos anos, o conceito de RV tem vindo a tornar-se cada vez mais popular e, com os custos associados a diminuírem drasticamente, acessível a praticamente todo mundo. Surgiram produtos como os das marcas *Oculus*¹ ou *Google Cardboard* (GC)² que permitem aos seus utilizadores experimentarem a sensação de imersão em ambientes virtuais a custos relativamente baixos.

Com a crescente disponibilidade de *hardware* adequado, tem surgido cada vez mais interesse por parte da comunidade científica, mas também por parte de empresas comerciais, em desenvolver normas, padrões e *Application Programming Interface* (API)s para uniformizar e assegurar a compatibilidade entre diferentes produtos no mercado. É neste contexto que surge o *Web-based Virtual Reality* (WebVR) que recentemente foi substituído pelas especificações mais abrangentes do *Web-based*

¹<https://www.oculus.com/>

²<https://arvr.google.com/cardboard/>

Extended Reality (WebXR)¹. Este último, na sua essência, descreve e define normas através das quais aplicações, desenvolvidas para a *World Wide Web* (Web), podem aceder aos diferentes sensores e produtos de RV.

Apesar dos desenvolvimentos galopantes na área do *hardware* e *software*, a conceção e customização de ambientes virtuais, mesmo os dirigidos ao ensino e formação, não está ao alcance de todos. Esta dificuldade, especialmente na atual conjuntura, com milhões de estudantes e professores forçados a abraçar o ensino à distância, é uma barreira à adoção em massa das tecnologias de RV como veículo de ensino à distância e instrumento de combate ao absentismo e diminuição na capacidade de aprendizagem.

1.2 Objetivos

Esta dissertação pretende encontrar possíveis soluções para que utilizadores não-técnicos possam tirar proveito das tecnologias mais recentes na área da RV de uma forma fácil e intuitiva e assim disporem de mais uma ferramenta de apoio ao ensino, independentemente da sua área de atuação. Para atingir este objetivo, irá ser produzida uma PdC, com recurso a tecnologias de código aberto e livre, de uma atividade de RV cujo propósito será o de apresentar um conjunto variável de questões de resposta dicotómica e/ou múltipla num ambiente virtual em que o utilizador final (aluno) possa interagir com o sistema de forma a selecionar a(s) opção(ões) correta(s) para cada questão colocada. Por outro lado, o utilizador (docente/tutor) deverá ser capaz de criar e/ou modificar a atividade sem necessitar de conhecimentos prévios de programação. Finalmente também é pretendido integrar essa atividade num sistema de gestão de aprendizagem como o Moodle, por exemplo.

1.3 Análise S.W.O.T.

Aquando do planeamento e organização desta dissertação, foi efetuada uma análise S.W.O.T.² para delinear a melhor estratégia de organização e desenvolvimento do trabalho proposto. Desta reflexão, emergiram os seguintes pontos-chave:

1.3.1 *Strengths*

O desenvolvimento deste trabalho não está dependente de fatores externos como o acesso à bases de dados ou resultados de inquéritos. O ónus do desenvolvimento recai, única e exclusivamente no autor do trabalho. A área sobre a qual o trabalho será realizado, é lhe muito próxima visto incidir sobre a temática do ensino e educação, atividade em que se vê envolvido diariamente, quer a nível profissional, quer a nível pessoal. Esta proximidade deverá proporcionar motivação ao longo do tempo de desenvolvimento desta tarefa. A inclusão de uma PdC, é uma mais-valia importante dado que permite demonstrar, na prática, uma possível solução ao problema proposto.

¹<https://www.w3.org/TR/webxr/>

²Do Inglês - **S**trengths, **W**eaknesses, **O**pportunities e **T**hreats

1.3.2 *Weaknesses*

As principais fraquezas no desenvolvimento desta dissertação centram-se, decididamente, no autor. Apesar de possuir experiência na área da programação com diversas linguagens de programação, por motivos circunstanciais, *JavaScript* (JS) não é uma delas. É difícil prever a adaptação e a curva de aprendizagem da linguagem que terá de ser utilizada durante o desenvolvimento da PdC. A manutenção da dedicação e do empenho ao longo do trabalho também é um fator interno de risco visto que o autor já não é estudante a tempo inteiro há muitos anos.

1.3.3 *Opportunities*

Este é, francamente, o melhor tempo possível para a investigação nesta área. A situação actual em que vivemos despoletou uma mudança de paradigma no ensino e educação a nível planetário. Há muito tempo que a sociedade tem vindo a caminhar, lentamente, em direção à inclusão do ensino não presencial, mas foi apenas devido às súbitas circunstâncias conjunturais que, de um momento para outro, despoletaram um interesse global e real em explorar muitas das tecnologias emergentes nesta área.

1.3.4 *Threats*

A principal ameaça ao desenvolvimento desta tese é apenas o tempo disponível, que poderá flutuar imenso de período em período, de semestre a semestre e que não pode ser controlado pelo autor.

1.4 Estrutura

O presente trabalho encontra-se dividido em seis capítulos principais.

- **Capítulo 1** - Apresentação, enquadramento e definição dos objetivos principais deste trabalho;
- **Capítulo 2** - Revisão ao estado da arte, nas várias áreas que são abordadas por este projeto, nomeadamente, a RV como ferramenta no ensino e aprendizagem; o uso da RV por parte de empresas comerciais; atuais soluções usadas para a simplificação de criação de conteúdo; e o estado atual das normas e especificações na área;
- **Capítulo 3** - Especificação do problema a tratar e identificação das linguagens, normas, arquiteturas e *plug-ins* a serem utilizados para desenvolver uma solução funcional;
- **Capítulo 4** - Desenvolvimento de uma possível solução ao problema proposto, descrevendo todo o processo e metodologias empregues durante a sua construção;
- **Capítulo 5** - Aferição dos resultados obtidos através da solução anteriormente proposta;
- **Capítulo 6** - Ideias e considerações delegadas para trabalho futuro a realizar nesta área.

2

Estado da Arte

As tecnologias de RV têm visto um desenvolvimento enorme nos últimos anos e, atualmente, muito devido à conjuntura pandêmica, tornaram-se ainda mais apetecíveis. Este capítulo pretende providenciar uma breve panorâmica sobre o estado atual da investigação no campo da realidade virtual aplicada ao ensino e focar-se em aspetos cruciais que este trabalho pretende resolver, como é o caso da simplificação da criação de atividades RV.

2.1 RV no Ensino e Educação

Segundo [NAOM20], a reação, por parte dos estudantes do ensino superior, quando expostos à tecnologia de RV, pode variar imenso de acordo com as metodologias empregues e em que fases do ensino a mesma é adotada. Por outro lado, estudos como [YGG20] focam-se, principalmente na resolução da problemática da falta de interação entre os estudantes e professor quando em contexto de aprendizagem remota. Os autores evidenciam vários benefícios que a utilização de tecnologia de RV pode trazer para combater os típicos problemas associados ao ensino à distância, nomeadamente

o *aborrecimento* e *tédio* que alguns estudantes evidenciam quando não têm contacto direto com o tutor. No estudo são analisados os resultados obtidos através do uso de uma aplicação (AulaVR) para simular uma sala de aula em ambiente RV.

A RV não é uma área recente por si só. Tem sido utilizada há dezenas de anos (e há quem afirme há milhares de anos, quando o termo é interpretado no sentido mais lato, i.e., englobando a própria *imaginação* humana [OMH]) no ensino/formação, como por exemplo, na indústria de aviação comercial, para o treino de pilotos em simuladores de voo.

Outras áreas em que a RV também tem ganho terreno, há muitos anos, é nos campos da medicina e química, onde o recurso às tecnologias RV para treinar os estudantes em diversos tópicos se tem mostrado bastante positivo [PD21].

Atualmente, existem muitos projetos académicos e comerciais nesta área. Por exemplo, *Imikode* [BNO⁺19] é um protótipo de um jogo baseado em RV, desenvolvido com o intuito de transmitir aos alunos, conceitos-chave da programação orientada a objetos, de uma forma mais apelativa e envolvente. O jogo consiste em apresentar ao utilizador, por meio de uma história, conceitos como *objetos*, *métodos* e *argumentos* de uma forma que envolva o estudante e explique mais facilmente as relações que existem entre eles.

Existem igualmente diversos trabalhos na área da aprendizagem de línguas onde são apresentados benefícios da utilização da RV para melhorar a pronúncia e fluidez no uso da língua inglesa, por exemplo. [NMI⁺20] propõem uma arquitetura conceptual baseada na aplicação *Talk VR*, ainda em desenvolvimento, que combina o uso da RV com análíticas de aprendizagem para melhorar o desempenho dos estudantes.

O Moodle, apresentou há poucos anos, a sua intenção em debruçar-se mais sobre a aplicação da tecnologia RV ao seu sistema de e-aprendizagem [OMH] e tem vindo, desde daí, através do seu responsável por essa área, desenvolver avanços muito interessantes que, neste momento, incluem o desenvolvimento de um módulo Moodle de RV, totalmente integrado no sistema, que converte exercícios do tipo *Quiz*, numa experiência virtual e interativa [Oke21]. O módulo consiste numa espécie de jogo, em que o utilizador é confrontado com uma pergunta e cujas opções de resposta flutuam, dentro do ângulo de visão do jogador, como se se tratassem de naves espaciais que podem ser selecionadas, via a utilização de instrumentos como o GC, simulando disparos de uma arma imaginária.

Outras empresas comerciais, como a *MindBox VR*¹, apostam no uso da RV para oferecer serviços de treino de colaboradores nas mais diversas áreas e situações. Dispõem de um vasto leque de simulações pré-elaboradas que, asseguram os seus criadores, melhoram o empenho e a capacidade de aprendizagem dos estudantes, aumentam o ritmo de aprendizagem, e permitem desenvolver laços mais profundos entre o trabalhador e a empresa.

2.2 Conceção e Desenvolvimento de Atividades de RV

A grande maioria da investigação, na área da RV aplicada ao ensino, aparenta estar dirigida ao utilizador final, i.e., o potencial recetor do conhecimento. Da mesma forma, muitas das empresas

¹<https://mindboxvr.com/>

comerciais na área da RV focam-se em providenciar serviços *prontos-a-usar* pelos adquirentes sem evidenciar muito a possibilidade de adaptar ou estender as atividades por pessoas leigas na temática.

Um dos pontos abordados por [AML16] refere a problemática do envolvimento de educadores sem *background* técnico no desenvolvimento de atividades de RV. A proposta nesse estudo, referente às aplicações por eles desenvolvidas, inclui a simplificação da criação de ambientes virtuais através da inclusão do conceito de livros *tangíveis*. Neste cenário, o professor apenas tem de adicionar páginas de texto à simulação ficando o conteúdo disponível, no ambiente virtual, através desse livro.

A problemática no processo de construção também é levantado por [GBJK19] num projeto de RV aplicado ao ensino sobre as estruturas moleculares na área de química. Neste caso particular, a solução proposta passou por desenhar as moléculas químicas numa aplicação externa e depois submetê-las *online* através de um serviço como o providenciado por *Sketchfab*¹, uma empresa que desenvolveu uma plataforma de partilha, compra e venda de modelos 3D e que possui um sistema de visualização em WebVR, o que dispensa a necessidade de desenvolver um ambiente virtual customizado.

2.3 Arquiteturas e Especificações no Âmbito da RV

Em 2019, [SC19] analisaram a arquitetura *A-Frame* [AFc] como ferramenta de desenvolvimento de ambientes virtuais para aplicações WebVR por parte de indivíduos não especializados na área com resultados positivos. A tecnologia permite, de facto, desenvolver atividades simples em 3D com necessidades limitadas de conhecimentos em JS. A tecnologia foi utilizada para criar uma visita virtual ao Museu Monográfico de Conimbriga.

Outro trabalho, desenvolvido por [HKS20], identifica as atuais limitações das tecnologias de WebVR e WebXR (utilizadas em arquiteturas como o *A-Frame*) no que respeita à locomoção do utilizador dentro de um ambiente virtual. Apesar de existir suporte nas especificações do WebXR, surgem limitações quando são utilizados dispositivos móveis como *interface* para aplicações de RV. O constrangimento mais visível é a falta de suporte para *Six Degrees of Freedom* (6-DOF) i.e., um verdadeiro ambiente virtual caracteriza-se por suportar todos os seis tipos de movimentos naturais conforme a Figura 2.1.

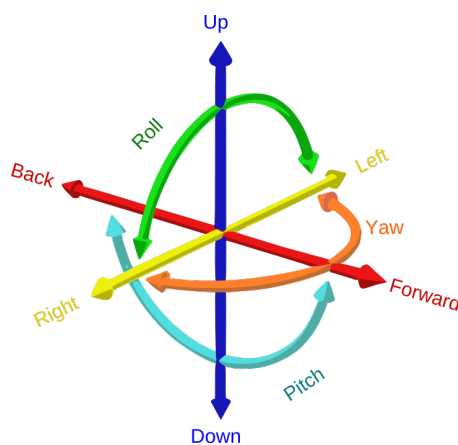


Figura 2.1: Liberdade de Movimentos 6-DOF num Ambiente Virtual [Com20]

¹<https://sketchfab.com/>

No entanto, quando são utilizados *smartphones*, estes estão limitados a apenas *Three Degrees of Freedom* (3-DOF), sendo que os movimentos de deslocação no espaço (frente/trás, cima/baixo, direita/esquerda) são ignorados. [HKS20] propõem uma solução, na forma de componentes, desenvolvidos ao abrigo do projeto *LIRKIS*¹, que permitem ultrapassar esta limitação com a deteção de movimentos de deslocação pelos dispositivos móveis.

A utilização das especificações de WebXR para providenciar conteúdos RV através de simples exploradores de Web aparenta ser o caminho para a globalização do uso da simulação de ambientes virtuais no futuro. Uma outra limitação dessa metodologia, no entanto, é evidenciada por [PSM21], que refere quebras de responsabilidades e diminuição das taxas de refrescamento quando a tecnologia é combinada com a simulação de ambientes virtuais multi-utilizador, i.e., quando o processamento de um ambiente de RV é construído numa lógica de arquitetura do tipo *Cliente-Servidor* em que todos os dispositivos têm receber e enviar dados de e para um servidor central. Os autores do trabalho propõem o uso de um algoritmo por eles criado: *Adaptive Resolution Scheme for VR* (ARSVR) que, através do ajuste automático da resolução da simulação para cada dispositivo, demonstra conseguir colmatar as limitações encontradas.

¹Laboratory of Intelligent inteRfaces of Communication and Information Systems, acessível em: <http://lirkis.kpi.fei.tuke.sk/>

3

Especificação da Problemática

O objetivo deste trabalho é conseguir conciliar as vantagens pedagógicas da tecnologia de RV com as necessidades de flexibilização e facilidade na configuração de exercícios e atividades do ponto de vista do utilizador. No fundo, pretende-se que um utilizador não técnico consiga gerar, autonomamente, sem conhecimentos técnicos sobre programação, atividades RV pedagógicas de forma a que as mesmas possam, futuramente, ser integrados em plataformas de e-aprendizagem.

Para este projeto, cujo âmbito é claramente marcado pelo espírito de código livre e aberto, foi decidido, baseado nas pesquisas realizadas no capítulo anterior, avançar com um conjunto de linguagens, especificações e estruturas que a seguir são descritas.

3.1 Linguagens, Arquiteturas e *Plug-ins*

3.1.1 A-Frame

O projeto *A-Frame* é uma *arquitetura*¹ cujos objetivos principais incluem facilitar o desenvolvimento de ambientes de RV através da utilização de uma linguagem simples, intuitiva e largamente adotada, denominada *Hypertext Markup Language* (HTML). A iniciativa também aponta providenciar suporte em múltiplos sistemas operativos e funcionar com a maioria dos dispositivos RV, incluindo soluções como GC.

De acordo com a documentação disponível em [AFc], o *A-Frame* assenta na biblioteca *Three.js* [Thr], expandindo-a e providenciando uma verdadeira arquitetura entidade-componente que é declarativa, estruturada e reutilizável. Os autores evidenciam a sua dedicação em garantir uma elevada performance da biblioteca em qualquer sistema dado que a mesma foi otimizada, desde o início, para aplicações WebXR, realizando todas as operações sobre objetos 3D na memória com poucos ou nenhuns desperdícios. A biblioteca também inclui um inspetor nativo que deixa transparecer a natureza *open-source* do projeto pois permite a qualquer utilizador inspecionar, verificar e, inclusive, modificar componentes e objetos, em tempo real, em qualquer aplicação que use esta tecnologia.

A inclusão da biblioteca dentro de um projeto consegue-se, facilmente, através da simples declaração da mesma dentro de um documento HTML conforme a Listagem 3.1

```
<html>
  <head>
    <meta charset="UTF-8" />
    <script src="https://aframe.io/releases/1.2.0/aframe.min.js">
    </script>
  </head>
  <body>
  </body>
</html>
```

Listagem 3.1: Invocação da biblioteca *A-Frame*

Originalmente desenvolvido no seio da empresa *Mozilla*², o projeto agora é mantido pelos seus co-autores, através da empresa *Supermedium*³.

3.1.2 H5P

HTML5 Package (H5P)⁴ é um *plug-in* desenvolvido para sistemas de gestão de conteúdos como é o caso do Moodle, *Drupal*, *WordPress*, *Blackboard*, entre outros. A ferramenta tem como propósito

¹Do Inglês - Framework

²<https://www.mozilla.org>

³<https://www.supermedium.com>

⁴<https://h5p.org>

primordial, assistir os criadores de conteúdos em desenvolver, de uma forma fácil, projetos e distribuí-los num formato compacto e uniformizado. Nota-se, através da documentação disponibilizada, que é dada muita ênfase aos vários formatos de licenciamento e proteção dos direitos de autor. É uma ferramenta especialmente útil para educadores, professores e tutores visto que a grande maioria das atividades são de carácter educacional e pedagógica. Todos os tipos de atividades vêm acompanhadas por um editor que permite ao seu utilizador desenvolver, rapidamente e sem grandes conhecimentos técnicos, atividades específicas, como por exemplo, vídeos e imagens interativas, questionários, gráficos, etc.

A plataforma H5P, no entanto, também oferece serviços pagos que consistem, maioritariamente, na disponibilização de espaço para alojar os projetos desenvolvidos pelos utilizadores, capacidade de produzir relatórios a partir das interações dos alunos nas atividades e suporte personalizado [Mag21] conforme podemos verificar na Tabela 3.1.

Forma de integração e plataformas / características		Alojamento	Relatórios	Suporte	Mantido pela equipa principal	Atualizado	Subscrição necessária
Ligação direta ou incorporada	H5p.com	Sim	Não	Sim	Sim	Sim	Sim
Integração com plataformas de ensino	Canvas	Sim	Sim	Sim	Sim	Sim	Sim
	Brightspace	Sim	Sim	Sim	Sim	Sim	Sim
	Blackboard	Sim	Sim	Sim	Sim	Sim	Sim
	Moodle	Sim	Sim	Sim	Sim	Sim	Sim
<i>Plug-in gratuito</i>	Drupal	Não	Não	Não	Sim	Sim	Não (gratuito)
	Moodle	Não	Limitado	Não	Sim	Sim	Não (gratuito)
	WordPress	Não	Não	Não	Sim	Sim	Não (gratuito)

Tabela 3.1: Integrações e funcionalidades providenciadas pelo H5P [Mag21]

3.1.3 JavaScript

O nome oficial da especificação da linguagem JS é *ECMAScript*. É uma linguagem de programação por excelência em aplicações Web mas não só. É normalmente conhecida por ser uma linguagem *client-side*, i.e. é executada no sistema do cliente quando uma aplicação é invocada a partir de uma página de Web, por exemplo. Neste momento, segundo [Sur21b], esta linguagem é empregue em 97.6% das páginas Web existentes. No entanto, a mesma também pode ser utilizada do lado servidor (*server-side*) onde é usada para interagir com o próprio sistema e/ou outros componentes *back-tier* como, por exemplo, no acesso e manipulação de bases de dados. Normalmente, estas

funcionalidades são providenciadas via *Node.js*¹, uma compilação da linguagem JS para uso fora do ambiente Web tradicional.

Segundo [SR17], a portabilidade e a expressividade têm sido fatores de sucesso em tornar esta uma das linguagens mais populares hoje em dia. As suas capacidades dinâmicas em adicionar e remover atributos de objetos durante a sua execução é, segundo estes autores, uma mais valia.

3.2 Especificação do Problema

Este problema pode dividir-se em quatro partes fundamentais, nomeadamente:

1. criar um ambiente de RV imersivo, interativo, adaptável e educativo que possa ser explorado com recursos simples e de baixo custo, como por exemplo, através do uso de GC;
2. o ambiente definido no ponto anterior tem que permitir que um utilizador (do tipo docente) possa, de uma forma simples, fácil e sem necessidade de dominar uma linguagem de programação específica, definir uma atividade composta por uma ou mais perguntas, de resposta única ou múltipla, constituída por uma ou mais opções de resposta e cujos resultados sejam passíveis de serem registados;
3. um utilizador (do tipo discente) deve poder interagir com o ambiente preparado anteriormente, responder a uma questão através da seleção de uma ou mais opções, marcar a questão como concluída (ou não) e mover-se para a próxima questão sem necessidade de meios auxiliares de controlo para além da visão até que toda a atividade seja completada.
4. desenvolver várias bibliotecas em H5P de forma a integrar a PdC, num sistema de gestão de aprendizagem, como o Moodle.

¹<https://nodejs.org>

4

Metodologia

A Resolução do problema proposto passou por dividir o trabalho em duas fases: **exploração** e **desenvolvimento**.

Esta metodologia deveu-se a diversos fatores entre os quais o facto de que o autor não tinha conhecimentos prévios em nenhuma das áreas abordadas mas também pela necessidade de testar diversas tecnologias alternativas para aferir quais as mais adequadas para resolver o desafio enunciado.

4.1 Ambiente de trabalho e Ferramentas

O ambiente de desenvolvimento escolhido foi o *Code - OSS*¹, uma versão de código aberto do popular editor *Visual Studio Code*, ambos da *Microsoft*. Para este trabalho específico, foram ainda

¹<https://github.com/microsoft/vscode>

incluídas várias extensões, nomeadamente *Prettier - Code formatter*¹, para a formatação automática do código fonte e *GitLab Workflow*², para a integração do desenvolvimento, via Git³.

Foi igualmente criado um repositório privado no popular GitLab⁴ para sincronizar o trabalho a partir dos diversos postos de trabalho que foram utilizados no decurso desta dissertação. Por outro lado pretendeu salvaguardar-se possíveis problemas que pudessem surgir durante o desenvolvimento do projeto ao adoptar estas estratégias de versionamento.

É de notar que não são necessárias ferramentas extra para o desenvolvimento com a arquitetura *A-Frame* para além de um editor e um navegador Web, mas para o carregamento das fontes e possíveis texturas, é necessário executar o código dentro de um servidor de Web. O carregamento de bibliotecas externas como o *afree-environment-component* obriga, adicionalmente, a recorrer a um certificado *Secure Socks Layer* (SSL) para que a estrutura aceite o mesmo. Para esta PdC recorreu-se a um servidor Web simples, desenvolvido na linguagem *python* e a um certificado gerado no próprio sistema.

Finalmente, durante a exploração do H5P, também foi montada uma instância "dockerizada"⁵ do Moodle para isolar as experiências do resto do sistema.

4.2 Fase de exploração

4.2.1 A-Frame

Em primeiro lugar, foi analisada a estrutura empregue pelo *A-Frame* e realizaram-se diversas experiências na criação de ambientes virtuais assim como a colocação de diversos objectos.

A estrutura do *A-Frame* é bastante linear e de fácil compreensão. Um documento é sempre iniciado pela declaração de uma cena do tipo `<a-scene>` seguindo-se a definição do ambiente envolvente como o céu `<a-sky>` e, opcionalmente, um chão ou plano `<a-plane>`.

Existem várias entidades, chamadas primitivas, pré-configuradas na biblioteca para mais fácil utilização e configuração:

- `<a-box>`
- `<a-circle>`
- `<a-triangle>`
- `<a-cone>`
- `<a-sphere>`
- `<a-cylinder>`

¹<https://open-vsx.org/extension/esbenp/prettier-vscode>

²<https://open-vsx.org/extension/GitLab/gitlab-workflow>

³Global Information Tracker

⁴<https://gitlab.com>

⁵<https://www.docker.com/>

- `<a-plane>`
- `<a-tetrahedron>`
- `<a-octahedron>`
- `<a-dodecahedron>`
- `<a-torus>`
- `<a-torusknot>`

A base de qualquer uma das primitivas acima enunciadas é sempre uma entidade representada por `<a-entity>`. Este objeto é a base fundamental de construção no *A-Frame*. Existe, inclusive, a possibilidade de registar e partilhar novas primitivas através da inclusão de bibliotecas partilhadas por membros da comunidade como é o caso de `<a-ocean>`, um objeto que representa um oceano com várias opções de configuração e que foi desenvolvido pela comunidade e incluído na biblioteca *aframe-extras*¹.

A estrutura base pode então ser representada conforme a Listagem 4.2 que gera um ambiente simples com um cubo. É de notar a facilidade com a qual o *A-Frame* resolve situações em que são omitidos outros elementos base como a posição e especificação da(s) câmara(s) e/ou iluminação direcional ou ambiental.

```
<html>
  <head>
    <meta charset="UTF-8" />
    <script src="https://aframe.io/releases/1.2.0/aframe.min.js">
    </script>
  </head>
  <body>
    <a-scene>
      <a-sky></a-sky>
      <a-box></a-box>
    </a-scene>
  </body>
</html>
```

Listagem 4.2: Estrutura base de uma aplicação A-Frame

É de notar que a estrutura no *A-Frame* é muito flexível. Na figura anterior, podemos verificar que além dos elementos de câmara e luz que não foram explicitamente declarados, também não foi incluída qualquer declaração relativa aos controlos RV ou Realidade Aumentada (RA). No entanto, tal como já foi referido em relação aos outros elementos, a biblioteca inclui automaticamente estes controlos com configurações padrão.

Durante as primeiras experiências, foram testados vários aspectos na criação de ambientes virtuais como a encapsulação de elementos dentro dos seus parentes (Figura 4.1) e a utilização de texto

¹<https://github.com/n5ro/aframe-extras>

(Figura 4.2). Neste último, veio a verificar-se que a inclusão de texto não é simples quando o mesmo é redigido na língua portuguesa uma vez que, a biblioteca do *A-Frame* não suporta, nativamente, caracteres latinos, i.e., com acentos. Foi necessário gerar uma matriz com os caracteres *American Standard Code for Information Interchange* (ASCII) normalmente utilizados na língua portuguesa¹ para que fosse possível a inclusão de acentuação nos textos a serem utilizados nos ambientes virtuais. Um aspeto importante, dados os objetivos definidos para este trabalho.

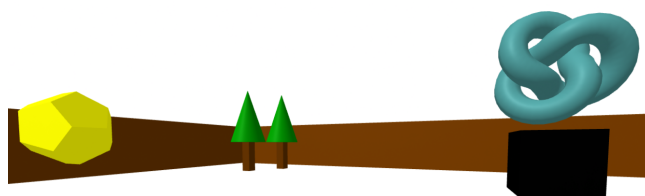


Figura 4.1: Experiência com vários elementos *A-Frame*

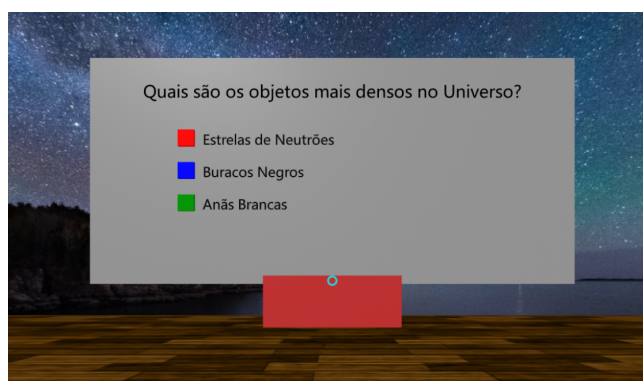


Figura 4.2: Experiência com encapsulação de vários elementos e texto *A-Frame*

Nesta fase, também foram realizadas diversas experiências com interatividade entre o utilizador e os objectos dentro do ambiente virtual, mas estas serão abordadas mais especificamente na secção dedicada à linguagem de programação JS.

4.2.2 H5P

Durante as explorações iniciais do *plug-in* H5P, verificaram-se alguns problemas importantes. O primeiro obstáculo encontrado foi o facto da documentação disponibilizada estar incompleta e conter várias lacunas. Por exemplo, quando um *plug-in* personalizado é gerado para ser enviado para um sistema de gestão de conteúdos, como é o caso do *Moodle*, o ficheiro H5P (que na realidade é um ficheiro comprimido *zip*) tem de ser criado com a opção de **Guardar estrutura de pastas no nome do ficheiro**. É um requerimento que não vem mencionado na documentação oficial e mesmo quando principiantes recorrem à ajuda da comunidade, esse pormenor é muitas vezes ignorado, facto que cria, de certa forma, uma barreira à entrada de novos indivíduos que pretendam explorar e desenvolver atividades dentro da infra-estrutura do H5P. Não se compreende se estas lacunas são

¹<https://msdf-bmfont.donmccurdy.com>

resultantes do facto do H5P fornecer, mediante subscrição paga, suporte e acesso a uma plataforma de comunicação diferenciada em relação à versão livre.

A estrutura modular do H5P é resultado da utilização de pequenos blocos de construção que são as chamadas bibliotecas. Para a construção de um pacote simples de H5P, são sempre necessários, pelo menos três componentes:

- biblioteca em JS com o código necessário para correr o projeto. Por ex: `exercicio.js`;
- ficheiro `semantics.json` descrevendo a estrutura de dados que poderá ser editada posteriormente;
- ficheiro `library.json` descrevendo o pacote, as suas dependências assim como a sua versão.

O *plug-in* H5P não permite a invocação de bibliotecas externas. Visto que a arquitetura *A-Frame* necessita, no mínimo, da biblioteca `aframe.min.js` para gerar um ambiente RV, optou-se por, primeiramente, construir um pacote H5P apenas com uma cópia da biblioteca *A-Frame* encapsulada dentro de uma função H5P como se pode observar na Listagem 4.3.

```
H5P.AFrame = (function () {
  function H5PAFrame() {
    // código da biblioteca A-Frame
  }
})
```

Listagem 4.3: Encapsulação da biblioteca *A-Frame*

De seguida, foi criado um novo pacote H5P, já com uma experiência simples de *A-Frame* invocando a biblioteca anterior dentro do projeto de exploração conforme a Listagem 4.4.

Após algumas experiências, verificaram-se dois problemas adicionais. Em primeiro lugar, não é possível ter mais do que um ambiente *A-Frame* na mesma página. A biblioteca gera uma exceção e termina o processo. Isso é resolúvel através da utilização de *iframes*. No entanto, o uso destes

```
var libaf = H5P.AFrame.toString();
var width = this.options.width;
var height = this.options.height;

$("head").prepend(
  "<script>" +
  libaf + "</script>" +
  "<script>H5PAFrame();</script>" +
  "<script>AFRAME.registerComponent(){
  // Código para A-Frame
  }" + "</script>");
```

Listagem 4.4: Invocação da biblioteca encapsulada *A-Frame* a partir de um projeto *H5P*

últimos, cria um segundo problema que é o facto do ambiente *A-Frame* não transmitir o tamanho de ocupação do ecrã ou *iframe*. Desta forma, o plugin H5P não consegue detetar e, conseqüentemente, definir, o tamanho inicial de publicação da atividade. Disto resulta, que um exercício *A-Frame* publicado via uma atividade H5P, num sistema de e-aprendizagem tipo *Moodle*, irá aparecer com um tamanho inicial de 0 pixels.

Após entrar em contacto com a comunidade H5P [AFb], foram confirmados estes problemas e a solução, sugerida por um investigador a trabalhar na mesma área, foi a de recorrer a *iframes* encapsulados, sendo que o *iframe parent* definiria o tamanho da atividade. Uma solução pouco elegante e de recurso mas que ele próprio tinha adoptado.

4.2.3 JavaScript

Toda a exploração inicial acabou por ter impacto na aprendizagem desta linguagem de programação, especialmente pelo facto do autor não ser versado na mesma e nunca a ter utilizado anteriormente, quer a nível profissional, quer a nível académico ou pessoal. Foi uma curva de aprendizagem muito íngreme mas satisfatória.

Para o presente trabalho, foram exploradas as bases da linguagem, focando especialmente as extensões da linguagem orientadas à interação com o *Document Object Model* (DOM) presente nas páginas de HTML.

4.2.4 YAML

Visto o objetivo primordial ser a criação de uma atividade RV dinâmica e facilmente editável por um leigo, rapidamente se tornou claro que seria necessário providenciar uma configuração externa que pudesse ser carregada pelo ambiente *A-Frame* durante a fase de inicialização. Foram analisadas várias opções sendo que o formato *YAML Ain't Markup Language* (YAML) se destacou pela facilidade de leitura, capacidade de serialização e por dispor da possibilidade de concatenar múltiplos documentos de configuração num só. Em trabalho futuro, isto poderá ser muito útil se for desejado especificar o próprio ambiente RV. A única grande desvantagem é que, neste momento, o carregamento de um ficheiro de configuração em YAML não é suportado nativamente pela linguagem JS pelo que foi necessário recorrer a uma biblioteca externa para providenciar essa funcionalidade.

4.3 Fase de Desenvolvimento

4.3.1 Especificação da Estrutura *A-Frame*

Este projeto iniciou-se com a especificação da estrutura *A-Frame* que seria necessária para criar um ambiente virtual em que um dado utilizador pudesse interagir com objetos que o rodeassem de uma forma fácil e simples. Foi dada primazia à possibilidade de utilizar meios de RV simples, de baixo custo e que não necessitassem de controladores específicos como, por exemplo, o GC.

Seguindo a especificação definida na Secção 3.2, começou por ser estruturado o ambiente virtual

de acordo com o diagrama de classes representado na Figura 4.3 que mostra, esquematicamente, as dependências entre os vários objetos que constituem a entidade *Panel*.

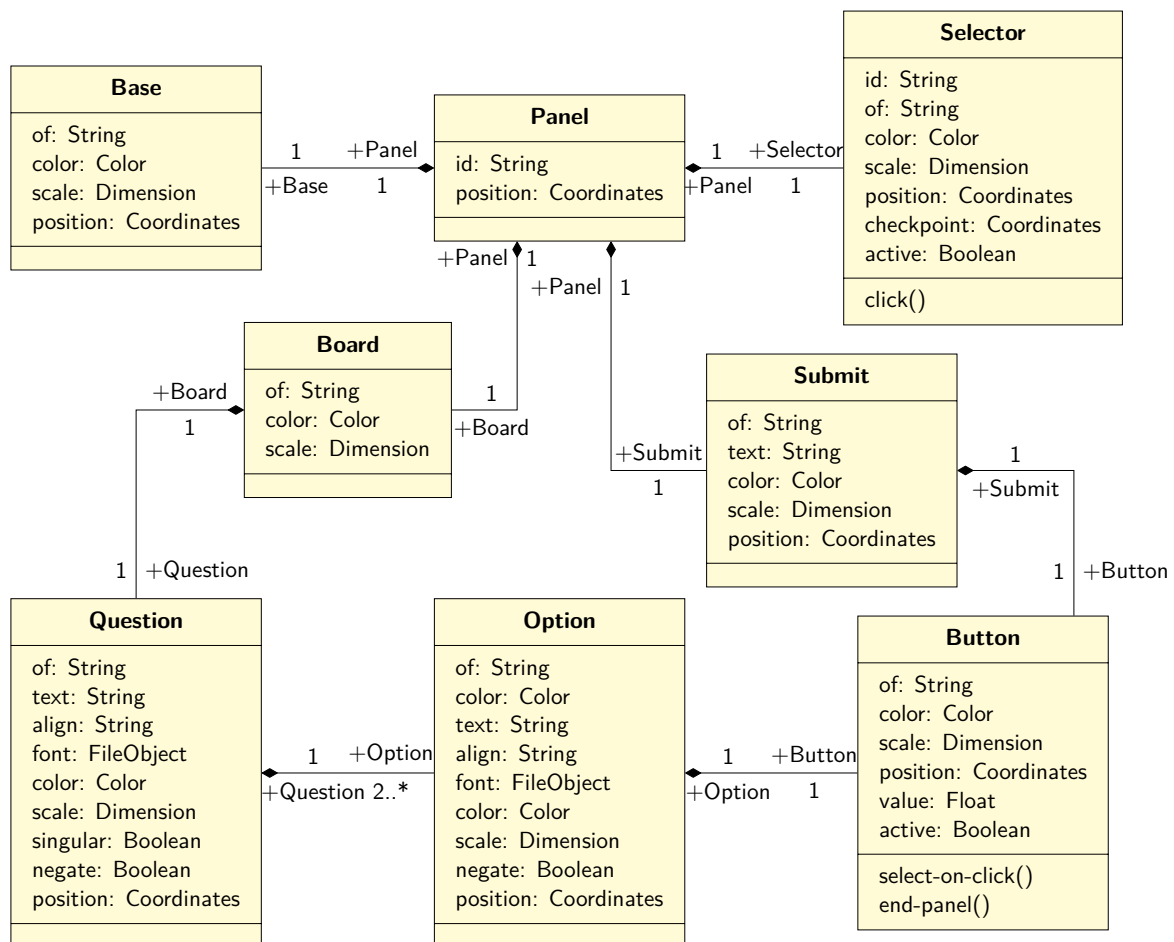


Figura 4.3: Diagrama de Classes da entidade *Panel*

Como podemos observar, um *Panel* é constituído por quatro objetos sendo que é o *Board* que irá albergar as questões fornecidas. Esta relação de dependência na estrutura é importante pelo facto do *A-Frame* permitir a utilização de posicionamento relativo de objetos em relação a outros quando os mesmos são nós dependentes. Isto é, o atributo *position*, em todos os objetos subordinados ao *Board*, será sempre relativo ao próprio quadro.

Uma vez construído o *Panel*, procedeu-se à sua colocação, manual, num ambiente virtual para verificar a sua funcionalidade como podemos observar na Figura 4.4. Após alguns testes, concluiu-se que a distância ideal entre o utilizador e o painel seriam 4 metros, de acordo com o dispositivo GC usado durante o desenvolvimento. Medida que mais tarde foi aplicada a todos os painéis posteriores. Para acelerar o desenvolvimento, optou-se por incluir uma biblioteca, apelidada de *aframe-environment-component* e cujo desenvolvimento e manutenção tem sido realizada pela própria empresa *Supermedium*, já referenciada anteriormente.

Esta biblioteca providencia uma forma mais fácil de gerar diversos ambientes envolventes e que também poderá facilitar, em trabalho futuro, que um utilizador possa especificar, no ficheiro de configuração, o tipo de ambiente que quer utilizar para a sua atividade.

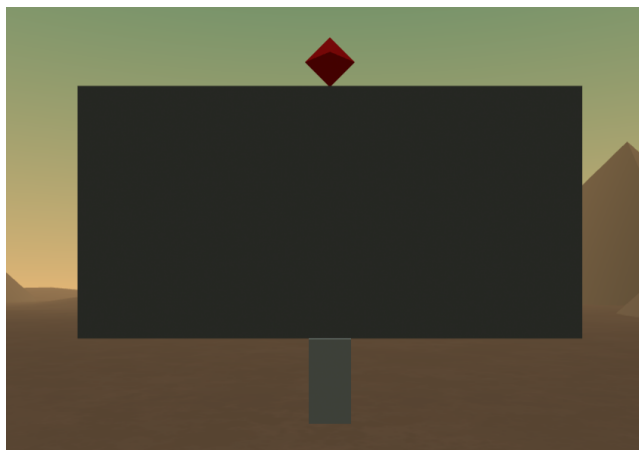


Figura 4.4: Painel Base da Atividade *A-Frame*

Usando a entidade *Board* como referência posicional, procedeu-se então à criação e colocação, manual, dos nós dependentes até que se obtivesse um painel visualmente exemplificativo da atividade conforme a Figura 4.5.



Figura 4.5: Painel Completo da Atividade *A-Frame*

Até esse momento, ainda não tinha sido desenvolvida qualquer interatividade ou código JS para além do estritamente necessário para executar a atividade.

4.3.2 Desenvolvimento do Código de Suporte à Prova de Conceito

Como já foi referenciado, a linguagem de desenvolvimento preferencial para ambientes construídos em *A-Frame* é o JS. No entanto, existem algumas peculiaridades que diferem do modo tradicional de colocação e uso de *scripts*, em páginas de Web. Em primeiro lugar, segundo as boas práticas sugeridas pela documentação oficial [AFa], o código deve ser sempre colocado no início do projeto para garantir que o seu carregamento se inicie antes do ambiente *A-Frame* ser inicializado. Em segundo lugar, e talvez o mais importante, seja o facto de que não é, de todo, recomendado que o código seja executado diretamente na página Web mas sim registado como um ou vários

componentes *A-Frame*, que depois são invocados, quando necessários, pelo ambiente de RV.

De forma prática, isto significa que funções ou partes do código que devam ser inicializados automaticamente ao iniciar o ambiente virtual, tem de ser encapsulados num componente *A-Frame* e invocados logo nas declarações iniciais da estrutura conforme exemplificado na Listagem 4.5.

```
// Código JS
AFRAME.registerComponent("build-dom", {
  init: function () {
    /* Código da Função */
  }
}
...
<a-entity id="exercicio" build-dom>
```

Listagem 4.5: Registo de componente e inovação pela estrutura *A-Frame*

Para este trabalho foram definidos quatro componentes fundamentais para o funcionamento da PdC:

- **build-dom** → Componente principal que irá gerar as entidades e os seus componentes de acordo com uma configuração YAML fornecida;
- **select-on-click** → Componente que irá definir como as propriedades dos objetos se irão alterar quando sujeitos à interação do utilizador;
- **end-panel** → Componente de registo de conclusão de um painel, i.e., o componente responsável por processar as respostas dadas pelo utilizador e fechar o painel;
- **checkpoint** → Componente de navegação.

As secções seguintes irão descrever cada um dos componentes, pela ordem em que foram desenvolvidos ao longo do trabalho na PdC.

4.3.3 Interatividade com Objetos

Após a construção do painel referido na Secção 4.3.1, o passo seguinte seria definir o comportamento dos vários botões de opção presentes no mesmo. Para isso foi criado o componente **select-on-click** que adiciona um *event listener* (observador de eventos) às entidades correspondentes.

Quando o evento, neste caso a seleção, é ativado, o componente verifica primeiramente se o painel ainda está ativo e, caso esteja, de que tipo de questão se trata (singular ou múltipla). De acordo com o tipo de pergunta, a função deseleciona, automaticamente, as outras opções quando uma delas é escolhida (singular) ou adiciona a opção ao leque de respostas (múltipla). Este componente também é responsável por definir a cor da seleção como podemos observar pela Figura 4.6

Em trabalho futuro, a cor ou comportamento poderiam ser definidos externamente, num ficheiro de configuração. Nesse cenário, também está aberta a possibilidade de adicionar e/ou modificar valores para cada uma das opções se tal for pretendido.

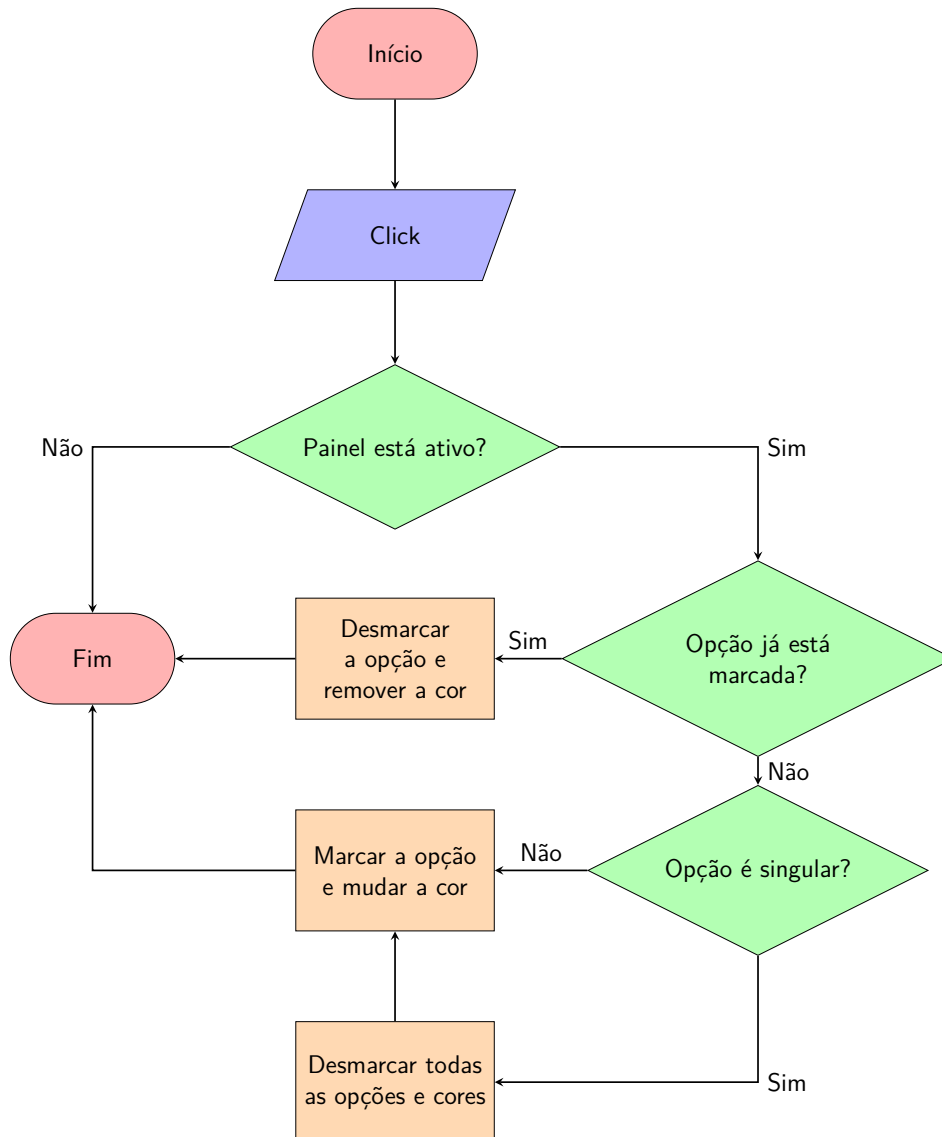


Figura 4.6: Fluxograma da função *select-on-click*

A próxima função criada, **end-panel**, tem como propósito validar e concluir um determinado painel na atividade. A função desabilita o seletor do painel de forma a que o utilizador não consiga voltar à mesma, muda a cor do seletor para negro (indicando visualmente que o painel está concluído), desabilita todos os botões de seleção de opção, incluindo o da submissão e, finalmente, gera um alerta em JS com as opções selecionadas.

A estrutura da função é bastante linear conforme a Figura 4.7. É de notar que as opções têm que ser desabilitadas individualmente pelo que é necessário um ciclo que desabilite cada opção.

4.3.4 Processamento das Respostas de um Painel

Durante o desenvolvimento da função **end-panel**, apresentada na secção anterior, foi decidido apresentar apenas as opções que tinham sido selecionadas ao responder à questão colocada no painel

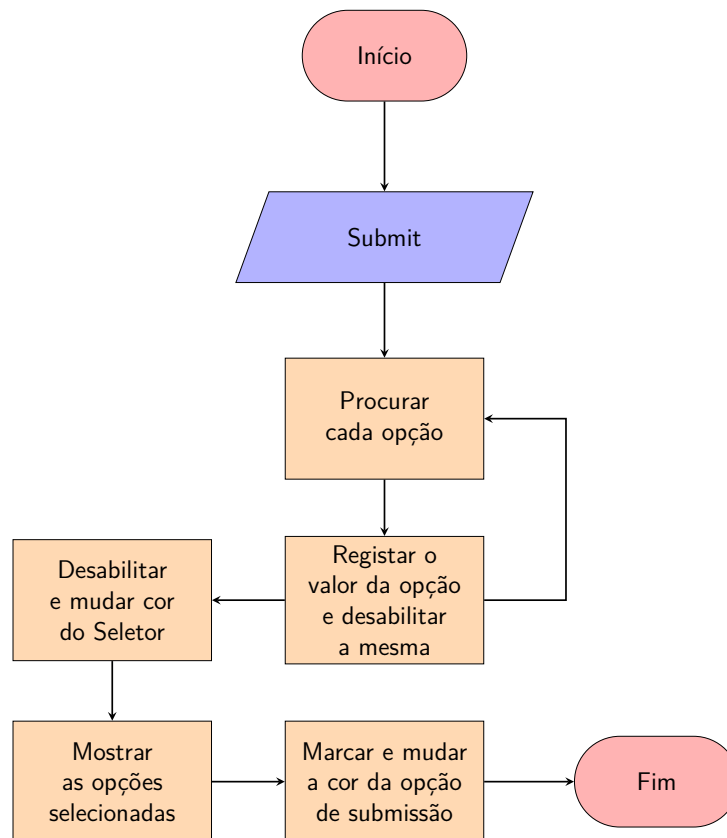


Figura 4.7: Fluxograma da função *end-panel*

ativo. Isto deveu-se, principalmente, para demonstrar que a PdC é capaz de coletar as repostas individuais de cada painel o que facilitará a integração futura, num sistema de e-aprendizagem. Neste momento, a função não tem memória de um painel para o seguinte mas isso seria resolúvel em trabalho futuro se se pretendesse devolver apenas o resultado final da atividade para o sistema, integrando um sistema de cotação, por exemplo.

Também seria possível fazer o contrário, i.e., devolver os resultados ao sistema de e-aprendizagem e deixar este calcular as cotações e o resultado final visto que a maioria destes sistemas já possuem módulos indicados para o efeito.

4.3.5 Navegação no Ambiente

A navegação no ambiente virtual sempre foi uma preocupação, desde o início deste trabalho. Apesar dos custos dos controladores de RV terem diminuído nos últimos anos, ainda colocam uma barreira à entrada a este tipo de experiências. Visto que a orientação primordial desta PdC é para a área de educação, foi importante desenhar o sistema de forma a que o mesmo fosse acessível e realizável com o menor número de recursos possível. Desta forma, optou-se por garantir a compatibilidade e funcionamento com dispositivos móveis com recurso a uma simples armação/encaixe do tipo GC.

A escolha deste tipo de dispositivo RV levantou, no entanto, imediatamente, várias questões, nomeadamente:

- providenciar uma forma de seleção de ação sem recurso a botões (o botão magnético do GC nem sempre funcionou no telemóvel do autor deste documento);
- permitir uma forma de exploração/navegação no ambiente virtual sem recorrer ao uso de controladores externos.

Após várias pesquisas, verificou-se que seria possível programar um cursor que acompanhasse a visão do utilizador e que poderia desencadear uma seleção caso o mesmo fixasse o olhar sobre um objeto durante um intervalo de tempo pré-definido. A problemática do botão magnético do GC poderia ser assim ultrapassada.

O segundo problema apontado, revelou-se um desafio. Colocaram-se duas hipóteses em cima da mesa. A primeira opção seria a de apenas utilizar um painel que, à medida que fosse completado pelo utilizador, fosse substituído, de forma linear, pela próxima questão. A grande desvantagem desta opção seria óbvia. Não faz sentido desenvolver um ambiente virtual quando o utilizador não pode navegar nele. Seria perdida a oportunidade de trazer uma mais-valia importante à produção de atividades de questões de resposta dicotómicas ou múltiplas em ambientes 3D.

A segunda opção consistiria em empregar um *seletor*, em forma de diamante, colocado no topo do painel, que permitisse deslocar o utilizador até a um ponto predefinido, focando simplesmente o olhar nesse objeto. Isto iria possibilitar uma navegação, de painel em painel e, conseqüentemente, permitir ao utilizador, explorar o ambiente ao redor. Esta solução também iria facilitar a navegação entre painéis para a frente e para trás.

A grande desvantagem desta opção é que seria necessário garantir que um utilizador tivesse sempre, no mínimo, dois painéis em linha de vista para que a navegação estivesse sempre garantida.

No fim, foi decidido utilizar múltiplos painéis, dispostos num padrão regular, pelo ambiente virtual conforme Figura 4.8 e que, de alguma forma, acabariam por transmitir uma sensação de exploração de um labirinto ao utilizador.



Figura 4.8: Posicionamento dos Painéis na Atividade

Para ativar a função de navegação do elemento seletor previamente desenvolvido, foi criada a função **checkpoint** que, no fundo, se refere ao posicionamento do utilizador frente ao painel ativo, conforme será abordado na secção seguinte.

4.3.6 Construção do DOM

Uma vez definido o painel, o posicionamento no ambiente virtual e a forma de navegação, desenvolveu-se um algoritmo que permite, de forma automática, construir e colocar os objetos, na atividade. Aqui surgiu mais um obstáculo, uma vez que não se pretendia uma colocação aleatória dos painéis no meio ambiente para não sofrer do risco de sobreposições, painéis inacessíveis ou questões fora da ordem inicialmente definida (apesar de permitir ao utilizador navegar pelos exercícios como melhor entendesse).

O posicionamento de qualquer objeto num espaço tridimensional implica definir, não apenas, coordenadas nas 3 dimensões x, y, z mas também a pose do mesmo. Acrescido a isto, o elemento seletor/localizador de cada painel também terá de posicionar o utilizador corretamente à frente de cada questão de maneira a que o mesmo possa selecionar as opções que julgar corretas.

Desta forma o problema foi, primeiramente, subdividido em três partes, i.e., foi decidido separar a pose do objeto, tal como o posicionamento do utilizador à frente de cada questão da localização de cada painel.

Para a pose, definiu-se um vetor com todas as rotações pretendidas e válidas na atividade, conforme a Listagem 4.6.

```
/*  
Define as poses para as primeiras 4 instâncias do painel:  
0 graus, 90 graus contra o sentido dos  
ponteiros do relógio, 0 graus e 90 graus no sentido dos ponteiros do relógio  
*/  
  
const face = ["0 0 0", "0 90 0", "0 0 0", "0 -90 0"];
```

Listagem 4.6: Definição da pose do painel para quatro instâncias

Este vetor permite, de uma forma simples, aplicar a pose ao objeto, dentro de um ciclo, em que cada instância é mapeada a uma pose conforme o resto da divisão do número do ciclo por 4.

Para o posicionamento espacial dos painéis, definiu-se uma matriz, como podemos observar na Listagem 4.7 e que define as posições das quatro instâncias iniciais do objeto.

Também foi definida uma matriz para o posicionamento do utilizador, quando este utilizasse o seletor de cada questão, como está representado na Listagem 4.8.

Por último, combinaram-se ambas as matrizes e o vetor de forma a que cada painel, independentemente do número de questões configuradas, fosse posicionado com exatidão no ambiente virtual de forma a que o utilizador pudesse navegar "à vista" de painel em painel.

```

/* Código JS
Define as posições para as primeiras 4 instâncias do painel */

const place = [
  [0, 2.5, -4],
  [-10, 2.5, 0],
  [-6, 2.5, -14],
  [4, 2.5, -10],
];

```

Listagem 4.7: Definição do posicionamento do painel para quatro instâncias

```

/* Código JS
Define as posições do utilizador para as primeiras 4 instâncias do painel */

const cam = [
  [0, 1.6, 0],
  [-6, 1.6, 0],
  [-6, 1.6, -10],
  [0, 1.6, -10],
];

```

Listagem 4.8: Definição do posicionamento do utilizador para quatro instâncias do painel

A partir deste momento, o restante trabalho consistiu em automatizar a geração do ambiente virtual a partir de código JS, através da função **build-dom**, invocada logo no início da estrutura *A-Frame* e cujo fluxograma 4.9 demonstra a lógica de construção.

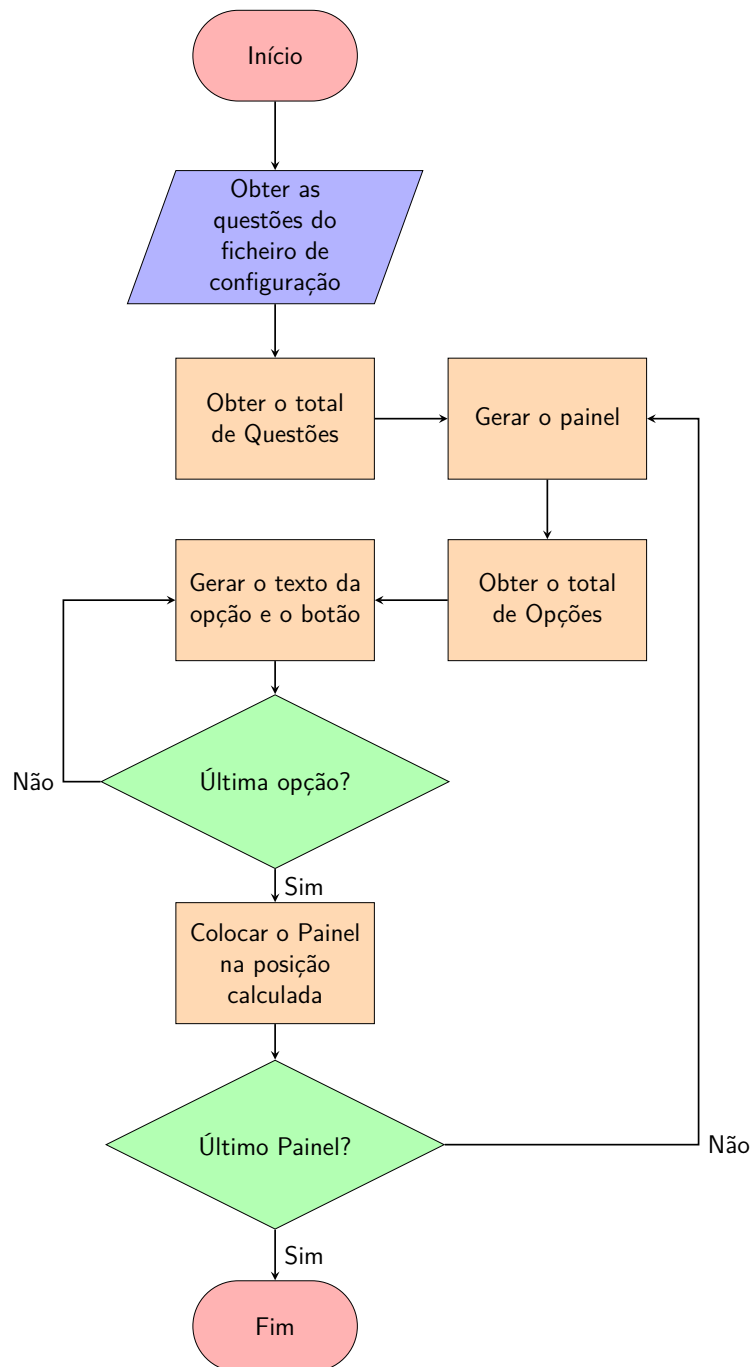
4.3.7 Especificação e Desenvolvimento da configuração YAML

Como já tinha sido definido, uma das premissas iniciais deste projeto consistiu no facto de que um utilizador (do tipo docente) pudesse, de uma forma simples e fácil, definir um conjunto de questões de maneira a gerar um ambiente virtual personalizado e dinâmico. Durante as pesquisas nessa área, surgiram várias opções possíveis, entre as quais a utilização de linguagens como o *eXtensible Markup Language* (XML), *JavaScript Object Notation* (JSON) e YAML.

A utilização da linguagem XML foi descartada à priori uma vez que a serialização dos sub elementos iria requerer a utilização de esquemas adicionais de validação (XSD)¹ e desta forma tornar o ficheiro de configuração mais complexo e, conseqüentemente, aumentar a probabilidade de ocorrerem erros durante a especificação da configuração. O formato JSON é aquele que melhor suporte nativo tem para a linguagem JS mas não está tão próximo da linguagem natural como é o caso do YAML.

A linguagem YAML caracteriza-se por utilizar a indentação para categorizar e agrupar elementos, atributos e valores, facto que, do ponto de vista do utilizador, a torna mais intuitiva na hora de

¹XML Schema Definition

Figura 4.9: Fluxograma da função *build-dom*

construir um ficheiro de configuração. A grande desvantagem, neste momento, são a falta de suporte nativo na linguagem JS o que implica a utilização de uma biblioteca externa. Foram feitas diversas pesquisas sendo que a opção recaiu sobre uma bifurcação da biblioteca *js-yaml* desenvolvida por Jérémy Faivre¹ [Fai], que atualmente, suporta a especificação mais recente do YAML 1.2 e conta com mais de vinte contribuidores.

A especificação do ficheiro de configuração YAML foi elaborada de acordo com a Listagem 4.9.

¹<https://github.com/jeremyfa/yaml.js>

```
questions:
- question: "Cores RGB"
  questiontext: "Que cores compõem o RGB?"
  type: choice
  singular: True
  options:
  - optiontext: "Verde, Amarelo e Azul"
  - optiontext: "Vermelho, Verde e Azul"
  - optiontext: "Castanho, Amarelo e Vermelho"
  - optiontext: "Nenhuma das anteriores"
```

Listagem 4.9: Especificação do ficheiro de configuração YAML

Como podemos observar, a configuração da pergunta é bastante linear. Definimos um nome curto para a questão, atribuímos um texto, especificamos um tipo (neste momento apenas poderemos definir questões do tipo *choice*), indicamos à atividade se a questão é do tipo dicotómica, ou não, e, finalmente, incluímos uma lista de opções a serem apresentadas. A estrutura tem uma apresentação limpa e fácil de entender e repetível pelo utilizador responsável pela sua edição.

5

Resultados Obtidos

A PdC, na sua versão final, foi disponibilizada em <https://feualg.pt/others/demo>. O código fonte pode ser encontrado em <https://gitlab.com/tchavei/demo-vr>.

Este capítulo divide-se em duas partes: os resultados obtidos através da PdC e os resultados globais deste trabalho.

5.1 Análise aos Resultados da PdC

No fim do capítulo 3 foram definidos quatro objetivos para a PdC que importa relembrar:

1. criar um ambiente de RV imersivo, interativo, adaptável e educativo que possa ser explorado com recursos simples e de baixo custo, como por exemplo, através do uso de GC;

2. o ambiente definido no ponto anterior tem que permitir que um utilizador (do tipo docente) possa, de uma forma simples, fácil e sem necessidade de dominar uma linguagem de programação específica, definir uma atividade composta por uma ou mais perguntas, de resposta única ou múltipla, constituída por uma ou mais opções de resposta e cujos resultados sejam passíveis de serem registados;
3. um utilizador (do tipo discente) deve poder interagir com o ambiente preparado anteriormente, responder a uma questão através da seleção de uma ou mais opções, marcar a questão como concluída (ou não) e mover-se para a próxima questão sem necessidade de meios auxiliares de controlo para além da visão até que toda a atividade seja completada.
4. desenvolver várias bibliotecas em H5P de forma a integrar a PdC, num sistema de gestão de aprendizagem, como o Moodle.

O primeiro ponto, foi, na minha opinião, totalmente cumprido. A utilização da arquitetura *A-Frame* permitiu desenvolver um ambiente RV interativo e adaptável incluindo a possibilidade de exploração do ambiente apenas com recurso a GC e um *smartphone* que, hoje em dia, está ao alcance de praticamente todo o mundo.

O segundo objetivo também me parece que foi plenamente atingido uma vez que, tal como tinha sido proposto, toda a atividade pode ser definida através de um único ficheiro de configuração, estandardizado, na linguagem YAML. Nesse ficheiro, foram definidos todos os campos necessários à caracterização de cada questão e as suas respetivas opções. Ao utilizar a linguagem YAML, que é muito próxima da linguagem natural, é dada a possibilidade ao tutor/docente definir a atividade sem necessitar de conhecimentos específicos na área da programação. A atividade é facilmente adaptável a qualquer tipo de matéria que possa ser avaliada por meio de questões de texto com opções de resposta do tipo dicotómico ou múltiplo. A PdC mostra, ao concluir cada painel, a(s) opção(ões) selecionada(s) através de um alerta gerado via JS o que demonstra que as repostas ficam disponíveis no ambiente de desenvolvimento e que poderão ser enviadas para qualquer sistema subjacente, no futuro.

O terceiro objetivo, referente à interatividade, também foi cumprido. Para satisfazer todos os critérios definidos, foi encontrada uma solução de navegação, via foco por parte do utilizador. Foi desenvolvido um sistema de navegação por meio de objetos, do tipo *diamante*, que agem como *checkpoints* ou localizadores. Quando estes objetos são ativados, via foco, o utilizador é transportado no espaço de RV até ao painel associado a esse localizador. Desta forma, é dada liberdade de movimento sem a necessidade de empregar um controlador adicional. Aliás, toda a interatividade com os painéis e as respetivas questões e opções são ativadas/desativadas através do foco por parte do utilizador, permitindo assim cumprir também uma das premissas definidas no primeiro objetivo (custos controlados).

Por fim, em relação ao quarto objetivo, diria que foi parcialmente atingido. Foi despendido bastante tempo e trabalho na área do H5P, resolveram-se inúmeras questões relacionadas com a própria construção, trocaram-se opiniões junto da comunidade [AFb] e foi possível construir um *wrapper* em JS para disponibilizar a biblioteca *A-Frame* em formato H5P. No entanto, não foi possível desenvolver e incluir, em tempo útil, a componente *H5P* na PdC. Algo que possivelmente ficará para desenvolvimento futuro na área.

5.2 Análise aos Resultados Globais

Considero que, na globalidade, o resultado deste trabalho foi muito positivo. Os objetivos inicialmente delineados foram claros, explícitos mas também honestos. Não se conheciam os resultados de antemão e o caminho trilhado revelou-se incerto a cada passo. Uma verdadeira aventura de exploração científica, quer durante a revisão ao estado da arte, quer durante o desenvolvimento da PdC. Praticamente todos os objetivos que foram propostos, foram cumpridos. No entanto, no decurso do desenvolvimento desta dissertação, apercebi-me da imensidão que esta área, em particular, tem para oferecer. Sinceramente, considero que o desenvolvimento de ferramentas de RV para o ensino poderia, facilmente, ser encarada como o trabalho de uma vida. Há sempre mais por explorar, mais para integrar, mais para desenvolver. Neste contexto, creio que esta tese tenha dado um contributo para que arquiteturas como o *A-Frame*, em conjugação com uma visão mais abrangente sobre como facilitar a personalização de ambientes de RV para o utilizador comum, se possam tornar *mainstream* no futuro.

6

Conclusão e Trabalho Futuro

Este trabalho demonstrou ser desafiante mas, ao mesmo tempo, enriquecedor. Não apenas pelas diversas áreas que tiveram de ser exploradas mas também pelo desenvolvimento prático de uma demonstração funcional. A conjugação de diversas tecnologias numa só aplicação com um propósito claro, apesar de simples, deixou uma marca de realização pessoal.

No entanto, seria demasiado pretensioso declarar o trabalho concluído e pronto para ser implementado. As áreas envolvidas são demasiado vastas, o tema está em constante evolução e existem muitos aspetos que poderão ser desenvolvidos, em mais detalhe, abrindo caminho para uma verdadeira integração funcional e humana entre as tecnologias de RV e os atuais sistemas de e-aprendizagem.

Ao longo do projeto foram identificados vários pontos, que por questões de abrangência, restrições temporais ou limitações pessoais, foram delegados para trabalho futuro e que aqui são enumerados.

6.1 Integração com a tecnologia H5P

Como já foi abordado, a tecnologia H5P é muito interessante, de código aberto e livre, e em franca expansão. Foi demonstrado que pode ser utilizada como veículo para distribuir soluções *A-Frame* dentro de sistemas de e-aprendizagem apesar de que, presentemente, com muitas limitações. A tecnologia ainda é relativamente recente e não existem bibliotecas nativas disponíveis para permitir o desenvolvimento direto de aplicações pedagógicas/educativas em formato *A-Frame*. Para isso acontecer, será necessário apostar no desenvolvimento de vários aspetos relacionados como a seguir enumerados.

6.1.1 Biblioteca nativa de *A-Frame* para H5P

Um dos grandes desafios neste trabalho foi o de tentar executar código de *A-Frame* dentro de um contentor H5P visto que este último não permite correr bibliotecas externas ao mesmo. Isto obrigou a que a biblioteca *A-Frame* tivesse que ser integrada através de artifícios, como já enunciado, via JS e que, pela natureza da experiência, torna a biblioteca estática e de difícil manutenção. Seria muito interessante, no futuro, talvez em conjunto com outros programadores do projeto H5P, desenvolver uma forma de permitir a ligação a bibliotecas externas ou, em alternativa, pensar na criação de um projeto de desenvolvimento e manutenção de uma versão H5P da biblioteca principal de *A-Frame*.

6.1.2 Adição de conectores *A-Frame* ou H5P

Presentemente, projetos desenvolvidos em *A-Frame* têm que ser exclusivos na página em que são executados. Não é possível fazer funcionar duas ou mais instâncias de ambientes RV na mesma página Web sem recurso a *iframes* embutidos o que levanta problemas de integração dentro de sistemas de e-aprendizagem como, por exemplo o Moodle. Outro aspecto relacionado é o facto do ambiente *A-Frame* correr em formato maximizado e, conseqüentemente, não devolver para o subsistema em que está inserido (neste caso um contentor H5P), o espaço que ocupa numa página Web ou ecrã. Como já foi explicado anteriormente, isto levanta, presentemente, grandes problemas para conseguir mostrar um exercício desenvolvido em *A-Frame* dentro de uma atividade H5P. Seria importante, em futuro trabalho, resolver esta questão para permitir, aliado aos anteriores trabalhos mencionados, uma fácil integração de exercícios RV em sistemas de e-aprendizagem.

6.1.3 Desenvolvimento de um editor H5P

Um dos pontos fortes da tecnologia H5P é que não apenas permite a fácil criação de atividades educativas para serem colocadas num sistema de e-aprendizagem mas também possibilita o desenvolvimento de editores para essas mesmas atividades. Dado que, no decurso deste trabalho, foi demonstrado o uso de um ficheiro de configuração do tipo YAML como fonte de geração de um ambiente RV, seria uma mais-valia, criar um editor para esta atividade via pacote H5P.

6.2 Refatoração do código *JavaScript* utilizado no PDC

O código incluído neste projeto foi desenvolvido de forma estruturada mas seguindo um estilo de programação procedimental, que se deveu, em pequena parte, às limitações temporais, mas muito também devido às limitações pessoais no que concerne à programação por objectos aplicada a uma linguagem de programação em que o autor ainda não é muito versado. Um ponto chave, em trabalho futuro, será o de refatorar o código atual para uma programação orientada a objetos, criando métodos e sobrecargas que permitam, de forma muito mais eficiente, gerar o DOM necessário à execução do ambiente.

6.3 Inclusão do cálculo de resultados

Um dos aspetos, que definitivamente será interessante em trabalhos futuros, será o desenvolvimento opcional de um sistema de pontuação dentro da própria atividade RV. Através da adição de valores iniciais e de seleção seria possível não só a inclusão de diversas técnicas e regras de pontuação empregue pelo avaliador, mas também simplificar os dados a enviar para o contentor H5P e consequentemente para o sistema de e-aprendizagem envolvente.

6.4 Desenvolvimento de um conector *xAPI*

Sendo o exercício *A-Frame* subjugado à atividade H5P, que já possui um conector *Experience API (xAPI)*¹ para transmitir resultados para um sistema de aprendizagem como o Moodle, é essencial que seja desenvolvida uma forma de transmitir os resultados obtidos da interação entre o utilizador e o ambiente RV para dentro da atividade H5P para que a mesma possa ser retransmitida para o sistema envolvente.

6.5 Expansão do tipo de atividades

No presente trabalho apenas foi abordada a problemática das questões de resposta dicotómica ou múltipla. Com a perspetiva de massificação do uso de tecnologias RV e 3D em contexto de aprendizagem, seria muito interessante expandir o atual trabalho para outro tipo de exercícios como, por exemplo, os de emparelhamento de conceitos, os de classificação por escala ou até os do tipo *Likert* [Rob14], amplamente utilizados hoje em dia, com recurso a tecnologias mais tradicionais.

6.6 Desenvolvimento de múltiplos modos de execução da atividade

Um outro aspecto que poderá ser abordado em trabalhos futuros será o de incluir mais do que um modo de execução da atividade de RV. Nesta dissertação, o foco principal foi demonstrar a

¹<https://xapi.com/>

possibilidade de desenvolver exercícios dinâmicos em ambiente RV em contexto de aprendizagem independentemente da orientação desses mesmos exercícios. Com algum desenvolvimento e expansão do código agregado, poderiam implementar-se múltiplos modos de funcionamento como por exemplo modos de *treino* e modos de *avaliação*. Estes modos não deverão ser confundidos com os modos que alguns sistemas de e-aprendizagem já disponibilizam, mas sim modos que poderiam detetar as lacunas na aprendizagem de um indivíduo e ajustar o tipo de questões a apresentar de forma a treinar o utilizador nas áreas onde demonstre maiores debilidades.

Paralelamente, também poderia ser implementada uma codificação da tipologia das perguntas e um agrupamento por temas relacionados. Isto serviria não só como um alicerce de apoio para a criação dos modos de funcionamento descritos mas também facilitar a geração automática de perguntas se for esse um dos caminhos por onde enveredar.

6.7 Diversificação do Ambiente RV

Como já enunciado durante o desenvolvimento do PdC, recorreu-se a uma biblioteca externa para gerar o ambiente de RV circundante. Esse ambiente é, neste momento estático, ainda que possa ser facilmente modificado através de pequenas alterações no código fonte. Um futuro trabalho poderá incluir várias opções pré-definidas para a escolha do utilizador ou inclusive poderão ser desenvolvidos ambientes personalizados de forma a serem incluídos numa biblioteca própria.

6.8 Biblioteca YAML

A demonstração prática nesta dissertação envolveu a leitura, em tempo real, de um ficheiro de configuração YAML conforme já foi apresentado. Essa leitura foi possível devido à inclusão de mais uma biblioteca externa (*yaml.js*) desenvolvida por [Fai], uma bifurcação da biblioteca *js-yaml* que faz parte do pacote *Node.js*.

O problema atual é que os autores do *Node.js*, atualmente, o pacote de distribuição mais popular dos binários da linguagem JS para ambientes não Web, afirmam claramente que a biblioteca *js-yaml* que faz parte do pacote, não é dirigida para exploradores Web pelo que o suporte será sempre limitado e residual [Cha]. Seria importante e útil, criar uma bifurcação do *js-yaml* dedicada e simplificada para uso em outros projetos de desenvolvimento RV com base em ficheiros de configuração dinâmicos como é o caso neste trabalho.

Bibliografia

- [AFa] A-FRAME: *Best Practices – A-Frame*. <https://aframe.io/docs/1.2.0/introduction/best-practices.html>. – último acesso a 11-11-2021
- [AFb] A-FRAME: *Call A-frame library from H5P | H5P*. <https://h5p.org/node/1085512>. – último acesso a 12-11-2021
- [AFc] A-FRAME: *Introduction – A-Frame*. <https://aframe.io/docs/master/introduction/>. – último acesso a 01-11-2021
- [AF20] AMBROSIO, Adriana P. ; FIDALGO, M. Isabel R.: Past, present and future of Virtual Reality: Analysis of its technological variables and definitions. In: *Culture & History Digital Journal* 9 (2020), 6, e010-e010. <http://dx.doi.org/10.3989/CHDJ.2020.010>. – DOI 10.3989/CHDJ.2020.010. – ISSN 2253–797X
- [AML16] ALAMAN, Xavier ; MATEU, Juan ; LASALA, Maria J.: Designing virtual world educational applications, IEEE Computer Society, 5 2016. – ISBN 9781467386333, S. 1134–1137
- [Bla] BLACKBOARD: *What is Blackboard*. <http://www.blackboard.com/>. – último acesso a 03-03-2021
- [BNO⁺19] BOUALI, Nacir ; NYGREN, Eeva ; OYELERE, Solomon S. ; SUHONEN, Jarkko ; CAVALLI-SFORZA, Violetta: Imikode: A VR Game to Introduce OOP Concepts. (2019). <http://dx.doi.org/10.1145/3364510.3366149>. – DOI 10.1145/3364510.3366149. ISBN 9781450377157
- [Cha] CHAPDELAINÉ, Julie: *js-yaml - GitLab*. https://gitlab.istic.univ-rennes1.fr/jchapdelaine/jxc_tp_vuejs/-/tree/master/node_modules/js-yaml. – último acesso a 12-11-2021
- [Cla] CLAROLINE: *Claroline*. <https://www.claroline.com/>. – último acesso a 03-05-2021

- [Com20] COMMONS, Wikimedia: *File:6DOF.svg* — *Wikimedia Commons, the free media repository*. <https://commons.wikimedia.org/w/index.php?title=File:6DOF.svg&oldid=482826102>. Version: 2020. – último acesso a 12-11-2021
- [Fai] FAIVRE, Jérémy: *GitHub - jeremyfa/yaml.js: Standalone JavaScript YAML 1.2 Parser & Encoder*. <https://github.com/jeremyfa/yaml.js>. – último acesso a 03-11-2021
- [FLZH09] FRANCESCHI, Katherine ; LEE, Ronald ; ZANAKIS, Stelios ; HINDS, David: Engaging group E-Learning in virtual Worlds. In: *Journal of Management Information Systems* 26 (2009), 7, 73-100. <http://dx.doi.org/10.2753/MIS0742-1222260104>. – DOI 10.2753/MIS0742-1222260104. – ISSN 07421222
- [GBJK19] GARCIA-BONETE, Maria J. ; JENSEN, Maja ; KATONA, Gergely: A practical guide to developing virtual and augmented reality exercises for teaching structural biology. In: *Biochemistry and Molecular Biology Education* 47 (2019), 1, 16-24. <http://dx.doi.org/10.1002/BMB.21188>. – DOI 10.1002/BMB.21188. – ISSN 1539-3429
- [HAAA⁺19] HATCHARD, Timothy ; AZMAT, Freeha ; AL-AMIN, Mohammad ; RIHAWI, Zeina ; AHMED, Bilal ; ALSEBAE, Alaa: Examining student response to virtual reality in education and training, Institute of Electrical and Electronics Engineers Inc., 7 2019. – ISBN 9781728129273, S. 1145-1149
- [HKS20] HUDÁK, Marián ; KOREČKO Štefan ; SOBOTA, Branislav: *Advanced User Interaction for Web-based Collaborative Virtual Reality*, IEEE, 2020
- [Mag21] MAGRO, Juliana: H5P. In: *Journal of the Medical Library Association* 109 (2021), 7. <http://dx.doi.org/10.5195/JMLA.2021.1204>. – DOI 10.5195/JMLA.2021.1204. – ISSN 1558-9439
- [Moo] MOODLE: *Moodle - Open-source learning platform*. <https://moodle.org/>. – último acesso a 06-10-2021
- [Moz] MOZILLA: *Internet for people, not profit* — *Mozilla*. <https://www.mozilla.org/en-US/>. – último acesso a 06-11-2021
- [MSP19] MOROZOV, Viktor ; SHELEST, Tetiana ; PROSKURIN, Maksym: Create the model for development of virtual reality E-learning, Institute of Electrical and Electronics Engineers Inc., 7 2019. – ISBN 9781728138824, S. 1265-1270
- [NAOM20] NESENBERGS, Krisjanis ; ABOLINS, Valters ; ORMANIS, Juris ; MEDNIS, Artis: Use of Augmented and Virtual Reality in Remote Higher Education: A Systematic Umbrella Review. In: *Education Sciences 2021, Vol. 11, Page 8 11* (2020), 12, 8. <http://dx.doi.org/10.3390/EDUCSCI11010008>. – DOI 10.3390/EDUCSCI11010008. – ISSN 22277102
- [NMI⁺20] NA, Kew S. ; MOHAMED, Farhan ; ISHAM, Muhammad Ismail M. ; SIANG, Chan V. ; TASIR, Zaidatun ; ABAS, Mohd A.: *Virtual Reality Application Integrated with Learning Analytics for Enhancing English Pronunciation: A Conceptual Framework*, Institute of Electrical and Electronics Engineers Inc., 11 2020. – ISBN 9781728192420, S. 82-87

- [Oke21] OKELY, John: *GitHub - Quiz Game Moodle activity module to create dynamic games based on the question bank*. https://github.com/xow/moodle-mod_quizgame. Version: 2021. – último acesso a 20-11-2021
- [OMH] OKELY, John ; MOODLE-HQ: *Virtual Reality in Moodle*. https://www.youtube.com/watch?v=cnCqNpWarFI&t=896s&ab_channel=Moodle
- [PD21] PIRKER, Johanna ; DENGEL, Andreas: The Potential of 360° Virtual Reality Videos and Real VR for Education - A Literature Review. In: *IEEE Computer Graphics and Applications* 41 (2021), 7, S. 76–89. <http://dx.doi.org/10.1109/MCG.2021.3067999>. – DOI 10.1109/MCG.2021.3067999. – ISSN 15581756
- [PP15] PERSICO, Donatella ; POZZI, Francesca: Informing learning design with learning analytics to improve teacher inquiry. In: *British Journal of Educational Technology* 46 (2015), S. 230–248. <http://dx.doi.org/10.1111/bjet.12207>. – DOI 10.1111/bjet.12207. – ISBN 1467–8535
- [PSM21] PATHAK, Rishabh ; SIMISCUKA, Anderson A. ; MUNTEAN, Gabriel-Miro: An Adaptive Resolution Scheme for Performance Enhancement of a Web-based Multi-User VR Application, IEEE, 2021
- [RM12] RAGBIR, Diana ; MOHAN, Permanand: Using social networking and virtual reality to harness creativity in students in e-learning situations, 2012. – ISBN 9780769547022, S. 734–735
- [Rob14] ROBINSON, John ; MICHALOS, Alex C. (Hrsg.): *Likert Scale*. Dordrecht : Springer Netherlands, 2014. – 3620–3621 S. http://dx.doi.org/10.1007/978-94-007-0753-5_1654. http://dx.doi.org/10.1007/978-94-007-0753-5_1654. – ISBN 978–94–007–0753–5
- [SC19] SANTOS, Solange G. ; CARDOSO, Jorge C. S.: Web-based Virtual Reality with A-Frame, IEEE, 2019
- [SGM+19] STEFFEN, Jacob H. ; GASKIN, James E. ; MESERVY, Thomas O. ; JENKINS, Jeffrey L. ; WOLMAN, Iopa: Framework of Affordances for Virtual Reality and Augmented Reality. In: *Journal of Management Information Systems* 36 (2019), 7, 683–729. <http://dx.doi.org/10.1080/07421222.2019.1628877>. – DOI 10.1080/07421222.2019.1628877. – ISSN 1557928X
- [SMV20] SHENOY, Veena ; MAHENDRA, Sheetal ; VIJAY, Navita: COVID 19 – Lockdown: Technology Adaption, Teaching, Learning, Students Engagement and Faculty Experience. In: *Mukt Shabd Journal* IX (2020), 4
- [SR17] SUN, Kwangwon ; RYU, Sukyoung: Analysis of JavaScript programs: Challenges and research trends. In: *ACM Computing Surveys* 50 (2017), 8. <http://dx.doi.org/10.1145/3106741>. – DOI 10.1145/3106741. – ISSN 15577341
- [SSK20] SHAKIROVA, Nurzhanat D. ; SAID, Nidal A. ; KONYUSHENKO, Svetlana M.: The Use of Virtual Reality in Geo-Education. In: *International Journal of Emerging Technologies in Learning (iJET)* 15 (2020), 10, S. 59. <http://dx.doi.org/10.3991/ijet.v15i20.15433>. – DOI 10.3991/ijet.v15i20.15433

- [Sup] SUPERMEDIUM: *Supermedium | The VR Comic Book Reader*. <https://supermedium.com/>. – último acesso a 06-11-2021
- [Sur21a] SURVEYS, W3Techs World Wide Web T.: *Usage Statistics and Market Share of Server-side Programming Languages for Websites, November 2021*. https://w3techs.com/technologies/overview/programming_language. Version: 2021. – último acesso a 21-11-2021
- [Sur21b] SURVEYS, W3Techs World Wide Web T.: *Usage Statistics of JavaScript as Client-side Programming Language on Websites, November 2021*. <https://w3techs.com/technologies/details/cp-javascript>. Version: 2021. – último acesso a 21-11-2021
- [Thr] THREE.JS: *Three.js – JavaScript 3D Library*. <https://threejs.org/>. – último acesso a 06-11-2021
- [YGG20] YÉPEZ, Jonathan ; GUEVARA, Luis ; GUERRERO, Graciela: *AulaVR: Virtual Reality, a telepresence technique applied to distance education.*, IEEE, 2020



UNIVERSIDADE DE ÉVORA
INSTITUTO DE INVESTIGAÇÃO
E FORMAÇÃO AVANÇADA

Contactos:
Universidade de Évora
Instituto de Investigação e Formação Avançada — IIFA
Palácio do Vimioso | Largo Marquês de Marialva, Apart. 94
7002 - 554 Évora | Portugal
Tel: (+351) 266 706 581
Fax: (+351) 266 744 677
email: iifa@uevora.pt