

FACULDADE DE ENGENHARIA DA UNIVERSIDADE DO PORTO

Ambientes Urbanos Virtuais para a Web

Joana Filipa Vieira Barbosa



Mestrado Integrado em Engenharia Informática e Computação

Orientador: António Fernando Coelho (Prof. Auxiliar)

Julho de 2013

© Joana Filipa Vieira Barbosa, 2013

Ambientes Urbanos Virtuais para a Web

Joana Filipa Vieira Barbosa

Mestrado Integrado em Engenharia Informática e Computação

Aprovado em provas públicas pelo Júri:

Presidente: Rosaldo José Fernandes Rossetti (Prof. Auxiliar)

Vogal Externo: Daniel Castro Silva (Prof. Auxiliar)

Orientador: António Fernando Coelho (Prof. Auxiliar)

18 de Julho de 2013

Resumo

Desde os anos 60 que a computação gráfica tem vindo a evoluir e a obter cada vez mais relevância na maioria das aplicações informáticas. Em especial os gráficos 3D e os ambientes virtuais encontram-se cada vez mais presentes nos produtos atuais, seja na indústria ou no entretenimento, tal como nos jogos ou no cinema.

Contudo, apesar da grande evolução das tecnologias Web, existe ainda uma grande lacuna no que toca a este tipo de gráficos, tridimensionais e interativos, disponibilizados na World Wide Web.

Esta dissertação tem como objetivo explorar o desenvolvimento de uma aplicação interativa que disponibilize ambientes 3D de espaços urbanos que funcionem de uma forma nativa no *browser*. Este tipo de aplicação deverá ter em conta conceitos de interação pessoa-computador baseando-se no modo como o utilizador adquire informação sobre espaços através de conteúdos tridimensionais para oferecer uma interface usável e apelativa.

Este sistema surge no contexto do produto aLIVE Places, a ser presentemente desenvolvido pela empresa 3Decide. No âmbito do projeto foi prototipado um módulo direcionado à divulgação de eventos, possibilitando a qualquer responsável pela organização de um evento utilizar o modelo tridimensional interativo como uma espécie de convite, atribuindo-lhe todas as informações associadas ao mesmo.

Através da oferta uma interação simples e clara e não requerendo a execução de qualquer tipo de ação extra para aceder à aplicação, como a instalação de plug-ins, é possível alcançar um maior número de utilizadores pela facilidade de acesso e utilização da mesma.

Abstract

Since the 60's, computer graphics has been evolving and obtaining more and more relevance for most part of computer applications. Especially 3D graphics and virtual environments, which find themselves even more present in modern products, whether in the engineering industry or the entertainment business, such as video games or cinema.

However, despite the Web technologies' great evolution, there is still a wide gap when it comes to three-dimensional and interactive graphics available on the Web.

The goal of this dissertation is to explore the development of an interactive application that provides 3D urban environments made available natively on the browser. This kind of application must rely on human-computer interaction concepts, considering the way a user acquires data on spaces through tridimensional contents to offer a pleasant and appealing interface.

This system appears in the context of aLIVE Places, a product in current development by 3Decide company. This project will be also composed of a module directed to event promotion, allowing to any event organizer to use a tridimensional interactive model like a kind of invitation, attaching to it all the useful associated data.

Providing a simple and clear interaction that doesn't require executing any kind of extra operation to access the application, like installing a plugin, it's possible to reach out to a larger number of users for its simplistic manner of use and access.

Agradecimentos

Obrigada aos meus Pais por me terem possibilitado realizar este longo percurso.

Obrigada ao Prof. António Coelho pela orientação dada ao longo deste processo.

Os meus agradecimentos a toda a equipa da 3Decide, pelo auxílio, colaboração e distintas opiniões em todas as fases do meu progresso.

E obrigada ao Tiago pela paciência incansável.

Joana Barbosa

Conteúdo

Introdução.....	1
1.1	Enquadramento e Motivação..... 1
1.2	Descrição do Problema..... 2
1.3	Objetivos 3
1.4	Resultados Esperados 4
1.5	Trabalho Relacionado 4
1.6	Metodologia 5
1.7	Estrutura do Documento..... 5
Revisão Bibliográfica	7
2.1	Representação de Ambientes Urbanos Virtuais em 3D 7
2.2	Interação em ambientes urbanos 10
2.3	Divulgação de espaços de eventos 11
2.4	Tecnologias Web3D..... 14
2.4.1	VRML 15
2.4.2	X3D 15
2.4.3	Java3D e JOGL 16
2.4.4	Flash 17
2.4.5	Silverlight 5 17
2.4.6	WebGL..... 17
2.4.7	Outras Tecnologias..... 19
2.5	Síntese / Escolha de Tecnologia 20
2.6	Resumo e Conclusões..... 21
Ambientes Urbanos Virtuais em WebGL	23
3.1	aLIVE Places..... 23
3.2	Descrição da Solução 24
3.3	Requisitos Funcionais e Casos de Utilização 25
3.3.1	Atores 25
3.3.2	Componente da Cena 3D..... 25
3.3.3	Componente de Informação de Espaços 27

3.3.4	Componente de Gestão Eventos.....	28
3.4	Arquitetura da Solução.....	31
3.5	Descrição da Interface.....	32
3.6	Resumo e Conclusões.....	33
Implementação		35
4.1	Ambiente 3D e Módulo de Informação de Espaços.....	35
4.1.1	WebGL e Three.js	35
4.1.2	Criação do elemento <i>Canvas</i> do HTML5.....	37
4.1.2.1	Visualização da cena	37
4.1.2.2	Interação com a cena	41
4.1.3	Criação da Página Web	43
4.1.3.1	Estruturação	43
4.1.3.2	Implementação	45
4.1.4	Interação com espaços.....	47
4.2	API - Interação entre cena e conteúdos	48
4.3	Módulo de Gestão de Eventos.....	49
4.3.1	Estruturação.....	49
4.3.1.1	Dados	49
4.3.1.2	Visual	51
4.3.2	Painéis informativos de Eventos	52
4.3.2.1	Calendário	52
4.3.2.2	Listagem de Dados	53
4.3.2.3	<i>Slider</i> de Detalhes	54
4.3.3	Percursos	54
4.3.3.1	Visualização	54
4.3.3.2	Animação	55
4.4	Resumo e Conclusões.....	56
Análise e Avaliação de Resultados		59
5.1	Interfaces	59
5.2	Testes.....	61
5.2.1	Testes de compatibilidade	62
5.2.2	Testes de Usabilidade.....	63
5.2.2.1	Descrição dos procedimentos	63
5.2.2.2	Análise e avaliação das Observações	64
5.3	Resumo e Conclusões.....	71
Conclusões e Trabalho Futuro		73
6.1	Trabalho Futuro.....	74
Referências.....		75

Anexo A – Testes de Usabilidade	79
A.1 – Tarefa nº1: Lista de Passos e Resultados	79
A.2 – Tarefa nº2: Lista de Passos e Resultados	80
A.3 – Tarefa nº3: Lista de Passos e Resultados	81
A.4 – Tarefa nº4: Lista de Passos e Resultados	82
A.5 – Tarefa nº5: Lista de Passos e Resultados	83
A.6 – Tarefa nº6: Lista de Passos e Resultados	84
A.7 – Tarefa nº7: Lista de Passos e Resultados	85
Anexo B – Questionários.....	86
B.0.1 – Enunciado do questionário realizado: Parte I.....	86
B.0.2 - Enunciado do questionário realizado: Parte II	87
B.0.3 – Enunciado do questionário realizado: Parte III	88
B.1.1 – Parte I: Quantos pisos tinha o edifício apresentado?	89
B.1.1 – Parte I: Qual era a forma do edifício apresentado?	90
B.1.3 – Parte I: Em que piso se encontrava o auditório?	91
B.1.4 – Parte I: Em que lado do edifício se encontrava o auditório?	92
B.1.5 – Parte I: Em que piso se encontravam as salas da empresa Tecla Colorida?	93
B.1.6 – Parte I: Quantos eventos decorriam no mês de Junho?	94
B.2 – Parte II: Numa escala de 1 a 6, como avalia os seguintes pontos:.....	95
B.3.1 – Parte III: Acha que uma aplicação deste género tem utilidade?	97
B.3.2 – Parte III: Qual seria a sua avaliação se o seu espaço de eventos favorito (Casa da Música, Alfândega do Porto, Coliseu do Porto, etc) passasse a utilizar esta aplicação para divulgar as suas atividades e os seus espaços?.....	98
B.3.3 – Parte III: Tem algum comentário ou sugestão que gostaria de partilhar sobre esta aplicação?.....	99

Lista de Figuras

Figura 1 - Exemplo de cenário da plataforma Second Life	8
Figura 2 - Exemplo de um ambiente virtual aplicado a um centro comercial [14]	8
Figura 3 - Sistema de tour virtual na Universidade de Agricultura de Shenyang	9
Figura 4 - Ambiente virtual do projeto de reconstrução da mesquita Alaca Imaret [10]	9
Figura 5 - Aplicação interativa do London Convention Center	12
Figura 6 - Aplicação interativa do Vancouver Convention Center	13
Figura 7 - Aplicação interativa do Walter E. Washington Convention Center – Vista de piso	13
Figura 8 - Aplicação interativa do Walter E. Washington Convention Center – Vista de sala	14
Figura 9 - Interface do produto aLIVE Places	24
Figura 10 - Diagrama de casos de uso: Cena 3D - Manipulação	26
Figura 11 - Diagrama de casos de uso: Cena 3D - Navegação	27
Figura 12 - Diagrama de casos de uso: Informação de Espaços	28
Figura 13 - Diagrama de casos de uso: Gestão de Eventos - Eventos	29
Figura 14 - Diagrama de casos de uso: Gestão de Eventos - Atividades	30
Figura 15 - Diagrama de casos de uso: Gestão de Eventos - Percursos	31
Figura 16 - Diagrama de Arquitetura do Sistema	32
Figura 17 - Mockup da interface da página	33
Figura 18 - Sistema de coordenadas seguido	37
Figura 19 - Representação de uma vista ortogonal à esquerda, e em perspectiva à direita	38
Figura 20 - <i>Mockup</i> da página web em relação ao módulo de informação de espaços	45
Figura 21 - Diagrama de base de dados relativo ao módulo de gestão de eventos	50
Figura 22 - Mockup da página web em relação ao módulo de gestão de eventos	52
Figura 23 - Máquina de estados da animação do percurso	55
Figura 24 - Interface final relativa à cena 3D conjuntamente com a informação de espaços	59
Figura 25 - Interface final relativa à cena 3D conjuntamente com a informação de eventos e atividades	60

Figura 26 - Interface final relativa à cena 3D conjuntamente com percursos da atividade selecionada	61
Figura 27 - Gráfico de resultados da questão 'Quantos pisos tinha o edifício apresentado?'	69
Figura 28 - Gráfico de resultados da questão 'Quantos eventos decorriam no mês de Junho?'	70
Figura 29 - Gráfico de resultados da questão 'Como avalia: o acesso a este tipo de aplicação diretamente no <i>browser</i> ?'	70
Figura 30 - Gráfico de resultados da questão 'Como avalia: A navegação no painel de eventos? '	71

Lista de Tabelas

Tabela 1 - Comparação das tecnologias 3D para a Web	20
Tabela 2 - Componentes da biblioteca Three.js	36
Tabela 3 - Tarefas realizadas pelos utilizadores no teste de usabilidade	64
Tabela 4 - Percurso mais eficaz para realizar a tarefa nº2	64
Tabela 5 - Registo do teste de usabilidade à tarefa nº 2	65
Tabela 6 - Percurso mais eficaz para realizar a tarefa nº3	66
Tabela 7 - Registo do teste de usabilidade à tarefa nº 3	66
Tabela 8 - Percurso mais eficaz para realizar a tarefa nº6	67
Tabela 9 - Registo do teste de usabilidade à tarefa nº6	68

Abreviaturas e Símbolos

2D	Bidimensional
3D	Tridimensional
3DMLW	3D Markup Language for Web
AMD	Advanced Micro Devices
API	Application programming interface
AWT	Abstract Windows Toolkit
BD	Blu-Ray Disk
CD	Compact Disk
CSG	Constructive Solid Geometry
COLLADA	COLLABorative Design Activity
DOM	Document Object Model
DVD	Digital Versatile Disc
HTML	HyperText Markup Language
ISO	International Organization for Standardization
JOGL	Java binding for OpenGL
JVM	Java Virtual Machine
NCSA	National Center for Supercomputing Applications
RIA	Rich Internet Applications
URL	Uniform Resource Locator
VRML	Virtual Reality Modeling Language
WWW	World Wide Web
X3D	Extensible 3D
XML	Extensible Markup Language
XUL	XML User Interface Language

Capítulo 1

Introdução

1.1 Enquadramento e Motivação

O método mais eficaz de absorvermos informação do que nos rodeia é através de imagens. A partir da visualização de uma imagem conseguimos adquirir numa fração de segundo uma quantidade de dados que nenhum outro sentido nos é capaz de fornecer. Esta premissa aplica-se nos diferentes contextos da nossa vida, o que torna fácil de compreender a relevância crescente que tem sido dada à Computação Gráfica nas aplicações informáticas.

A compreensão da importância da computação gráfica iniciou-se já nos anos 60. O primeiro grande exemplo de uma aplicação gráfica foi o Sketchpad de Ivan Sutherland[1]. Neste *software* era possível usar uma caneta ótica para desenhar e interagir com formas simples no ecrã.

Além da sua importância para a computação gráfica, este sistema contribuiu de forma fundamental na área da Interação Pessoa-Computador, sendo uma das primeiras aplicações gráficas interativas, servindo de base para vários *standards* de interfaces gráficas ainda hoje aplicados.

A presente dissertação enquadra-se precisamente no contexto das aplicações gráficas interativas, pretendendo explorar uma lacuna ainda existente nas versões mais recentes destas aplicações: a disponibilização destas de forma nativa na Web.

O cidadão comum vê-se cada vez mais dependente da internet para a realização de diferentes tarefas do seu dia-a-dia, sejam estas lúdicas ou laborais. Com essa evolução do modo de interação com o mundo da *World Wide Web*, foi necessário que a criação tecnológica evoluísse de igual forma para responder às novas exigências dos utilizadores, fornecendo-lhes constantemente novos e melhores métodos de realizarem as tarefas desejadas diretamente na Web, tornando-as acessíveis de qualquer computador com ligação à internet, sem dispêndio de tempo em instalações de *software* ou ocupação de espaço de armazenamento.

INTRODUÇÃO

É no seguimento desta evolução que a presente dissertação procurou investigar e desenvolver uma aplicação gráfica interativa em 3 dimensões para exploração de um ambiente urbano virtual. Esta está acessível a partir de um qualquer browser e tem por base o estudo das recentes evoluções das tecnologias para a Web.

O projeto foi desenvolvido conjuntamente com a empresa 3Decide[2], no contexto de um produto que a mesma se encontra a desenvolver, o aLIVE Places¹.

Este produto oferece aos seus clientes a oportunidade de promover um espaço, fornecendo uma representação 3D deste, desenvolvida na ferramenta Unity3D. Esta representação, criada a partir de plantas CAD e gerada através da ferramenta CityEngine², permite que o utilizador final execute um conjunto de interações, analisando diferentes andares, salas e outras estruturas do edifício permitindo-lhe aceder a um aglomerado de informações associadas às mesmas, relativamente a entidades a que pertencem, serviços que fornecem, atividades que estão a ocorrer nesse momento, ou qualquer outro tipo de informação que o cliente lhe deseje associar. Todos estes dados poderão ser atualizados, e estas representações terão a possibilidade de ser integradas com redes sociais ou outros *websites*.

Este é um produto que pretende oferecer a empresas e outras organizações uma ferramenta inovadora de partilha de informação relevante de um modo prático, dinâmico e apelativo com o objetivo de deixar uma marca positiva juntos dos utilizadores finais, beneficiando de forma essencial as instituições clientes.

1.2 Descrição do Problema

A adoção em massa da *World Wide Web* causou uma mudança radical no desenvolvimento de *software*, fazendo com que uma grande parte das aplicações anteriormente desenvolvidas apenas para *desktop* estejam atualmente a ser desenvolvidas num ambiente Web.

Embora os conteúdos gráficos 3D estejam já bastante explorados e desenvolvidos, não só no contexto dos jogos digitais, mas também em aplicações de simulação (médicas, militares...), a maioria destes *softwares* ainda chegam às nossas casas pelos meios de distribuição tradicionais, tais como Digital Versatile Discs (DVDs) ou Blu-Ray Discs (BDs). A migração destes conteúdos gráficos do ambiente *desktop* para o contexto Web é ainda um problema.

Apesar de as tecnologias que permitem a *renderização* 3D acelerada por *hardware* num *browser* já existirem desde cerca de 2001, as dificuldades de instalação do *software* requerido e os problemas relacionados com as variações de performance em diferentes plataformas, fizeram com que estas tecnologias não tenham sido bem-sucedidas junto do grande público. Alguns especialistas afirmam mesmo que a razão para a falta de sucesso destas tecnologias foi o facto destas só se focarem nos gráficos 3D e terem por isso desprezado a importância da interação com

¹ aLIVE Places. <http://www.3decide.com/content/aliveplaces>

² Esri CityEngine. <http://www.esri.com/software/cityengine>

INTRODUÇÃO

o utilizador, não tendo realizado qualquer investigação sobre técnicas ótimas para as interfaces Web baseadas em hipertexto[3].

Mas, com a emergência de novas tecnologias capazes de trazer ao *browser* do utilizador comum um ambiente tridimensional com novas e otimizadas formas de interação, a evolução destes *softwares* torna-se uma necessidade.

É neste contexto que se insere o problema desta dissertação: como criar um sistema de representação de ambientes urbanos baseado nas mais recentes e promissoras tecnologias 3D para a Web, que tenha em conta a interação com o utilizador e que torne estes ambientes e tecnologias bem aceites pelo público geral, levando-os à sua utilização habitual em situações do dia-a-dia, especificamente, dado o âmbito do protótipo, na participação de um evento.

Para isso é necessário investigar um conjunto de temáticas-chave. A primeira dessas será a representação de ambientes urbanos tridimensionais procurando métodos de apresentação de cenários com carácter acessível para o utilizador comum. Revela-se igualmente importante estudar como é concretizada a promoção de locais dedicados à realização de eventos e como esta informação alcança os utilizadores. Para além disto, terão de ser estudadas abordagens para a construção de interatividade nos cenários representados, tendo em conta os conceitos de interação entre pessoa e computador. Num carácter mais técnico será necessário investigar, comparar e seleccionar uma tecnologia apropriada à resolução do problema proposto.

1.3 Objetivos

O principal objetivo deste projeto de dissertação consiste no desenvolvimento de um ambiente urbano tridimensional e interativo que esteja acessível de forma nativa num *browser* e que permita a divulgação de um espaço a um vasto e diversificado número de utilizadores. Para concretizar este objetivo principal é necessário concretizar os seguintes objetivos específicos:

- Investigar o estado da arte quanto a soluções tecnológicas para a *renderização* 3D em *browsers*;
- Estudar técnicas de interação com ambientes 3D otimizadas para o contexto Web;
- Desenvolver uma metodologia de criação da cena 3D na tecnologia Web a partir dos modelos exportados da ferramenta CityEngine;
- Criação do módulo de interação com o ambiente tridimensional;
- Implementar a comunicação com a API e o *WebService* existentes;
- Desenvolver um protótipo aplicando as funcionalidades referidas nos pontos anteriores ao contexto de uma promoção de um evento;
- Realizar testes e uma avaliação do protótipo implementado;
- Escrita da documentação final sobre o projeto desenvolvido.

1.4 Resultados Esperados

No âmbito do projeto a desenvolver, espera-se no final obter uma solução integrada com o produto aLIVE Places. Esta solução irá consistir numa ferramenta de divulgação de um espaço por parte de um cliente a qualquer tipo de utilizador, já que a representação tridimensional do espaço estará acessível a partir de um *browser* e sem qualquer outro requisito. A plataforma deverá ter em conta técnicas de interação apropriadas à *World Wide Web*, de forma a tornar a solução apelativa e simples de aceder por parte do cibernauta. No âmbito mais específico do protótipo a realizar, os conceitos mencionados serão utilizados para a promoção de um evento no espaço a divulgar, fornecendo um conjunto de interações úteis a este contexto, tendo em conta as características humanas associadas à aquisição de informação espacial.

Além dos resultados anteriores, espera-se conseguir obter conclusões quanto à viabilidade da criação de cenários tridimensionais disponíveis diretamente a partir de um *browser* e quanto à praticabilidade e utilidade da aplicação destes a problemas do dia-a-dia do utilizador comum.

1.5 Trabalho Relacionado

Os artigos de Grupp[4] e Ortiz[5] serviram de base de estudo das tecnologias de desenvolvimento. Através da apresentação e comparação das tecnologias mais relevantes, acompanhadas de opiniões de especialistas diretamente envolvidos no desenvolvimento de algumas destas tecnologias e de dados noticiados ao longo das evoluções das mesmas, estes artigos auxiliaram a escolha da melhor tecnologia, nomeadamente no contexto das aplicações 3D para a Web com renderização acelerada por *hardware*, a utilizar para o presente projeto. A escolha recaiu sobre o WebGL devido essencialmente à sua notável e atual evolução e às características *open-source*, *plugin-free* e *cross-platform* que apresenta.

Dada a importância da interação com o sistema, e o contexto da promoção de um evento a ser desenvolvido para o protótipo, o artigo de Chittaro et al.[6], assim como os estudos realizados por Oulasvirta et al.[7] foram de incrível importância para compreender como o ser humano adquire informação sobre um espaço e sobre a navegação de percursos nesse espaço, e de como os utilizadores reagem aos diferentes formatos de disponibilização dessa informação. Em Chittaro et al., e tendo por base um caso de estudo com um mapa 3D com GPS, as conclusões tiradas foram de que as pessoas não tinham dificuldades em associar objetos reais a objetos da cena tridimensional, o que mostra as vantagens deste formato em situações de orientação e localização por parte dos utilizadores. Já nos estudos de Oulasvirta et al., comparando situações de orientação mas também navegação para alcançar uma localização de destino através de mapas 2D e 3D, os resultados mostraram que apesar da conclusão de Chittaro et al. se verificarem, nos mapas 3D os utilizadores apresentavam dificuldades de localização relativa ao destino a alcançar, acabando por vezes por se perderem, nomeadamente por falta de uma perspetiva ‘geral’. Estes são dados de

vital relevância tidos em conta na construção da interação do presente projeto, de forma a alcançar uma melhor aceitação por parte dos utilizadores finais.

Por último, no contexto dos projetos semelhantes ao nível da representação de ambientes urbanos virtuais tridimensionais, aqueles que possibilitaram uma mais próxima comparação com a solução a desenvolver, permitindo assim avaliar as diversas componentes a abordar, foram os projetos de divulgação do campus da Universidade de Agricultura de Shenyang[8], e também o sistema de promoção da mesquita Alaca Imaret[9]. Estes relacionam-se também de forma próxima com a ideia de divulgação de uma representação espacial para as grandes massas, conceito objetivo do presente projeto.

1.6 Metodologia

Para abordar o problema desta dissertação a metodologia a seguir compreendeu cinco fases distintas. A primeira fase consistiu na pesquisa do estado da arte das temáticas envolvidas pelo projeto. De seguida, existiu uma etapa de levantamento de requisitos do sistema junto com a análise dos módulos existentes no produto 3DPlace. A fase seguinte disse respeito à implementação do módulo de forma iterativa e incremental em conjunto com a integração com o back-office do produto aLIVE Places, sendo também realizados simultaneamente testes ao sistema. A seguir, realizou-se o protótipo aplicando as funcionalidades já desenvolvidas na fase anterior ao contexto de um evento. E por fim houve uma fase de testes finais, avaliação da solução construída e elaboração da documentação necessária.

1.7 Estrutura do Documento

O presente relatório é composto por um capítulo 2 dedicado à investigação do estado da arte das temáticas e das tecnologias relevantes ao projeto realizado. De seguida, no capítulo 3 é abordada a solução a construir, explicitando o seu contexto, os seus detalhes, os requisitos e os casos de utilização a ter em conta no processo de construção. O capítulo 4 diz respeito à implementação da solução definida, apresentando todas as fases desde a construção do ambiente 3D, passando pelo módulo base de informação de espaços e terminando com uma secção dedicada ao protótipo relativo à divulgação de eventos. Posteriormente, no capítulo 5, é feita a análise e avaliação de resultados obtidos com os testes realizados à aplicação e finalmente, no capítulo 6, são apresentadas as conclusões tiradas e o trabalho futuro.

Capítulo 2

Revisão Bibliográfica

Ao longo deste segundo capítulo será abordado o estado da arte das diversas áreas envolvidas no projeto de dissertação. A primeira secção a surgir diz respeito à representação de ambientes urbanos virtuais tridimensionais, mencionando a crescente utilização destes ambientes e alguns projetos relacionados. Já na segunda secção, é abordada a temática da interação pessoa-computador e a importância desta interatividade para o sucesso da aplicação junto do utilizador final. É analisada a eficácia de diferentes métodos de representação de informação sobre ambientes urbanos e as preferências dos utilizadores. A terceira secção procura investigar aplicações direcionadas à divulgação de eventos. O quarto segmento debruça-se sobre o estudo da evolução das tecnologias Web3D mais relevantes, desde o VRML ao recente WebGL, comparando e analisando estas tecnologias no contexto da atualidade. E por último, a quinta secção é dedicada à justificação da tecnologia selecionada.

2.1 Representação de Ambientes Urbanos Virtuais em 3D

Ao longo dos anos e com a evolução da qualidade dos conteúdos gráficos, nomeadamente devido à crescente disponibilização de melhores componentes de *hardware* informático a preços mais acessíveis, e ao acesso mais globalizado, também com menores preços, de serviços de internet com maiores larguras de banda, têm surgido novas formas de aplicação de ambientes virtuais na resolução de problemas da sociedade moderna.

Existem alguns formatos de aplicação destes ambientes que se tornaram mais popularizados, como é o caso do famoso Second Life[10], uma plataforma de mundo virtual focada na interação social, mas que tem sido também utilizada para exploração de outros contextos como sistemas de planeamento urbano[11]. Uma outra aplicação destes ambientes que tem sido testada e experimentada por vários desenvolvedores, consiste na criação de centros

REVISÃO BIBLIOGRÁFICA

comerciais virtuais, pelos quais um utilizador pode passear avistando lojas e suas montras, e fazer as suas compras, se assim o desejar, sem sair do conforto do sofá[12][13]. Outro sistema, cada vez mais popular, diz respeito a ambientes de simulação em situações de desastres naturais ou outras emergências. Nestes casos, o ambiente virtual permite aos seus utilizadores adquirirem conhecimento e habituação das práticas a seguir, evitando as grandes despesas e gastos de tempo em simulações em ambientes reais.

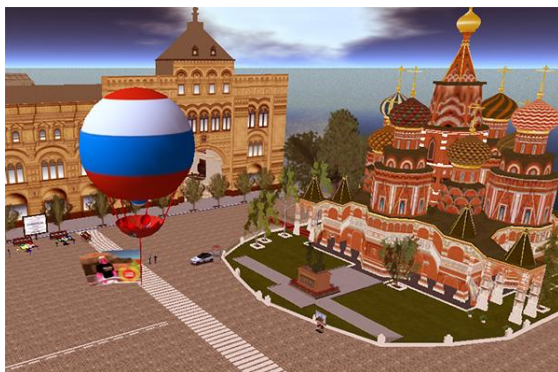


Figura 1 - Exemplo de cenário da plataforma Second Life



Figura 2 - Exemplo de um ambiente virtual aplicado a um centro comercial [14]

Porém, existem outras áreas da sociedade ainda menos exploradas por estes ambientes, para as quais começam a surgir aplicações inovadoras.

Na área da arquitetura e da construção de edifícios, um projeto foi desenvolvido por Xiao et al.[14], que tem como objetivo, a partir da representação virtual de um ambiente, simular o processo de construção. A engenharia por trás da construção de edifícios consiste num processo complexo em que muitas variáveis devem ser tidas em conta. Atualmente, a otimização e organização dos métodos usados nesse processo baseiam-se na experiência humana. Acrescentar um sistema de simulação que permita, ainda na fase de planeamento, prever dificuldades e possíveis entraves ao desenvolvimento da construção, e depois, numa fase de desenvolvimento, acompanhar todo o processo de forma a verificar a segurança dos procedimentos, como por exemplo em situações de levantamento de objetos de enorme peso, provou-se como uma ferramenta de extraordinário valor, já que permitiu poupar tempo, evitar acidentes, garantir uma maior qualidade do edifício e claro, poupar muito dinheiro.

Quanto à área da promoção de um espaço, seja por motivos de trocas de experiência culturais ou para divulgação a possíveis interessados ou turistas, um projeto foi desenvolvido por Ping et al.[8]. Neste projeto, foi construído um ambiente virtual representativo da Universidade de Agricultura de Shenyang pelo qual o utilizador pode navegar, fazendo um tour em modo manual ou automático, e com o qual pode interagir. É um sistema que permite à universidade dar-se a conhecer, a turistas, curiosos ou mesmo novos e futuros alunos de uma forma simples,

REVISÃO BIBLIOGRÁFICA

apelativa e mais completa, já que apresenta ao utilizador um conjunto de informação histórica não observável no local.



Figura 3 - Sistema de tour virtual na Universidade de Agricultura de Shenyang

Ainda relacionado com a promoção de espaços, mas em combinação com as áreas da arquitetura e recuperação de edifícios, Oudatzi[9] desenvolveu um ambiente virtual representativo de um monumento histórico, atualmente em ruínas, na sua estrutura original. Baseando-se em plantas de reconstrução do edifício e com base em dados históricos, criou um modelo da mesquita Alaca Imaret, na Grécia, com o qual o utilizador final pode interagir, aprendendo mais sobre a sua história e impacto arquitetónico, o qual não poderia ser feito por outro método devido à degradação do edifício.



Figura 4 - Ambiente virtual do projeto de reconstrução da mesquita Alaca Imaret [10]

O que todos os casos referidos nesta secção têm em comum com o projeto da presente dissertação é o contexto da representação de um ambiente urbano virtual em três dimensões. No entanto, alguns têm ainda outras semelhanças, nomeadamente o contexto da promoção de um espaço, como é o caso dos projetos de Oudatzi[9] e Ping et al.[8], onde o objetivo final dessa representação é a divulgação desse espaço para que um grande número de utilizadores possa interagir e adquirir conhecimentos sobre o ambiente aí representado. Este é também o objetivo do resultado final desta dissertação.

2.2 Interação em ambientes urbanos

A área da interação pessoa-computador estuda como as pessoas interagem com sistemas informáticos. Mais especificamente, corresponde a quão habilitada está uma aplicação informática para ser utilizada, de forma simples e eficaz, por um utilizador[15].

Historicamente, quando os computadores surgiram para o público geral, a qualidade da interação ou as interfaces de uma aplicação não eram características tidas em conta pelos programadores de *software*. Todavia, com o crescimento da utilização de sistemas informáticos e uma maior exigência por parte dos utilizadores, foi necessária uma evolução de mentalidades. Afinal, por mais sofisticada que seja uma aplicação, o seu valor é nulo se o utilizador não a conseguir utilizar de forma apropriada para atingir os seus fins. Atualmente, já um grande número de importantes empresas e instituições académicas se dedicam ao estudo desta temática, que se revelou de excepcional importância na indústria do software[16][17].

Focando agora a análise na interação com ambientes urbanos, nomeadamente em representações *bi* ou tridimensionais destes ambientes, seja interiores ou exteriores, este é um problema sem uma solução genérica.

Existem diversos tipos de aplicações baseadas em mapeamento de ambientes urbanos. Inicialmente a maioria estava relacionada com aplicações para uso em veículos, nomeadamente em dispositivos GPS, e baseavam-se apenas em mapas 2D. Atualmente existem outras aplicações, como reconstruções de ambientes históricos baseados em descobertas arqueológicas[18], plataformas para visitas virtuais a museus[19] ou a centros comerciais[12][13], estas apoiadas em representações tridimensionais, interiores ou exteriores, de diferentes espaços.

A navegação é portanto uma aspeto essencial da interação com um ambiente urbano virtual. Um utilizador precisa de observar e compreender o espaço à sua volta de forma a conseguir encontrar o caminho até um local específico. Mas, quando a interatividade com a aplicação é fraca, esta descoberta pode tornar-se numa atividade muito complicada e frustrante[6]. Para evitar estes problemas, os programadores devem ter em conta algumas considerações, relativas à forma como o ser humano adquire conhecimento sobre o espaço e rotas, no processo de construção de um sistema deste género.

Um ser humano baseia-se em dois tipos de conhecimentos para navegar por um espaço: conhecimento de rota, adquirido duma perspectiva egocêntrica que representa um percurso familiar entre localizações específicas; e conhecimento de planta, desenvolvido duma perspectiva exocêntrica (vista de 3ª pessoa) e que representa relações entre diferentes localizações, permitindo ao utilizador encontrar caminhos alternativos entre esses locais[6].

Voltando à questão de qual o método que permite uma melhor aquisição de informação sobre um espaço e/ou rotas, diferentes autores têm defendido ideias distintas. Laakso[20] afirmou que um mapa 2D era mais rápido de utilizar para resolver problemas de orientação ou de procura de um percurso mas que um mapa 3D era ‘mais divertido’. Por outro lado, depois de realizarem alguns testes, Burigat e Chittaro[21] concluíram que os utilizadores correspondiam, sem dificuldades, objetos reais com objetos da vista 3D. Ao analisar o feedback dos utilizadores de um estudo realizado por Oulasvirta et al.[7], em que utilizadores tinham de se orientar por uma zona de um cidade a partir de uma representação, 2D ou 3D, dessa região, dirigindo-se até um local de destino, podemos retirar que as maiores dificuldades apontadas à perspectiva 2D eram: a dificuldade em descobrir a relação entre a orientação do mapa e a do utilizador e a falta de *landmarks* no mapa que os pudesse auxiliar na identificação de localizações. Já em relação à perspectiva 3D, a maior dificuldade consistia na falha da perceção da sua posição relativa ao destino final, por falta de uma ‘vista geral’ da zona.

Rakkolainen e Vainio[22], depois de alguns estudos com utilizadores, apuraram que estes tinham maior preferência por uma interface com uma vista combinada de 2D mais 3D do que por qualquer uma destas por si só. Concluindo, é ainda uma questão em aberto qual é a melhor representação para cada tipo de tarefa, porém, parece ser mais consensual que a solução mais abrangente é uma combinação dos dois formatos.

2.3 Divulgação de espaços de eventos

Um evento consiste numa iniciativa que tem como objetivo reunir pessoas com finalidades diversas, sejam eventos sociais, culturais, profissionais ou até académicos. Estes eventos podem ser dirigidos a um pequeno grupo de pessoas, ou a grandes multidões. Tendo em consideração estes últimos, a sua divulgação ao público-alvo torna-se de maior dificuldade.

Os diversos espaços que se dedicam a este tipo de eventos necessitam constantemente de encontrar formas eficazes de chegar aos seus destinatários e apresentar-lhes toda a informação relevante para a sua visita ao evento. Quando este público é também desconhecedor do espaço onde o evento ocorre, a informação a fornecer aos destinatários tem necessariamente de incluir detalhes espaciais do local do evento e zonas circundantes, de forma a auxiliar o visitante a orientar-se durante a sua visita. Sobre este ponto encontram-se graves dificuldades de comunicação - Como divulgar ao público todas as informações relevantes de um espaço de eventos de uma forma eficaz?

REVISÃO BIBLIOGRÁFICA

O modo mais básico consiste em enviar, junto com um convite contendo a agenda, uma espécie de planta do espaço, preenchido de pequenos indicadores que são explicados a partir de uma longa legenda a acompanhar. Quanto maior o espaço e a diversidade de oferta a nível de atividades mais complexa fica a compreensão de tal planta.

Com a evolução tecnológica e a expansão da internet, este tipo de representação passou a ser divulgado e acessado a partir da Web. Estas representações mantêm, muitas vezes, a estrutura e o aspeto dos mapas anteriores como pode ser visto na aplicação do London Convention Center³ apresentada na Figura 5. Dado o seu carácter interativo, é possível ter imagens associadas às salas, oferecendo ao utilizador uma representação um pouco mais completa. No entanto, a interatividade nesta aplicação limita-se a isso, visualização de fotografias, não tendo o utilizador acesso a qualquer outro tipo de interação ou informação além do nome das salas visível no mapa.

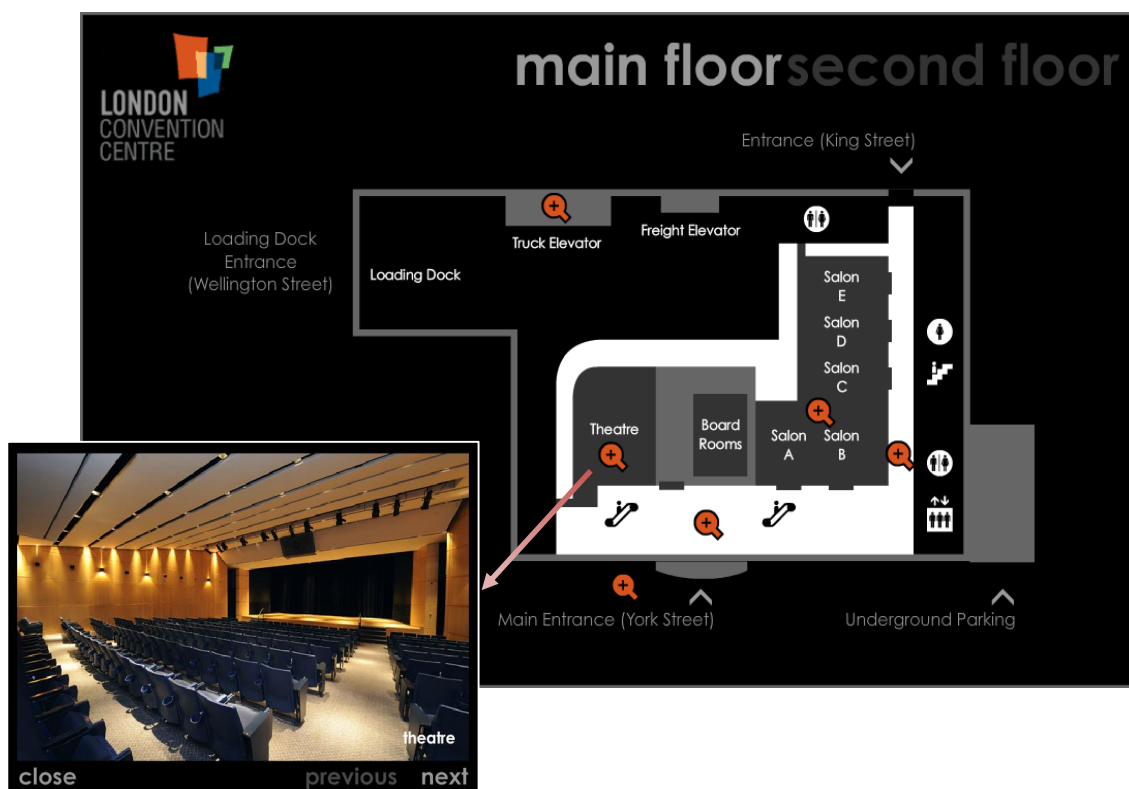


Figura 5 - Aplicação interativa do London Convention Center

Já o Vancouver Convention Center⁴ dispõe de uma aplicação que oferece uma espécie de visualização tridimensional, permitindo até selecionar os pisos a partir da interação com essa representação. Contudo, volta-se a ter a planta 2D típica quando se entra num piso. Aqui

³ <http://www.londoncc.com/sites/default/files/flash/floorplans.swf?width=960&height=600&iframe=true>

⁴ <http://www.vancouverconventioncentre.com/wp-content/uploads/2009/03/vcec.swf>

consegue-se fazer um pouco mais do que no exemplo anterior, tendo a possibilidade de dividir salas de acordo com o desejado e consultar alguns dados de cada uma destas. Já a usabilidade e a interface da aplicação (Figura 6) têm um ar antiquado e pouco agradável.

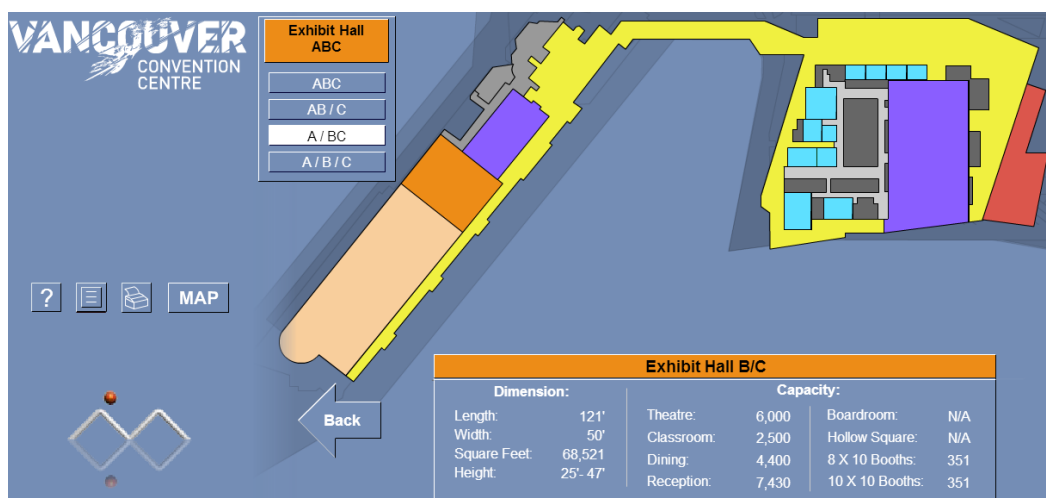


Figura 6 - Aplicação interativa do Vancouver Convention Center

Outro espaço de eventos que possui um mapa interativo na sua página é o Walter E. Washington Convention Center⁵. Nesta aplicação cada um dos pisos é representado de forma tridimensional e permitem ser selecionados para uma vista mais aproximada. Quando se seleciona uma sala, é possível aceder noutra página a todos os detalhes sobre a mesma assim como uma representação da sala, preenchida de forma diferente conforme a seleção.

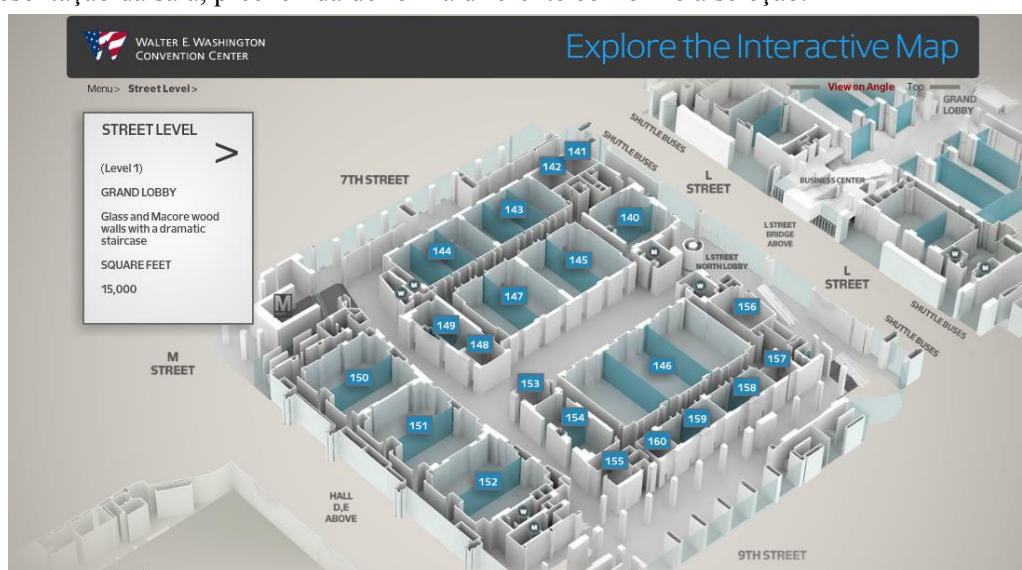


Figura 7 - Aplicação interativa do Walter E. Washington Convention Center – Vista de piso

⁵ <http://www.dcconvention.com/InteractiveMap.aspx>

REVISÃO BIBLIOGRÁFICA

Estas representações, apresentadas nas Figura 7 e Figura 8, são claramente mais realistas e mais apelativas do ponto de vista estético. Além disto a experiência de usabilidade é mais agradável para o utilizador assim como mais informativa dada a quantidade de detalhes apresentados.

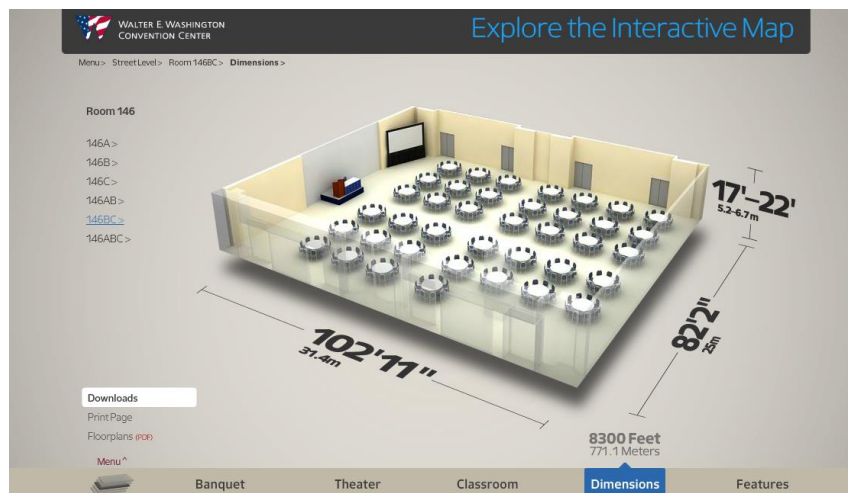


Figura 8 - Aplicação interativa do Walter E. Washington Convention Center – Vista de sala

Todavia, esta aplicação, assim como as anteriormente mencionadas, não contém qualquer ligação com os eventos que decorrem nos seus espaços. São direcionadas essencialmente para os organizadores dos eventos, e não tanto para o utilizador final. Este continuará a necessitar de consultar outra página, específica ao evento, de forma a obter dados das atividades que deseja frequentar e depois fazer, por si só, a relação com o espaço representado na aplicação do local.

Nenhum destes sistemas oferece uma representação integrada do espaço dos eventos com a informação relevante aos respetivos eventos.

2.4 Tecnologias Web3D

Quando a Internet surgiu, o seu funcionamento era apenas textual, não existiam gráficos. Essa realidade alterou-se em 1993, aquando do lançamento do *Mosaic* pelo NCSA (*National Center for Supercomputing Applications*), este foi o primeiro *browser* capaz de apresentar conteúdos multimédia integrados com texto[5]. O seu lançamento representou a abertura da *World Wide Web* para o público geral, numa época em que a internet se expandia rapidamente.

Atualmente, uma enorme diversidade de tarefas podem ser realizadas via Web, a maioria destas explorando os conteúdos multimédia de forma a tornar a experiência mais apelativa para o

utilizador. No entanto, um elemento essencial ainda não deixou a sua marca neste ambiente: o 3D. Ao longo dos anos, têm surgido algumas tecnologias 3D a tentar singrar na *WWW*, muitas destas com funcionamentos idênticos apenas utilizando diferentes formatos.

2.4.1 VRML

Lançando a sua primeira versão em 1995, o VRML, *Virtual Reality Modeling Language*, foi o primeiro formato a três dimensões direcionado para a Web com o objetivo de incrementar a experiência do utilizador num *browser*. Consiste numa linguagem de descrição de cenas 3D, que permite a decodificação e encapsulamento de conteúdos tridimensionais para a *World Wide Web*. Foi a linguagem que deu origem ao conceito Web3D antes mesmo deste termo surgir para descrever tais tecnologias[23].

Dois anos depois do seu lançamento, tornou-se o primeiro *standard* ISO para criação e visualização de conteúdos 3D na Internet, atraindo mais seguidores[24].

Sendo esta uma linguagem pensada para alcançar as massas, foi criada com o objetivo dos seus conteúdos tridimensionais estarem acessíveis a partir de qualquer sistema operativo, de qualquer browser e com um bom funcionamento, mesmo com baixas larguras de banda. Estas características ditaram o seu sucesso, começando a surgir plug-ins VRML, *players* VRML e muitas cenas 3D criadas por programadores independentes[23].

No entanto, a versão 1.0 do VRML estava limitada à *renderização* de cenas estáticas e não permitia interação com as mesmas, o que levou ao lançamento duma segunda versão no final de 1996. O VRML 2.0 permitiu a criação de ambientes mais complexos permitindo a inclusão de som, animação e interação com a cena. Foi também a base para o *standard* ISO conhecido como VRML97[23].

Apesar da sua evolução, o VRML continuou a apresentar limitações, nomeadamente no que diz respeito a *streaming* de vídeo ou múltiplas texturas em objetos. No entanto foi uma tecnologia pioneira que inspirou o desenvolvimento de novas tecnologias Web3D como é o caso do *standard* X3D, também conhecido como VRML Next Generation.

2.4.2 X3D

A tecnologia Extensible 3D, *standard* ISO desde 2004, é uma arquitetura *run-time* e um formato de ficheiro capaz de representar e comunicar ambientes e objetos 3D utilizando a linguagem XML[25].

O X3D, assim como todas as versões do VRML, foi criado e promovido pelo Web3D Consortium. O Web3D Consortium é uma organização sem fins lucrativos que tem como objetivo a criação de *open standards* para especificações Web3D e que patrocina programas de educação do mercado e de utilizadores para acelerar a procura de produtos baseados nesses *standards*[26].

Consistindo numa terceira geração do VRML, o X3D foi desenvolvido com o propósito de colmatar as suas falhas e limitações. Possui gráficos de maior qualidade, *real-time* e interativos, e inclui áudio e vídeo além da informação 3D. A sua integração com XML permite uma mais simples comunicação com *WebServices*, redes distribuídas e diferentes plataformas de desenvolvimento[25].

Assim como no caso do VRML, para *renderizar* uma cena X3D num *browser* é necessária a instalação de um plug-in. Também existem *players* para interpretação e renderização de cenas tridimensionais codificadas em X3D para aplicações *desktop*.

As funcionalidades mais maduras deste formato, tornaram-no útil no desenvolvimento de aplicações para diversas áreas, seja nas mais óbvias áreas da engenharia e arquitetura, na área da medicina (para treino e simulação), na educação ou no entretenimento[25].

2.4.3 Java3D e JOGL

O Java3D, lançado oficialmente em 1998, é uma API que permite o desenvolvimento de aplicações gráficas 3D, com base em grafos de cena, para a popular linguagem de programação Java.

Como uma extensão da linguagem Java, localizando-se numa camada superior, a interface Java3D é composta por um conjunto de classes que permite aos programadores adicionar gráficos 3D assim como sons às suas aplicações ou *applets*[23].

O Java3D tem semelhanças com o VRML e o X3D, sendo as três baseadas num grafo de cena numa estrutura em árvore que permite armazenar, organizar e *renderizar* os componentes do ambiente 3D. É também capaz de carregar ficheiros VRML dando aos utilizadores a possibilidade de aproveitar os modelos VRML já existentes[23].

Outra API existente para a linguagem Java é o Java OpenGL. Esta tecnologia consiste num *binding* Java para OpenGL e já existe desde o lançamento do OpenGL 1.3[4].

É parte de um conjunto de tecnologias *open-source* criadas pelo Game Technology Group na Sun Microsystems[27]. Disponibiliza também acesso completo às funções de OpenGL e pode ser integrada com o Abstract Windows Toolkit (AWT) e com o conjunto de *widjets* Java GUI (Swing)[28].

Recentemente, o Java3D passou a utilizar o JOGL 2.0 para a sua *renderização* OpenGL acelerada por *hardware*[29].

Esta foi a primeira plataforma a trazer para a *World Wide Web*, através do uso de *applets* Java, gráficos 3D acelerados por *hardware*. No entanto, para um utilizador ver Java *applets* no seu *browser* necessita de instalar a JVM e um plug-in Java, o que pode ser uma tarefa difícil para o utilizador comum, podendo esta ser uma das principais razões desta plataforma não ter tido grande sucesso junto do grande público[4].

2.4.4 Flash

O Adobe Flash Player é uma das plataformas mais populares para criação de aplicações interativas como animações, jogos web, Rich Internet Applications (RIA) ou páginas web.

A Adobe introduziu o suporte a conteúdos 3D com o lançamento do Flash Player 10, em 2008. Segundo Tom Barclay, gestor sénior da Adobe para o Flash Player, foram adicionadas novas classes e métodos, particularmente a capacidade de definir a posição tridimensional dum objeto na linguagem de programação ActionScript de forma a disponibilizar efeitos 3D[5].

A classe Stage3D, presente no Flash Player, é a responsável por oferecer uma arquitetura acelerada por *hardware* para que o programador possa desenvolver os seus conteúdos 3D, podendo o mesmo também fazer uso de um vasto conjunto de *frameworks* existentes para o player (Flare3D, Alternativa3D, Away3D, Minko...)[30].

Como as tecnologias mencionadas até ao momento, o Flash requer a instalação de plug-ins para o seu funcionamento num *browser*. Tem também a desvantagem do desenvolvedor necessitar de usar as suas ferramentas proprietárias para criar aplicações web 3D.

2.4.5 Silverlight 5

O Silverlight da Microsoft consiste numa *framework* com funcionalidades e objetivos semelhantes ao Adobe Flash. Na sua versão 5.0 lançada em Dezembro de 2011 passou a disponibilizar suporte a conteúdos gráficos 3D acelerados por *hardware*, trazendo para o ambiente do *browser* web algumas das funcionalidades disponibilizadas no seu *software* XNA.

O *toolkit* do Silverlight proporciona uma compatibilidade melhorada com XNA, incluindo *ContentManager*, *Model*, *SpriteBatch*, *Mouse*, *Keyboard*, *Custom Effects* e novos *templates* de Visual Studio para iniciar o desenvolvimento com o Silverlight 3D.

Assim como a maioria, e dado que se trata de um *software* proprietário, a sua utilização está dependente da instalação de um plug-in e tem problemas de compatibilidade com algumas plataformas, nomeadamente com determinados sistemas operativos e *browsers* não pertencentes à Microsoft[31].

Quanto aos dispositivos móveis o acesso a esta tecnologia está também limitado àqueles com sistema operativo Windows Phone, nas versões a partir da 7.0.

2.4.6 WebGL

Em 2009, o Khronos Group anunciou que estava a trabalhar numa tecnologia *open-source* capaz de *renderizar* gráficos 3D acelerados por *hardware* num *browser*, sem ser necessário instalar qualquer plug-in: o WebGL. Na sua especificação estiveram envolvidos grandes empresas das áreas dos *browsers* web e dos conteúdos gráficos como a Google, Mozilla, Opera, Apple, NVIDIA e AMD[32]. A versão 1.0 desta tecnologia foi lançada em Março de 2011 e atualmente

é suportada pela maioria dos *browsers*: Safari, Google Chrome, Firefox e Opera, com exceção do Internet Explorer – apesar do seu suporte estar a ser previsto para a próxima versão do *browser*.

O WebGL é baseado no OpenGL ES 2.0 e disponibiliza uma API para *renderização* de gráficos 3D acessível a partir da linguagem de programação Javascript. Para isso utiliza o elemento *canvas* do HTML5 e é acedido através de interfaces DOM (Document Object Model). O elemento *canvas* consiste num componente do HTML5, uma espécie de contentor que pode ser utilizado para *renderizar* gráficos, em tempo real, a partir de linguagens de *scripting* (habitualmente Javascript).

A grande distinção entre esta tecnologia e as outras referidas ao longo desta secção é que o WebGL não requer nenhuma instalação para a sua visualização num web *browser*, o que se torna uma grande mais-valia especialmente para alcançar o grande público. O utilizador final não precisa de gastar tempo nem de ter nenhum conhecimento ou habilidade específica para visualizar de forma imediata no seu *browser* o conteúdo 3D *renderizado* pelo WebGL.

No entanto, nem tudo são vantagens. O WebGL é uma linguagem bastante baixo nível, dada a sua conexão direta com o OpenGL, o que significa que para programar uma *renderização* de um simples sólido geométrico, por exemplo, são necessárias várias linhas de código, o que se torna muito pouco interessante do ponto de vista do programador. Mas como ‘a necessidade é a mãe da invenção’ rapidamente surgiram desenvolvedores a criar e partilhar bibliotecas que permitem uma programação de ambientes 3D em WebGL sem tanto esforço, ou seja mais alto nível. A seguir, está uma curta análise das bibliotecas mais populares.

SceneJS⁶: Disponibiliza as funcionalidades básicas de OpenGL. No entanto não é apropriada para desenvolvimento de cenas mais complexas já que o seu desenvolvimento nunca saiu da fase beta, sendo esta a única versão disponível na sua página web, datada já de 2011.

GLGE⁷: Suporte das funcionalidades básicas do OpenGL ES, desde luzes, materiais, modelos, câmaras e animações simples. Também tem suporte ao formato COLLADA. Necessita de definição de tipos de curvas explicitamente para animações mais complexas. Não tem sistema explícito de física e faltam-lhe classes de *Shader*.

CopperLicht⁸: Disponibiliza ao programador, além das funcionalidades base, animações, colisões, módulos matemáticos assim como opções de iluminação. Compila as *meshes* 3D em ficheiros binários de reduzido tamanho. Boa documentação e vários

⁶ *SceneJS - Homepage*. Obtido de: <http://scenejs.org/>

⁷ *GLGE - About*. Obtido de: <http://www.glge.org/about/>

⁸ *CopperLicht*. Obtido de: <http://www.ambiera.com/copperlicht/features.html>

demos e tutoriais na sua página Web. Pelo lado negativo, contém convenções de nomenclatura pouco consistentes com nomes de classes e métodos pouco explícitos.

CubicVR 3D⁹: Disponibiliza desde funções matemáticas básicas, *shaders*, COLLADA, alguma física e animações. Faltam efeitos mais avançados, como nevoeiro por exemplo, mas podem ser implementados usando *shaders* customizados.

X3DOM¹⁰: Esta *framework* experimental segue a especificação X3D para conteúdos web 3D sendo uma vantagem para desenvolvedores já familiarizados com o *standard*. Contém uma página web completa, com API detalhada e muitos tutoriais e demonstrações disponíveis. Algumas funcionalidades ainda pouco desenvolvidas.

Three.js¹¹: Pode ser usada com SVG e com o elemento *canvas* do HTML além do WebGL. Possui suporte às funcionalidades básicas, assim como animações, colisões e iluminação. Provavelmente a API mais popular para WebGL, e portanto fácil de encontrar tutoriais e demonstrações.

2.4.7 Outras Tecnologias

Estas não foram as únicas tecnologias de *renderização* 3D para a Web, no entanto, foram aquelas consideradas mais relevantes neste campo.

Outras tecnologias tiveram o seu papel na luta pelo domínio do conteúdo 3D na *World Wide Web*, algumas mesmo ainda em desenvolvimento e evolução. Algumas dessas são:

Ajax3D: Esta plataforma combina o standard para renderização 3D na Web, X3D, com a tecnologia AJAX, que permite o desenvolvimento de aplicações web rápidas e interativas de forma simples e célere[33].

3DMLW¹²: Plataforma *open-source* para a criação de conteúdos 2D ou 3D para a Web. Semelhante às tecnologias X3D ou Ajax3D, pois também codifica os seus conteúdos em ficheiros baseados no formato XML. Além disso, também necessita de um plug-in para funcionar num *browser* web. Como desvantagem, e comparando com a plataforma X3D, esta última possui recursos gráficos mais avançados do que esta tecnologia.

⁹ *CubicVR 3D* – Homepag.: <http://www.cubicvr.org/>

¹⁰ *X3DOM* – About. http://www.x3dom.org/?page_id=2

¹¹ *Three.js* - Homepage. <http://mrdoob.github.com/three.js/>

¹² *3DMLW*. <http://www.3dtech-rd.net/3dmlw>

O3D¹³: API Javascript *open-source* criada pela Google em 2009. Pode ser implementada num browser ou numa aplicação de desktop XUL (XML User Interface Language). Originalmente esta tecnologia utilizava uma arquitetura baseada em plug-in que permitia aos desenvolvedores integrar funcionalidades customizadas. No entanto, em Maio de 2010, a Google anunciou que o O3D ia passar a ser uma biblioteca Javascript funcionando sobre a plataforma WebGL.

2.5 Síntese / Escolha de Tecnologia

Após a investigação das tecnologias de *renderização* 3D disponíveis para a Web foram selecionadas apenas as 3 mais promissoras, especificamente as que apresentam um projeto mais atualizado e acessível ao utilizador comum e que permitem uma *renderização* acelerada por *hardware*. Foi criada a Tabela 1 para comparar as tecnologias a partir de uma análise mais detalhada.

	<i>Silverlight 5</i>	<i>Stage 3D (Flash)</i>	<i>WebGL (HTML5)</i>
Software	Proprietário	Proprietário	Livre
Browsers	IE6+, Firefox3+, Chrome, Safari	Todos	Versões recentes de Chrome, Firefox, Safari e Opera; Possivelmente no futuro IE11
Plugin	Sim	Sim	Não
Mobile	Apenas Windows Phone 7.x	Não	Blackberry e Android; iPad esperado para breve
Contras	Pouca aposta da Microsoft na componente 3D	Suporte a dispositivos móveis descontinuado	Desenvolvimento baixo nível; Ainda com diferenças de performance
Prós	Boa performance; Funcionalidades ligadas ao XNA	Ainda é a plataforma dominante nos browsers	Único Standard Web; Ligação com HTML5 que se encontra em crescimento

Tabela 1 - Comparação das tecnologias 3D para a Web

O3D. ¹³ <http://en.wikipedia.org/w/index.php?title=O3D&oldid=527550778>

Depois do estudo das tecnologias 3D disponíveis para a Web, a tecnologia eleita foi o WebGL. Esta decisão justifica-se pelo grande crescimento e evolução da mesma e pelas características singulares em relação às outras: o seu carácter *open-source* e o facto de ser única que permite a *renderização* de conteúdos 3D sem requerer instalação de plug-ins, tornando-se portanto muito mais acessível para um grupo diversificado de utilizadores. O facto de esta tecnologia ser o único *standard* web, apoiado por um conjunto de várias empresas relevantes na área da computação gráfica e dos *browsers* Web pode levar a que esta se torne, de forma mais madura, numa ferramenta de acesso globalizado permitindo aos seus desenvolvedores alcançar, de igual forma e sem desenvolvimento personalizado, os dispositivos, sistemas operativos e *browsers* de maior relevância.

De forma a contornar a característica *low-level* desta tecnologia foi também selecionada uma biblioteca mais alto nível construída sobre o WebGL, nomeadamente a *Three.js*. A escolha desta API está diretamente relacionada com a facilidade de obtenção de informação de aprendizagem, tal como exemplos, demonstrações e tutoriais além da já grande diversidade de funcionalidades que a mesma apresenta de momento.

2.6 Resumo e Conclusões

Tendo por base a investigação efetuada é fácil concluir que a utilização de ambientes virtuais tem-se tornado cada vez mais comum, e estes têm vindo a ser aplicados a diferentes áreas de estudo, não se ficando pela área de entretenimento. Apesar do contexto dos jogos ser aquele onde maior incidência se observou deste tipo de ambientes, atualmente estes já são utilizados para outros contextos mais sérios como os referidos neste relatório, e os quais podem trazer grandes vantagens a uma variedade de indústrias.

Porém, foi também visível que, no contexto web, estes ambientes ainda não se encontram tão desenvolvidos, o que se revela uma grande oportunidade para novos projetos, dado o crescimento exponencial da adesão dos utilizadores às aplicações e conteúdos na *Cloud*. Esta migração dos produtos para o mundo da *World Wide Web* está a ocorrer a grande velocidade, e a possibilidade de migração também de ambientes tridimensionais para este ambiente, mantendo, claro, um alto nível de qualidade e de velocidade, será certamente algo muito valorizado pelos utilizadores da Internet.

Outro grande crescimento que deve ser tido em conta, e que poderá ter relevância para o projeto a desenvolver, diz respeito à utilização de *tablets* e dispositivos móveis. A possibilidade de acesso a algo como o proposto no protótipo a realizar, uma espécie de convite para um grande evento representando um edifício de forma tridimensional e com possibilidade de interagir com

REVISÃO BIBLIOGRÁFICA

informação útil à navegação no edifício real, também num dispositivo destes, é algo que poderá igualmente acrescentar valor a este projeto.

Desta forma, justifica-se ainda mais a escolha da tecnologia WebGL para o desenvolvimento do presente sistema. Tendo em conta o crescimento da tecnologia HTML5 nos dispositivos móveis, além de nos computadores, e lembrado que o WebGL assenta no seu elemento *canvas*, esta tecnologia poderá aproveitar o seu crescimento para outras plataformas, disponibilizando também estes ambientes nas mesmas, pois tudo aquilo que o utilizador precisa para o acesso a estas representações, é um *browser* – e este está atualmente presente em qualquer dispositivo, seja esse móvel ou não.

Mas para o sucesso da aplicação há um ponto essencial que não pode ser esquecido: a interação. Durante muito tempo esta foi uma componente um pouco desprezada, mas atualmente a importância da interatividade com o utilizador é largamente reconhecida. Portanto, a investigação realizada nesta área, nomeadamente no que diz respeito às diferentes formas de aquisição de informação sobre um espaço e às vantagens e desvantagens dos formatos 2D e 3D na representação de espaços urbanos, irão servir de base para a concretização de uma interface simples, agradável e eficaz do ponto de vista do utilizador final.

De uma forma geral, apesar dos vários projetos relacionados encontrados, nenhum deles se relacionava diretamente com o objetivo do sistema desta dissertação. A ideia de representar um ambiente tridimensional e permitir a clientes utilizá-lo com diferentes objetivos, seja para a área da imobiliária (para venda ou arrendamento de espaços) ou para o contexto da promoção de um evento, pretende chegar ao grande público e oferecer-lhe uma forma mais apelativa, prática e interessante de procurar um local para arrendar, ou de conhecer o local do evento ao qual vai participar. Isto, beneficiando também os clientes, criadores destas representações, que poderão de forma simplificada alcançar um público mais abrangente para o que pretendam promover.

Capítulo 3

Ambientes Urbanos Virtuais em WebGL

Este capítulo tem como propósito a apresentação da solução pensada e planeada. Inicialmente, e de forma a contextualizar o seu surgimento, irá ser introduzido o produto aLIVE Places. De seguida serão explicitados os detalhes da solução a abordar, descrevendo os requisitos deste sistema assim como os seus casos de utilização.

3.1 aLIVE Places

O produto desenvolvido pela empresa 3Decide não consiste apenas numa representação tridimensional de um espaço que permite aos seus utilizadores uma experiência de conhecimento e exploração do mesmo de uma forma interativa e dinâmica, mas sim numa solução mais abrangente que aglomera a esta representação uma ligação a um sistema de informação. Sistema este que possibilita aos clientes, responsáveis pelo espaço, a gestão de toda a informação associada à sua representação, de uma forma simples, prática e em tempo real, necessitando para isso apenas de aceder à página web do *back-office*.

O produto foi criado para oferecer aos seus clientes uma solução moderna e completa para o desafio da comunicação de espaços e negócios. Depois do uso de plantas técnicas, plantas comerciais e de ambientes tridimensionais estáticos, o aLIVE Places pretende posicionar-se como o próximo passo evolutivo. Para alcançar estes objetivos, o produto foi construído tendo em foco vários componentes: a gestão da informação associada a edifícios ou espaços interiores a partir de uma plataforma amigável e *online*; a integração com diversos serviços *online* e redes sociais

(Google Maps, LinkedIn, Facebook,...); a portabilidade para dispositivos móveis e a qualidade e interatividade dum formato de comunicação de espaços universalmente compreendido - o 3D.

Este produto pode ser utilizado por todos os interessados em promover o seu negócio e os seus espaços, sejam estes imóveis para vender ou arrendar, centros de negócios ou de lazer, parques industriais, parques científicos, centros de congressos, museus, cinemas, teatros ou outros locais turísticos.

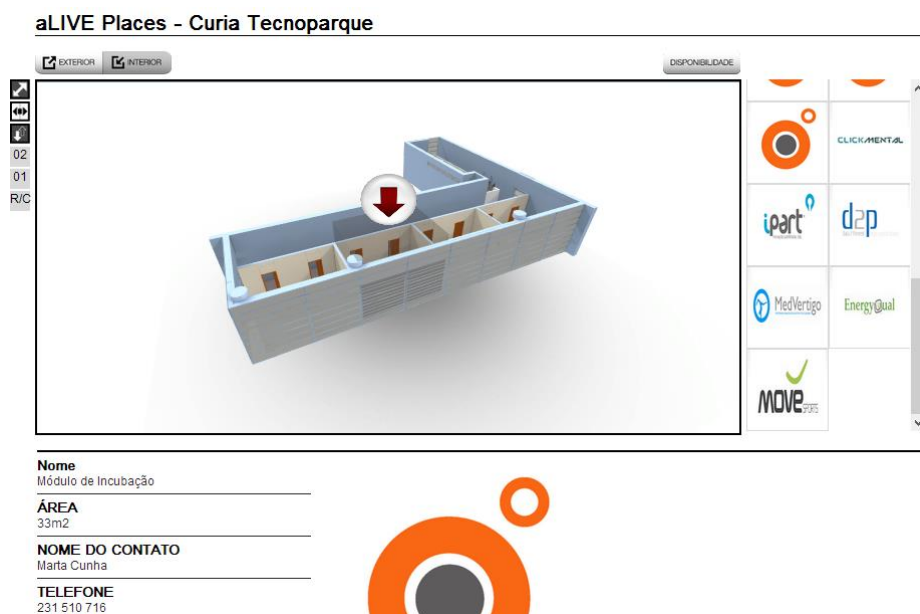


Figura 9 - Interface do produto aLIVE Places

3.2 Descrição da Solução

No contexto do produto previamente descrito, e no decorrer da sua divulgação a potenciais clientes, a empresa começou a deparar-se com uma nova problemática. Dado o carácter promocional deste tipo de aplicação, diferentes utilizadores, a partir de variados dispositivos, sistemas operativos e *browsers*, e com distintos níveis de experiência com *softwares* semelhantes, iriam ter acesso ao sistema. Assim, tornar-se-ia importante assegurar que em todos estes casos a aplicação deveria estar operacional e funcionalmente idêntica. Além disto, muitos destes utilizadores não estariam interessados em ter de instalar *software* adicional no seu computador ou dispositivo pessoal para poderem ter acesso à aplicação, reduzindo imediatamente o valor do produto.

Deste modo, surgiu o propósito desta dissertação, a migração da ideia base do produto aLIVE Places para um ambiente Web nativo, que seja capaz de alcançar um espectro de público mais abrangente.

Neste novo sistema deverá ser possível ao utilizador interagir com a cena 3D de variados modos – rotação, *zoom*, *panning*-, deverá também permitir a navegação pelos diferentes níveis do modelo – quarteirão, edifício e piso – e como consequência destas interações deverá ser apresentada, de forma integrada e a partir da ligação ao *back-office*, toda a informação associada ao espaço representado.

Existe ainda um outro objetivo associado a este sistema, que consiste na aplicação desta plataforma a um contexto de gestão de agenda de eventos. Esta implementação funcionará como uma nova camada de funcionalidades que será adicionada ao módulo base, já explicado em cima, e deverá consistir numa prototipagem que inclui informações sobre os eventos e as variadas atividades que os compõem, relacionando-os com os espaços onde estes decorrem. Este módulo deverá também incluir a integração da apresentação dos detalhes dos eventos à interação com a cena 3D.

Uma outra funcionalidade existente nesta nova camada será a apresentação de percursos associados a uma atividade, quer seja um percurso desde a entrada até ao espaço da atividade ou o caminho a percorrer desde esse local até ao sítio onde outra atividade decorrerá. Estes percursos deverão ser representados de forma interativa no modelo 3D.

3.3 Requisitos Funcionais e Casos de Utilização

Esta secção será dedicada à apresentação detalhada dos requisitos funcionais que o projeto deverá implementar no seu estado final. A acompanhar os requisitos irão ser apresentados também diagramas de casos de uso, tentando desta forma ilustrar a relação entre os requisitos da aplicação e as funcionalidades requeridas do ponto de vista do utilizador final.

3.3.1 Atores

Na aplicação a construir só existirá um tipo de ator, o **utilizador**, que terá acesso a todas as funcionalidades do sistema.

3.3.2 Componente da Cena 3D

Manipulação

- A cena 3D deverá sofrer rotação quando o utilizador executa um clique com deslocamento a partir do botão esquerdo do rato
- Deverá ocorrer um efeito de *zoom* à cena quando o utilizador interage a partir do *scroll* do rato

- A cena deverá sofrer um efeito de *panning* sempre que o utilizador realizar um *drag* utilizando o botão direito do rato

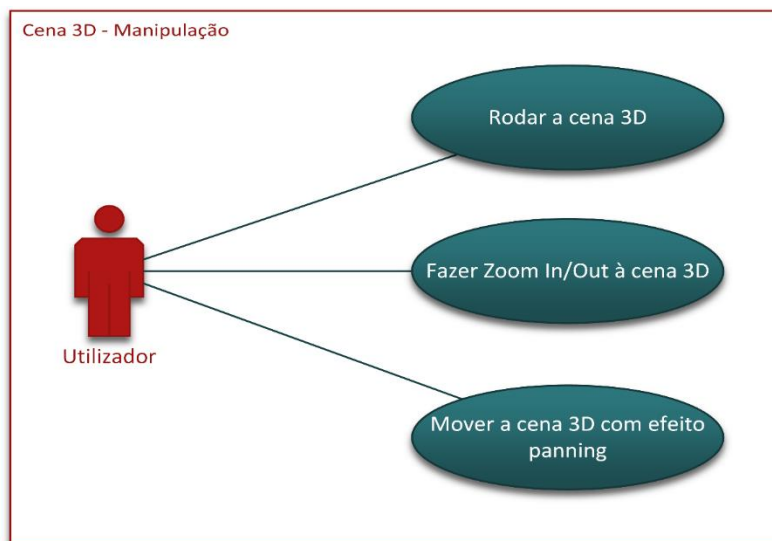


Figura 10 - Diagrama de casos de uso: Cena 3D - Manipulação

Na figura acima podemos ver um diagrama com os casos de uso a ter em conta para a construção das funcionalidades de manipulação da cena 3D.

Navegação

- Sempre que o utilizador posicionar o rato em cima de um espaço seleccionável, neste deverá ser visível um efeito distintivo
- Quando o utilizador clicar sobre um espaço, este deverá surgir com uma cor de seleção baseada na informação obtida do *back-office*
- No caso de o utilizador clicar fora de qualquer espaço na zona da cena 3D, deverá ser limpa qualquer seleção no modelo visível
- Assim que for realizado um segundo clique sobre um espaço selecionado, ou um duplo-clique sobre um não selecionado, deverá ser mostrado o respetivo espaço, quando possível. Ou seja, quando isto ocorrer sobre um edifício ou um piso, o modelo apresentado deverá alterar-se para mostrar a seleção com maior detalhe

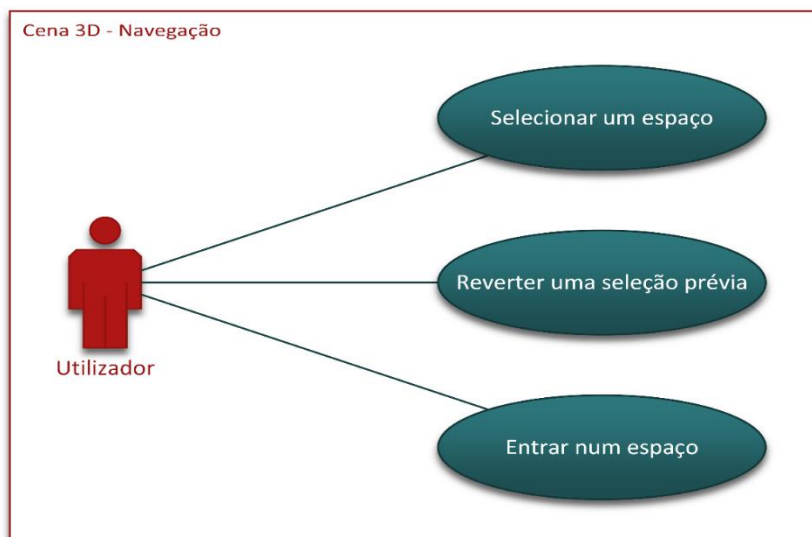


Figura 11 - Diagrama de casos de uso: Cena 3D - Navegação

Na Figura 11 é apresentado o diagrama de casos de utilização para a navegação na cena 3D da aplicação.

3.3.3 Componente de Informação de Espaços

- Deverá ser visível ao utilizador uma galeria de imagens referentes a cada um dos espaços seleccionáveis na cena 3D
- Sempre que o utilizador efetuar um clique sobre uma imagem, deverá ser visível na cena 3D o espaço associado a essa imagem selecionado
- Em todos os momentos durante a interação com a aplicação, deverá estar visível ao utilizador um pequeno painel informativo identificando qual o edifício, piso e sala ativos ou seleccionados
- Após o utilizador realizar um segundo clique sobre um espaço, aconteça este na cena 3D ou na imagem referente ao espaço, deverá surgir um painel *slider* com informação mais detalhada sobre o respetivo espaço
- Quando o utilizador executar um clique fora de qualquer espaço no painel da cena 3D, o *slider* informativo deverá ficar novamente oculto
- A qualquer momento, o utilizador pode também abrir ou fechar o painel de detalhes através de um clique sobre o mesmo

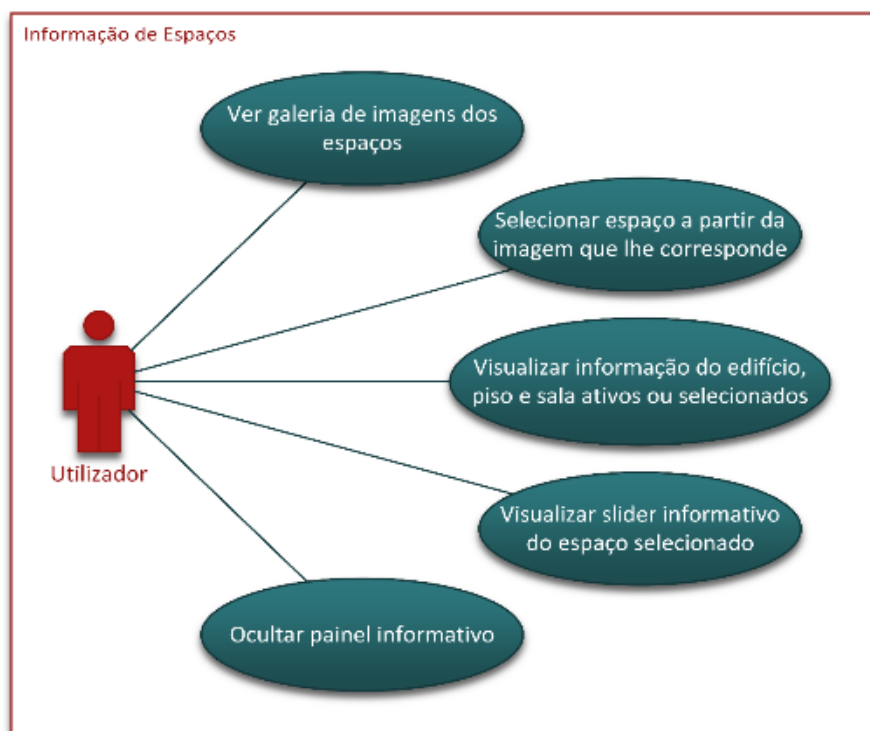


Figura 12 - Diagrama de casos de uso: Informação de Espaços

No diagrama de casos de utilização apresentado na Figura 12 é possível visualizar os casos de uso relativos ao componente de informação de espaços.

3.3.4 Componente de Gestão Eventos

Eventos

- O utilizador deverá ser capaz de visualizar um calendário junto com uma listagem dos eventos ainda não terminados
- Sempre que o utilizador clicar sobre um evento listado, os espaços onde este ocorrerá deverão ser selecionados na cena 3D com a cor associada a partir do *back-office*. Esta mesma cor deverá ser apresentada no elemento clicado representando a atual seleção
- Ao segundo clique do utilizador sobre um evento, o painel, além de apresentar a informação geral do mesmo, deverá abrir a listagem das atividades que o compõe
- Depois do utilizador «abrir» um evento, caso o piso atual não contenha nenhuma atividade desse evento, o andar visível na cena 3D deverá ser alterado de forma a apresentar um piso que possua atividades do evento selecionado
- Como consequência da abertura de um evento, deverá também ser visível no calendário uma seleção do período de tempo em que o evento decorre

- Quando o utilizador se encontra a visualizar a listagem de atividades de um evento, deverá ser possível voltar ao painel anterior, com a listagem de todos os eventos, a partir de um clique sobre o evento atual

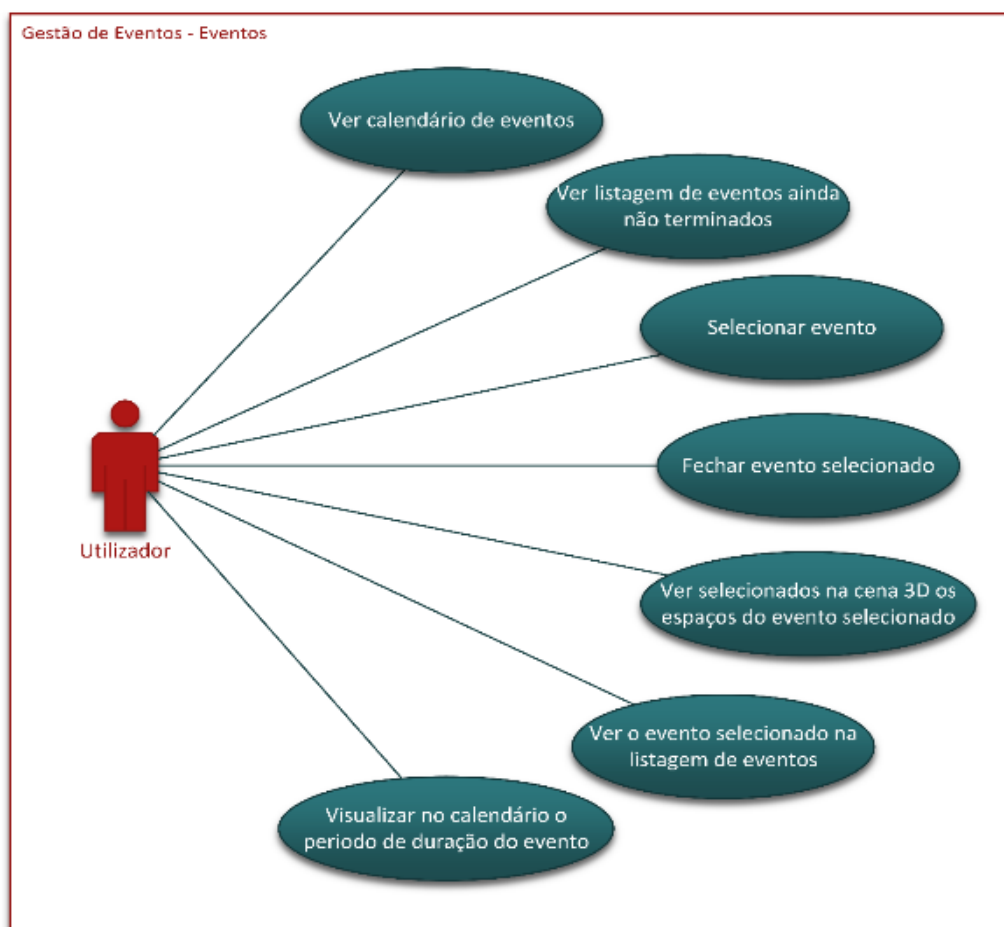


Figura 13 - Diagrama de casos de uso: Gestão de Eventos - Eventos

Na Figura 13 é possível consultar os casos de uso associados diretamente aos eventos no módulo de Gestão de Eventos.

Atividades

- No caso de o utilizador executar um clique sobre uma atividade listada, o espaço onde esta atividade decorre deverá aparecer selecionado na cena 3D
- Como consequência da seleção de uma atividade, a atual seleção deverá ser distinguível na listagem, junto com uma representação com a mesma cor da seleção na cena 3D
- Se o utilizador fizer um clique sobre o espaço vazio da cena, qualquer seleção de atividade deverá ser revertida

- Depois de um segundo clique sobre uma atividade listada, deverá surgir um painel *slider* contendo informação mais detalhada sobre a respetiva atividade
- Sempre que o utilizador seleccionar um dia do calendário deverá ser apresentada uma listagem com as atividades a decorrer nesse dia, caso existam

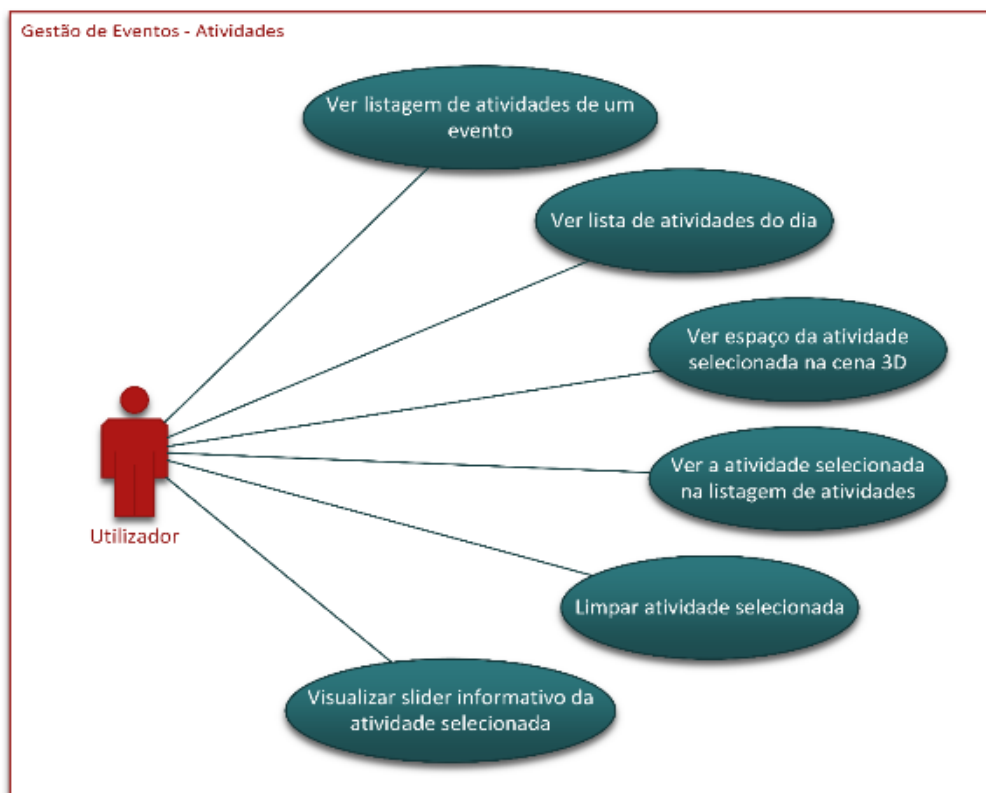


Figura 14 - Diagrama de casos de uso: Gestão de Eventos - Atividades

Na Figura 14 são representados diagramas de caso de uso relativos ao módulo Gestão de Eventos, mais especificamente os correspondentes à interação com o elemento 'Atividade' deste módulo.

Percursos

- Sempre que uma atividade estiver seleccionada, e existirem percursos com origem ou destino no espaço onde a mesma decorre, o utilizador deverá ter a possibilidade de seleccionar um percurso a partir de uma *combobox*
- Como consequência da seleção de um percurso deverá ser visível na cena 3D os espaços do percurso seleccionados com a cor definida para esse caminho no *back-office*

- Após a seleção de um percurso deverá estar disponível ao utilizador um botão de ‘Visualizar Percurso’ que ao ser pressionado deverá dar a origem a uma animação que percorre o caminho atualmente selecionado

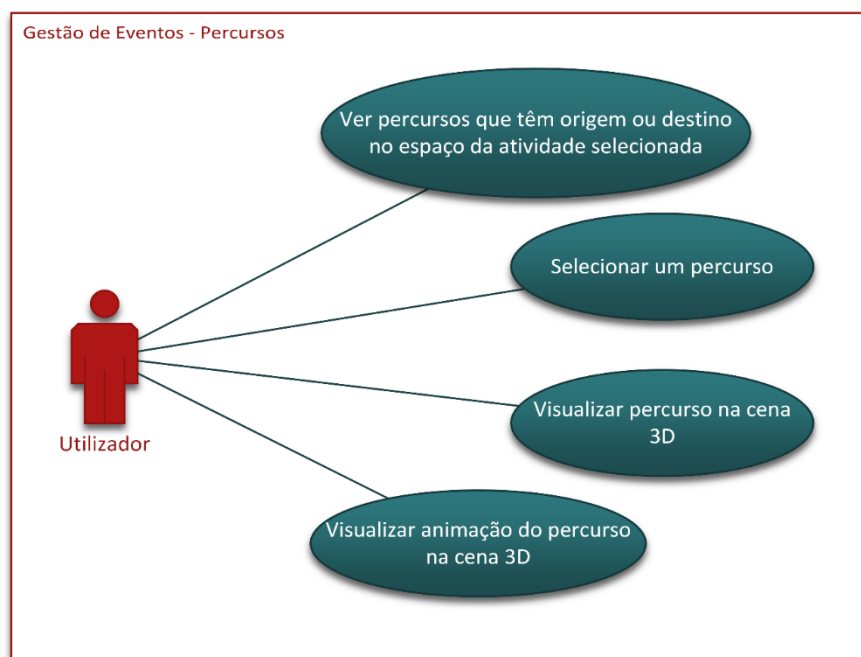


Figura 15 - Diagrama de casos de uso: Gestão de Eventos -

É possível ver na Figura 15 um diagrama com os casos de utilização que dizem respeito ao componente de percursos no corrente módulo de Gestão de Eventos.

3.4 Arquitetura da Solução

Como é possível ver na Figura 16 o sistema final será constituído por 2 elementos principais: o *back-office*, coincidente com o do produto aLIVE Places e onde são geridas todas as operações relativas à base de dados, e a aplicação Web que corresponde ao projeto a ser desenvolvido na íntegra.

A aplicação Web subdivide-se em alguns componentes. O módulo de cena 3D diz respeito à *renderização* e funcionalidades de interação com os modelos tridimensionais. O componente de informação, composto pela informação de espaços e pelos dados de gestão de eventos, é responsável por toda a apresentação e interação com os dados alfanuméricos que complementam a cena tridimensional.

De forma não tão visível ao utilizador final, existem ainda mais dois módulos. O primeiro, a API, é o elemento responsável pela gestão da interação entre a cena 3D e o componente da

informação, pois executa a comunicação necessária de forma aos dois módulos se encontrarem em sintonia. O último componente, é aquele que tem como responsabilidade tratar o carregamento de todos os dados a apresentar, digam estes respeito aos modelos a utilizar na cena 3D ou aos detalhes a apresentar na página web.

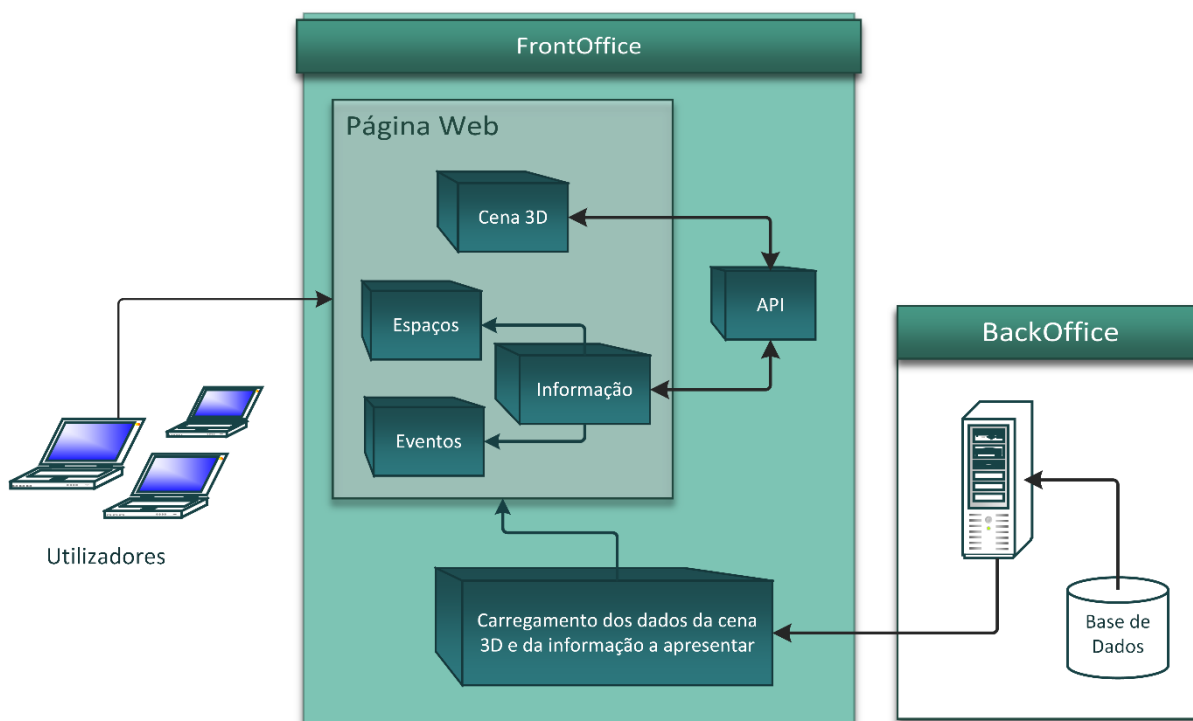


Figura 16 - Diagrama de Arquitetura do Sistema

3.5 Descrição da Interface

De maneira a implementar os módulos descritos na secção anterior, foi projetada uma interface capaz de encapsular os diferentes elementos necessários para satisfazer os requisitos do utilizador.

Com esse intuito, o foco principal foi o componente do ambiente 3D, sendo por isso o painel de maior dimensão no ecrã e mantendo-se sempre visível. Neste painel, de forma a complementar os modelos tridimensionais, está visível também uma pequena indicação do posicionamento atual do utilizador - o edifício, piso e sala presentemente ativos.

O espaço do ecrã foi partilhado com outro painel de fácil acesso e interação, onde está disponível informação básica e curta sobre os diferentes elementos a divulgar.

De forma a exibir dados informativos mais detalhados sobre esses mesmos elementos, foi ainda projetado um painel de detalhes, com carácter ocultável, para não tirar o foco da cena tridimensional.

De modo a ambientar o utilizador aos controlos necessários na utilização da aplicação, foi criado um pequeno painel de instruções que surge quando o utilizador inicializa a aplicação e que poderá ser novamente consultado a qualquer momento da sessão.

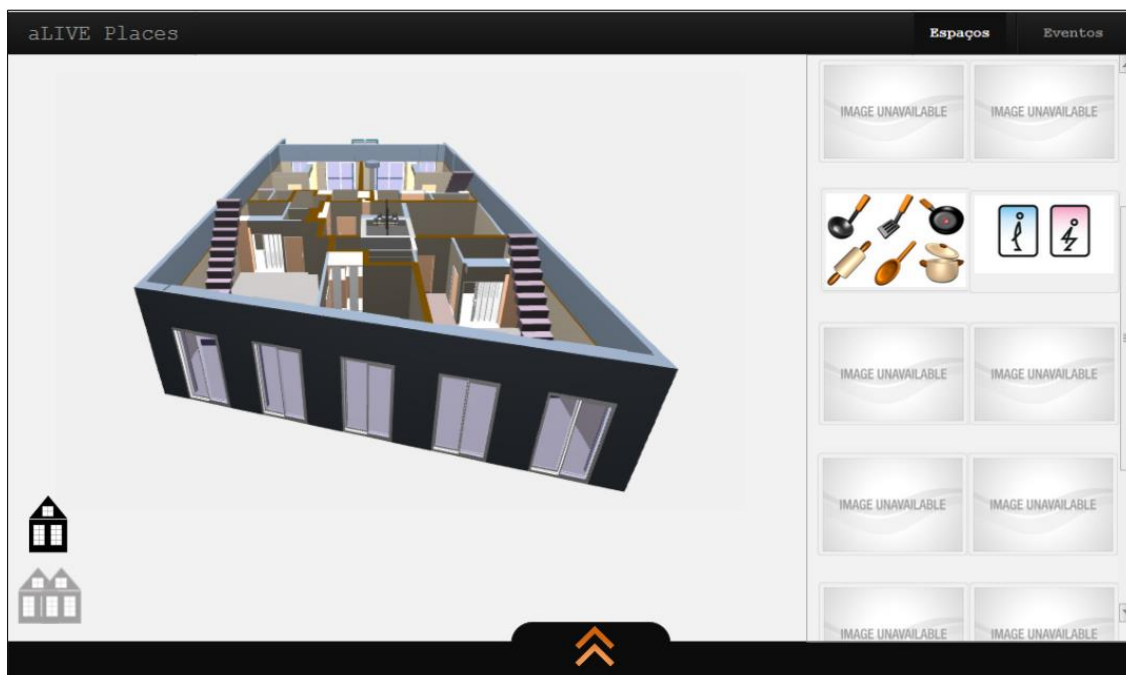


Figura 17 - Mockup da interface da página da aplicação

3.6 Resumo e Conclusões

A partir da informação contida neste capítulo é fácil concluir que a presente aplicação é constituída por uma componente muito forte de interação com o utilizador, sendo por isso os seus requisitos compostos em grande parte por necessidades relacionadas com a usabilidade, interatividade e correspondente *feedback* gráfico.

Neste capítulo podemos também deduzir que a implementação vai, numa primeira etapa, relacionar-se muito com o produto aLIVE Places. Contudo, irá ter numa segunda fase uma componente mais inovadora dedicada ao planeamento, estruturação e construção das funcionalidades apropriadas à divulgação de um âmbito ainda pouco explorado.

Capítulo 4

Implementação

Ao longo deste capítulo serão abordadas todas as fases do processo de implementação da solução para o problema descrito. Inicialmente será detalhada a construção de uma cena 3D, exemplificando com as metodologias e ferramentas utilizadas. De seguida será descrita a construção do módulo de informação de espaços e finalmente será apresentado o processo de implementação do módulo protótipo relativo à divulgação de eventos.

4.1 Ambiente 3D e Módulo de Informação de Espaços

Depois de realizado o estudo das tecnologias 3D disponíveis para a Web, e como já foi justificado antes, a tecnologia eleita para a construção do ambiente tridimensional foi o WebGL em conjunto com a biblioteca Three.js.

4.1.1 WebGL e Three.js

Uma das grandes vantagens que o WebGL oferece é obviamente a possibilidade de renderizar gráficos 3D, acelerados por *hardware*, diretamente no *browser*. No entanto, as suas vantagens não ficam por aqui: dada a sua base no OpenGL ES 2.0, a sua API é baseada em padrões de gráficos 3D familiares e largamente aceites pela comunidade de programadores. Tendo também em conta a sua integração com o *browser* e com o conteúdo HTML, isto permite-lhe fazer uso de funcionalidades destes componentes tais como a composição em camadas, interação com outros elementos HTML e o uso dos mecanismos padrão de controlo de eventos HTML. Outro ponto positivo é a sua compatibilidade com diversos *browsers* ou sistemas operativos, sejam estes em *desktops* ou dispositivos móveis.

IMPLEMENTAÇÃO

No entanto, o WebGL consiste numa API baixo-nível, o que tem como consequência que mesmo para programar coisas simples sejam necessárias muitas linhas de código. E é por isto que surge a Three.js. Esta consiste numa biblioteca que pretende tornar o desenvolvimento de ambientes tridimensionais no *browser* mais simples e rápido a partir da simples inclusão de um ficheiro Javascript. Com este objetivo em mente, esta API tem vindo a ser desenvolvida e continuamente incrementada desde Abril de 2010, e possui atualmente um grande conjunto de componentes que nos permitem criar todo um ambiente tridimensional, incluindo-se entre esses os referidos na Tabela 2.

Componente	Descrição/Exemplo
<i>Renderers</i>	WebGL, <canvas>, <svg>, ...
<i>Scenes</i>	É possível adicionar e remover objetos em tempo de execução
<i>Cameras</i>	Em perspetiva e ortogonais. Assim como vários controladores.
<i>Lights</i>	Ambiente, direcionais, de foco e de hemisfério
<i>Materials</i>	<i>Lambert</i> , <i>Phong</i> , entre outros - todos com texturas e <i>smooth-shading</i>
<i>Geometry</i>	Planos, cubos, esferas, texto 3D, etc. Também inclui alguns modificadores.
<i>Loaders</i>	Binários, de imagem, de JSON e de cenas (COLLADA, OBJ, entre outros)

Tabela 2 - Componentes da biblioteca Three.js

Estes são alguns dos elementos disponíveis na API Three.js, consistindo muitos destes também em componentes utilizados no desenvolvimento do presente sistema.

Como já referido anteriormente, as tecnologias WebGL e Three.js são acessíveis a partir da linguagem Javascript. Para este projeto o código foi inicialmente desenvolvido na linguagem CoffeeScript que foi posteriormente compilado para Javascript. Deste modo, os exemplos de código aqui apresentados serão sempre em CoffeeScript.

Coffeescript é uma linguagem de programação que adiciona algumas funcionalidades sintáticas ao Javascript, inspiradas por linguagens como Ruby, Python ou Haskell, de forma a melhorar a sua concisão e a facilitar a sua leitura. A tecnologia CoffeeScript permite criar programas com aproximadamente menos um terço das suas linhas em Javascript, compilando-se depois para essa linguagem sem consequências no seu desempenho.

4.1.2 Criação do elemento *Canvas* do HTML5

Como já explicado anteriormente, o WebGL assenta sobre um componente do HTML5, o elemento *canvas*. É dentro deste que todas as implementações a nível de ambiente 3D podem ser construídas. Assim, nesta secção vão ser abordados os vários passos necessários para a construção de um ambiente tridimensional com funcionalidades básicas.

4.1.2.1 Visualização da cena

Câmara, Luzes, Cena e *Renderer*

Uma cena, uma câmara, pelo menos uma luz e um *renderer* – estes são componentes essenciais de qualquer ambiente 3D, seja este em WebGL ou qualquer outra linguagem.

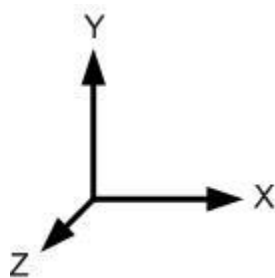


Figura 18 - Sistema de coordenadas seguido

Assim sendo, é por aí que a construção do presente ambiente será iniciado. Inicialmente, é necessário criar um elemento *Scene*, que irá funcionar como um contentor de todos os componentes do ambiente. O sistema de coordenadas seguido na Three.js, e portanto aquele que será sempre utilizado ao longo desta implementação, tem a orientação representada na Figura 18. Neste sistema o ecrã encontra-se sobre os eixos XY, e o utilizador encontra-se na direção positiva do eixo dos Z.

Seguidamente será criada uma câmara, e aqui surgem duas alternativas disponíveis pela Three.js: câmaras ortogonais e em perspetiva.

A partir de uma câmara ortogonal, o modelo exibido surge de forma em que todos os seus pontos são projetados ao longo de linhas paralelas às suas posições no ecrã, ou seja, nesta vista, um modelo 3D aparece plano ao contrário do que é observado no mundo real.

Utilizando uma câmara em perspetiva, os modelos apresentados surgem em projeção. Estas perspetivas correspondem a um efeito visual em que as linhas paralelas convergem num ponto de fuga único. Esta é a perspetiva equivalente à observação que o ser humano (ou uma câmara real) tem dos objetos no mundo.

IMPLEMENTAÇÃO

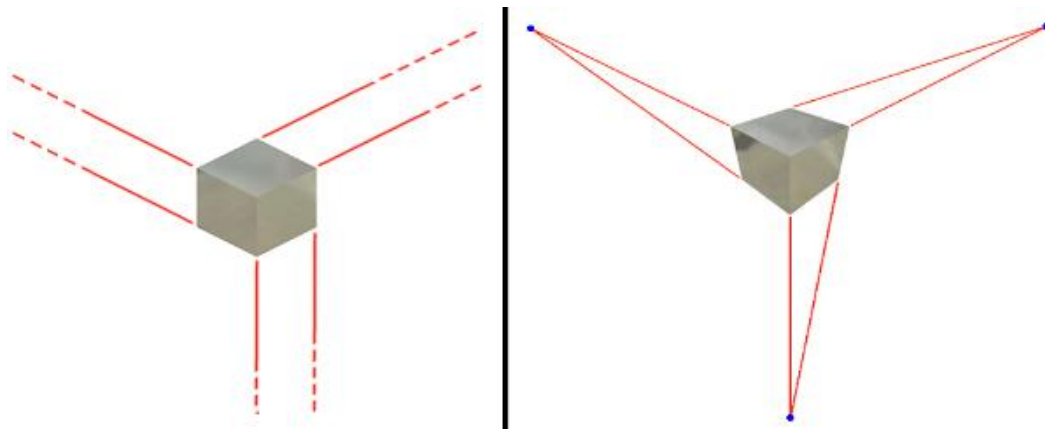


Figura 19 - Representação de uma vista ortogonal à esquerda, e em perspectiva à direita

Assim, considerando que o objetivo era uma vista realista e o mais natural e familiar possível para o utilizador, foi selecionada a câmara em perspectiva para o ambiente a criar, como pode ser observado na Listagem 1. Podemos também ver que, além da posição da câmara, foi definido um *target* inicial, correspondente à posição central de toda a cena. Mais à frente será explicada a sua utilidade.

```
1. init: =>
2.     # Scene
3.     scene = new THREE.Scene
4.
5.     # Camera
6.     camera = new THREE.PerspectiveCamera 70, width / height, 1, 10000
7.     camera.position.set(0,200,400)
8.     camera.target = scene.position
9.
10.    scene.add camera
11.
12.    # Lighting
13.    scene.add( new THREE.AmbientLight( 0x777777 ) )
14.
15.    pointLight2 = new THREE.PointLight 0x666666
16.    pointLight2.position.set(0,350,400)
17.    scene.add( pointLight2 )
18.
19.    # Renderer
20.    renderer = new THREE.WebGLRenderer( antialias: true, clearColor:
    0x000000 )
21.    renderer.setSize width, height
22.
23.    # Append rendered to html element
24.    document.getElementById(elementId).appendChild(renderer.domElement)
```

Listagem 1 - Inicialização de uma cena: elementos base [UIThree.coffee]

Após a criação de uma câmara, e para ser possível ver os objetos, é necessário inicializar algumas luzes. Neste caso, existe uma luz ambiente que permite iluminar de forma simples toda

IMPLEMENTAÇÃO

a cena. Mas se o ambiente contivesse apenas esta luz, os objetos visíveis iriam ter uma coloração plana (*flat*) onde não se distinguem arestas ou faces, portanto deverá adicionar-se outra(s) luz(es). No presente caso recorreu-se ao uso de um simples ponto de luz.

Por fim, criámos outro componente da biblioteca Three.js, o *renderer*, e definimos largura e altura do painel da cena a *renderizar*.

Objetos

Depois de preparado o ambiente, falta o essencial – os objetos. Há dois modos de inserir objetos na cena: (i) construí-los programaticamente a partir de sólidos básicos que a Three.js já disponibiliza; (ii) importar modelos a partir de formatos de ficheiros populares e facilmente exportáveis por diversos *softwares* de modelação 3D. Tendo em conta que o objetivo do presente sistema é apresentar modelos de ambientes urbanos, respetivamente conjuntos de edifícios ou estes por si só e os seus pisos constituintes, pretende-se pôr em prática a segunda abordagem.

Para fazer o carregamento de modelos 3D, a Three.js possui no seu repositório um conjunto de *loaders* que o desenvolvedor pode utilizar, tais como *ColladaLoader*, *BinaryLoader* ou *OBJLoader*, sendo que o *loader* incluído por *default* na biblioteca é o *JSONLoader*.

Browser:	Google Chrome 25.0.1364.152	
Formatos:	Memórias	Medição (k)
COLLADA	Private Memory	832,296
	Virtual Private Memory	847,180
JSON	Private Memory	702,260
	Virtual Private Memory	790,869
OBJ	Private Memory	757,872
	Virtual Private Memory	794,732

Tabela 3 - Medição de dispêndio de memória para diferentes formatos

Durante o desenvolvimento, foram testados os *loaders* para ficheiros JSON, COLLADA e OBJ de forma a analisar o desempenho e simplicidade de uso. No final, e de acordo com a Tabela 3, aquele que se mostrou a solução com menor gasto de memória, e um formato de fácil acesso foi o OBJ. Desta forma, podemos ver na Listagem 2 o carregamento dos modelos utilizando o *OBJMTLLoader*, que diz respeito a carregamento de modelos no formato .obj junto com materiais (definidos no ficheiro .mtl).

IMPLEMENTAÇÃO

```
25. loadFiles: =>
26.   @start()
27.
28.   main_key = null
29.
30.   # Loader initialization
31.   objLoader = new THREE.OBJMTLLoader
32.   objLoader.addEventListener( 'load', @addModelToScene )
33.
34.   for key of buildingsToLoad
35.     modelName = "models/"+jsonData.id+mobile_model+"/building_"
+jsonData.name+"/"+jsonData.name+"__"+main_key
36.
37.     objLoader.load modelName+modelsFormat , modelName+'.mtl'
38.     nextKey = main_key
39.     delete buildingsToLoad[key]
```

Listagem 2 - Função inicial de carregamento de modelos 3d [UIThree.coffee]

Depois de instanciado o *loader*, é necessário associar-lhe uma função *listener* para o evento 'load', evento despoletado quando o carregamento do ficheiro é terminado. Como o objetivo é carregar vários modelos, respetivos aos diversos edifícios e pisos, a função de *callback*, *addModelToScene*, deverá chamar-se recursivamente até todos os modelos terem sido carregados.

Ciclo de renderização

Agora que todos os elementos estão a postos, só é preciso mais uma coisa para ser possível visualizar o ambiente no *browser*: *renderizar* toda a cena de forma contínua. Com a Three.js isso deverá ser feito da seguinte forma:

```
40. start: =>
41.   @init()
42.   @render()
43.
44. render: =>
45.   requestAnimationFrame @render
46.
47.   camera.lookAt(camera.target)
48.
49.   renderer.render scene, camera
```

Listagem 3 - Função cíclica de renderização da cena [UIThree.coffee]

Neste momento já será possível ver no *browser* uma simples cena tridimensional. Pode-se também ver neste ciclo o uso da função *lookAt* na câmara, de forma a orientar a vista sempre para o ponto central da cena (definido anteriormente).

4.1.2.2 Interação com a cena

Após a criação de uma cena visível diretamente no *browser* é necessário permitir que o utilizador possa interagir com a mesma. Nesta secção irão ser descritas as possibilidades de interação com o ambiente previamente construído.

Interação com o conjunto

A forma mais básica de interatividade consiste na interação com o conjunto de objetos, ou seja, ações como rodar a cena, aproximar ou afastar – efeito de *zoom* -, ou ainda efeito de *panning*. O *panning* consiste num movimento que aparenta ser um deslocamento bidimensional de toda a cena em relação à vista do utilizador.

Para implementar este tipo de interatividade é possível utilizar mais um recurso fornecido pela biblioteca Three.js, os controladores de câmara. Neste campo a API permite mais algumas escolhas, pois possui alguma variedade de opções. Existem *controls* para vistas de primeira pessoa – *FirstPersonControls.js*, para ambientes de voo - *FlyControls.js*, orbitais – *OrbitControls.js*, entre outros.

O presente projeto baseia-se na apresentação de maquetes simplistas de ambientes urbanos, ou seja, quer-se permitir ao utilizador girar o espaço visível e consultá-lo a partir de diferentes perspetivas e/ou proximidades, no entanto nunca deve ser retirado de foco o edifício. Por essa razão foi escolhido o *OrbitControls*, que permite um movimento de câmara sempre relativo a um objeto sobre o qual se orbita, no caso, consistirá no espaço que se queira apresentar ao utilizador.

```

50. init: =>
51.     # (...)
52.     # Controls
53.     controls = new THREE.OrbitControls( camera, renderer.domElement )
54.     controls.minDistance = buildingScalingFactor*25
55.     controls.maxDistance = buildingScalingFactor*70
56.     controls.maxPolarAngle = Math.PI/2
57.
58. render: =>
59.     # (...)
60.     # Update controls
61.     controls.update()

```

Listagem 4 - Inicialização e configuração de controlador de câmara [UIThree.coffee]

Fazer uso de algum destes controladores é tão simples quanto incluir o ficheiro Javascript junto com os outros scripts, na página de *index.html* por exemplo. Contudo, é possível fazer um pouco mais. Existe um conjunto de parâmetros que são possíveis de personalizar e que podem ser úteis ao nosso projeto, como por exemplo a habilitação/deshabilitação de algum dos modos de interação (rotação, *zoom* e *panning*). No presente sistema foi configurada a distância mínima e máxima do efeito de *zoom*, e também o ângulo polar máximo de rotação, isto foi aplicado de

forma a que o utilizador não possa fazer uma rotação para zonas onde o ponto de observação se encontre abaixo do nível do chão.

Interação com objetos singulares

Um modo mais complexo de interatividade diz respeito à interação com objetos singulares que fazem parte da cena. Estas interações oferecem um vasto leque de possibilidades e de ações que podem estar disponíveis ao utilizador.

Para oferecer interação com diferentes pisos e diferentes salas desses mesmos pisos, todos os modelos importados têm já incorporadas estruturas referidas como ‘picking boxes’. Estas estruturas consistem numa espécie de caixas que circundam todas as salas e pisos com os quais deverá ser possível interagir.

Encontrar os objetos intersetados no ambiente 3D a partir das posições bidimensionais do cursor do rato no ecrã é uma tarefa que requer um algoritmo especializado. Para esse procedimento foi utilizado o algoritmo *Ray Casting*. Este algoritmo consiste no uso de testes de interseção entre raios e superfícies que permitem resolver uma variedade de problemas em computação gráfica. O termo *Ray Casting* surgiu pela primeira vez em 1982, num artigo de Scott Roth[34] que descrevia um método para *renderizar* modelos CSG (Constructive Solid Geometry).

```

62. recordMouseMove: (event) =>
63.   event.preventDefault
64.
65.   mouse.x = ( (event.clientX ) / width ) * 2 - 1
66.   mouse.y = - ( (event.clientY ) / height ) * 2 + 1
67.
68. recordMouseClicked: (event) =>
69.   event.preventDefault
70.   mouse.clicked = true

```

Listagem 5 - Funções de tratamento de eventos de movimento e clique de rato [UIThree.coffee]

No caso do presente projeto quer-se resolver o problema de interseção dum raio, lançado a partir da posição do cursor, com os diferentes objetos da cena visível. Para implementar essa funcionalidade irá ser utilizado outro recurso da Three.js, a classe *Raycaster*. Mas antes de fazer uso do algoritmo, é necessário armazenar, após cada mudança de posição ou após cada clique, a posição (x,y) do cursor do rato em relação ao painel *canvas*, conforme se pode ver na Listagem 5.

IMPLEMENTAÇÃO

Depois, no ciclo de *renderização*, é criado o componente *Raycaster*, com base na posição atual da câmara e na posição atual do cursor, e executado o cálculo do *array* de interseções com uma lista de objetos conforme a Listagem 6. A lista usada continha apenas as já referidas *picking-boxes* de forma a não atrasar o processamento calculando interseções com todos os objetos da cena. Após este procedimento é possível tratar os objetos resultantes, podendo criar efeitos de *onHover* ou *onClick*, com mudança de cor ou opacidade do material do objeto, ou mesmo mudar o modelo visível, de vista edifício para vista piso por exemplo.

```
71. render: =>
72.   # (...)
73.   # Mouse position vector
74.   vector = new THREE.Vector3( mouse.x, mouse.y, 1 )
75.
76.   projector.unprojectVector( vector, camera )
77.   raycaster = new THREE.Raycaster( camera.position, vector.sub(
    camera.position ).normalize() )
78.   intersects = raycaster.intersectObjects( collidableList )
79.
80.   # Check intersection
81.   if intersects.length > 0
82.     # (...)
```

Listagem 6 – Cálculo da lista de objetos intersetados pelo cursor [UIThree.coffee]

4.1.3 Criação da Página Web

O segundo componente desta aplicação que complementa a experiência do ambiente 3D é a página Web envolvente e as informações que esta disponibiliza ao utilizador. Todo este componente é alimentado a nível de dados pelo *back-office*. Quanto à sua visualização, foi criada tendo em atenção a consistência da página a partir de pontos de acesso diferentes, nomeadamente diferentes *browsers* e sistemas operativos.

4.1.3.1 Estruturação

Antes de ser iniciada a construção dos elementos informativos da página web foi necessário fazer uma análise da estrutura deste componente.

IMPLEMENTAÇÃO

Dados

```
1.  {  "id":"6",
2.    "language":"pt_pt",
3.    "name":"UPTEC",
4.    "description":"A brand new project...",
5.    "rooms":{
6.      "661":{
7.        "area_square_meters":"37.7404",
8.        "custom_fields":[
9.          {
10.             "label":"Nome",
11.             "value":"Sala de Co-working"
12.           }
13.        ],
14.        "ancestors":[
15.          "660"
16.        ]
17.      },
18.    "floors":{
19.      "660":{
20.        "floor_number":"0",
21.        "descendants":[
22.          "661",
23.          "662"
24.        ],
25.        "ancestors":[
26.          "659"
27.        ]
28.      }
29.    }
30.  }
```

Listagem 7 - Amostra da estrutura JSON obtida do *back-office*

Em relação à informação a disponibilizar, os dados lidos pela aplicação estão no formato JSON. O ficheiro transmitido contém todas as informações relevantes à apresentação de conteúdos ao utilizador.

Na Listagem 7 é possível analisar os elementos informativos mais importantes do resultado JSON de um projeto, nomeadamente dados do projeto, do edifício, pisos, salas e relações de descendência entre os mesmos.

Visual

Quanto à componente visual da página, foram criadas *mockups* de forma a estruturar os elementos essenciais a apresentar. Estas permitiram uma avaliação e melhoramento da interface antes de se iniciar o processo de desenvolvimento.

Complementar ao painel da cena 3D, existirá um painel lateral e um painel inferior deslizante. Quando o utilizador se encontrar a visualizar informação de espaços, o painel

IMPLEMENTAÇÃO

lateral deverá conter o conjunto de imagens dos espaços, já no painel inferior deverá ser visível os detalhes do espaço atualmente selecionado.

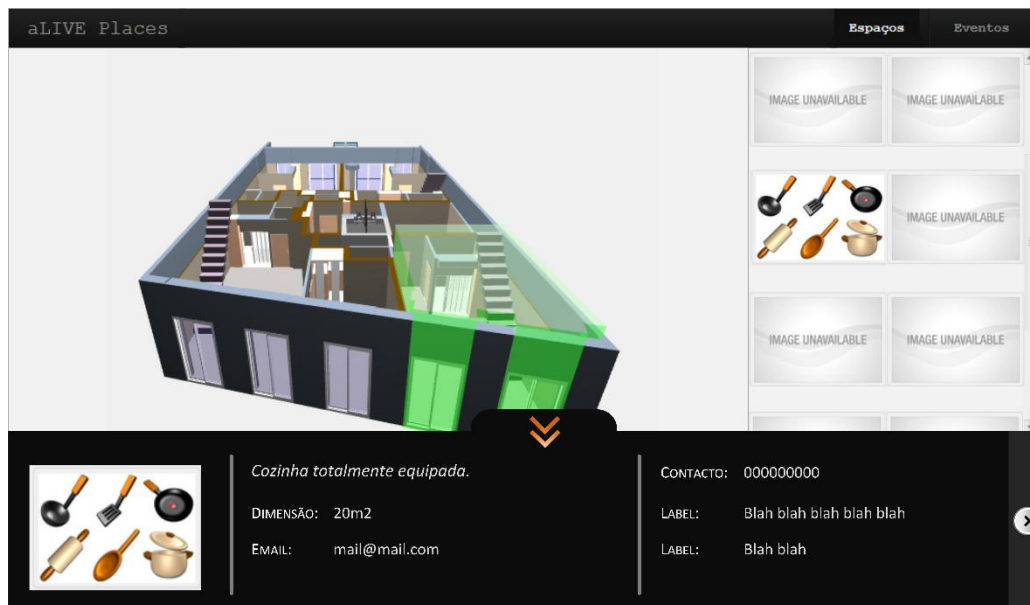


Figura 20 - *Mockup* da página web em relação ao módulo de informação de espaços

4.1.3.2 Implementação

De forma a implementar e apresentar a estrutura de dados representada em cima, em conformidade com uma interface amigável e prática, foram utilizados alguns recursos externos.

Com o intuito de uniformizar e acelerar o desenvolvimento para a web e também assegurar a consistência do conteúdo visualizado nas mais variadas resoluções de ecrã foi utilizada a *framework* Bootstrap. Esta *framework* oferece um conjunto de ferramentas para produzir aplicações web (HTML, CSS e Javascript *toolkit*) de forma mais célere. Contém também suporte a *design* responsivo, isto é, o conteúdo gráfico das páginas adapta-se de forma dinâmica de acordo com as características do dispositivo utilizado, seja este um PC, *tablet* ou *smartphone*.

Tendo em conta o carácter estático da linguagem HTML, ou seja a sua não apropriação quando o objetivo é criar vistas dinâmicas, foi utilizada também a tecnologia AngularJS de forma a estender o vocabulário HTML. A AngularJS consiste numa *framework* Javascript que permite aos seus utilizadores construir as suas aplicações com uma arquitetura MVC (*model-view-controller*), com o intuito de tornar tanto o desenvolvimento como o teste das aplicações mais simples.

Obtenção de Dados

Para obter os dados a utilizar na apresentação da página, foi necessário comunicar com o *back-office* do produto aLIVE Places utilizando uma das funcionalidades da AngularJS que nos permite criar serviços personalizados, e também um serviço que a mesma já disponibiliza, o *\$http*, que possibilita efetuar comunicações com servidores HTTP remotos.

```

1. app = angular.module('myApp.services', [])
2.
3. app.factory('jsonConfigService', ($http)->
4.   jsonConfigService =
5.     async: ()->
6.       promise = $http.get('json_file_'+ getUrlParameter("id")+'.js')
7.       .then( (response)->
8.         return response.data
9.       )
10. # (...)

```

Listagem 8 - Obtenção de dados de servidores externos [services.coffee]

A chamada a este serviço deverá ser realizada de forma assíncrona a partir do nosso *controller*. De seguida, efetua-se o carregamento dos restantes painéis, nomeadamente informativos e da cena 3D com as informações obtidas.

Apresentação da Página

Para realizar a apresentação das informações auxiliares ao ambiente tridimensional nos diferentes painéis da página web foi dado uso às diretivas da *framework* AngularJS. A biblioteca lê o código HTML que contém atributos *tag* adicionais, de seguida obedece às diretivas nesses atributos e associa secções da página a um modelo representado por simples variáveis Javascript. Desta forma, sempre que no decorrer do uso do presente sistema forem alteradas determinadas variáveis associadas ao *controller*, o conteúdo HTML irá ser atualizado.

```

11. <div class="span3" id="side-menu" >
12.   <ul class='thumbnails' id='roomList' >
13.     <li class='span6' ng:repeat="room in roomsOfSelectedFloor">
14.       <div class="thumbnail" id="room_{{room.id}}">
15.         </img>
16.       </div>
17.     </li>
18.   </ul>
19. </div>

```

Listagem 9 - Implementação de menu lateral com Bootstrap e AngularJS [index.html]

IMPLEMENTAÇÃO

Na amostra de código representada na Listagem 9 é possível ver o uso de diretivas da biblioteca da AngularJS, nomeadamente a *ng:repeat*, que é responsável por percorrer ciclicamente uma lista de imagens de forma a criar a listagem visível, e também a *ng-src*, que associa de modo dinâmico um endereço de *source* da imagem a apresentar. Foi também usado o recurso ‘thumbnails’ da Bootstrap para apresentação de imagens com dimensões dinâmicas adaptadas às dimensões do ecrã, e classes ‘span’ da mesma biblioteca para adaptar igualmente as larguras dos diferentes painéis.

4.1.4 Interação com espaços

De volta ao painel do ambiente tridimensional, foi sentida a necessidade de enriquecer a experiência do utilizador. Para isso foi implementada uma animação que deverá, a cada seleção de uma sala, transportar de forma linear o utilizador para uma vista centrada no espaço correspondente.

Para alcançar este objetivo foi utilizada uma nova biblioteca, a Tween.js. Um *tween* consiste na interpolação entre dois valores, da origem até ao destino. Ou seja, a biblioteca irá simplesmente gerar os valores intermédios necessários, mas com um ponto importante, o modo como estes são gerados. Isto poderá ser configurado com o método de *ease*, e existem várias alternativas desde interpolações lineares, cúbicas, sinusoidais, entre muitas outras, ainda com características de *ease in* e/ou *out* (início e/ou fim mais lento).

Para aplicar uma animação à câmara da cena primeiro começou por se calcular uma caixa delimitadora do objeto selecionado, a partir da função *computeBoundingBox()*, e de seguida encontrou-se o ponto central (apenas em XZ, dado que o objetivo era manter a altura) - *centerPosition*. Após isto é calculada a posição final em que a câmara se deverá localizar, mantendo a distância relativa ao ponto de observação atual - *finalPosition*.

Com estes dados já é possível criar um *tween* que anima a posição da câmara, de modo linear e durante 1000ms, por forma a finalizar na posição final definida.

```
1. tween = new TWEEN.Tween(camera.position)
2.   .to(finalPosition, 1000)
3.   .easing(TWEEN.Easing.Linear.None)
4.   .onUpdate(()-> console.log 'tween update;')
5.   .start();
6.
7. tween2 = new TWEEN.Tween(camera.target)
8.   .to(centerPosition,1000)
9.   .easing(TWEEN.Easing.Linear.None)
10.  .onComplete(()-> console.log 'tween 2 completed;')
11.  .start();
```

Listagem 10 - Criação de animações de câmara quando da seleção de um espaço

IMPLEMENTAÇÃO

Na Listagem 10 é possível ver também um segundo *tween* a ser criado e inicializado. Isto acontece devido a uma funcionalidade já mencionada anteriormente, o *target* da câmara. No ciclo de *renderização* foi colocada uma linha que chama a função 'lookAt' da câmara para o seu *target*. Isto faz com que a câmara se oriente para este ponto. Quando se seleciona uma nova sala, é desejado que a câmara passe a estar orientada para o respetivo espaço. Para que esta mudança de *target* não cause uma alteração drástica na orientação da câmara, foi criado um segundo *tween* que modifica o *target* de modo igualmente linear, para que o utilizador nem se aperceba desta alteração e a animação se mantenha apenas relativa à posição e não à orientação da câmara.

4.2 API - Interação entre cena e conteúdos

A interligação entre o módulo da cena 3D e o módulo da página e conteúdos informativos, foi criada graças a uma nova classe responsável pela gestão de todas as ações e comportamentos consequentes.

O que se encontrava inicialmente planeado seria a utilização da API do produto aLIVE Places, seguindo a mesma lógica funcional e adaptando apenas algumas características conforme o necessário. No entanto, após alguma análise e estudo da mesma, verificou-se que esta possuía elos e ligações em excesso, e que a quantidade de alterações a efetuar para adaptar a API a este sistema ia ser significativa. Em consequência disso justificou-se a criação de uma API própria para esta aplicação, mais limpa e apropriada às características do desenvolvimento. Após discussão com a equipa, optou-se então por esta via.

Foi então construída a classe *Behaviours*, e o método *handler*. Este funciona como o intermediário entre todas as ações e suas reações. Todos os eventos acionados, sejam estes resultantes da interação do utilizador com a aplicação ou do próprio sistema, chamam esta função para que sejam desencadeadas as consequentes reações desse acontecimento.

```
1. class Behaviours
2.
3.   handler: (event, id, origin) =>
4.     switch event
5.       when EventCategory.ON_LOADED
6.         then @uiThreeLoaded()
7.
8.       when EventCategory.ON_FLOOR_LOAD
9.         then @floorLoad id
10.
11.      when EventCategory.ON_ROOM_SELECTED
12.        then @roomSelected id, origin
13. # (...)
```

Listagem 11 - Classe *Behaviours* e função de gestão de comportamentos

IMPLEMENTAÇÃO

Dando um exemplo prático, cada vez que o utilizador seleciona uma sala, no ciclo de *renderização*, já descrito anteriormente, é chamado o método *handler* tendo como argumentos o evento *ON_ROOM_SELECTED*, o *id* do espaço e como origem da chamada a cena 3D. Este método reencaminhará a responsabilidade para a função *roomSelected* onde irão ser posteriormente executadas as seguintes ações:

- Animação da câmara
- Atualização de número de piso e sala visíveis
- *Scroll* para a posição da imagem dessa sala
- Atualização do painel de detalhes
- Ativar efeito de seleção no espaço atual e limpeza do anterior caso exista

4.3 Módulo de Gestão de Eventos

De modo a prototipar o processo desenvolvido até ao momento, foi projetada a implementação de um novo módulo relacionado com a gestão de eventos associados a um espaço.

Este módulo surge com o intuito de fazer chegar o presente projeto a um outro nível, de forma a colmatar a lacuna existente em espaços dedicados à promoção de acontecimentos, sejam estes de carácter cultural, académico ou profissional - a sua divulgação e comunicação de eventos e atividades ao público.

4.3.1 Estruturação

Dado que o presente módulo não estava previamente implementado no produto aLIVE Places, a estruturação do mesmo requereu uma análise mais aprofundada dos requisitos definidos, de forma a preparar um desenvolvimento bem-sucedido e que vá de encontro aos resultados esperados no que diz respeito à compatibilidade com o produto aLIVE Places.

4.3.1.1 Dados

Em relação ao modelo de dados a utilizar, depois de estudadas as necessidades dos potenciais clientes na área dos eventos, foi representada a estrutura de base de dados a implementar de forma complementar à já existente no *back-office*.

No diagrama da Figura 21 podemos ver a estrutura de dados planeada, ligando-se às tabelas *3dp_project* e *3dp_node* que pertencem ao produto aLIVE Places e dizem respeito às tabelas de um projeto e de um espaço, respetivamente. Foi então criada uma tabela evento, que contém os

IMPLEMENTAÇÃO

dados mais gerais de um acontecimento, associada a um projeto. Cada evento é constituído por atividades, apesar destas não necessitarem de um evento para existir, que estão associadas a um espaço e possuem um campo que diz respeito ao seu tipo (Workshop, palestra, cinema, musical, etc). Além disto temos ainda outra componente - os percursos. Estes consistem num conjunto ordenado de espaços, e estão diretamente associados aos espaços correspondentes à origem e destino. Esta última estrutura irá ser explicada com mais detalhe mais à frente.

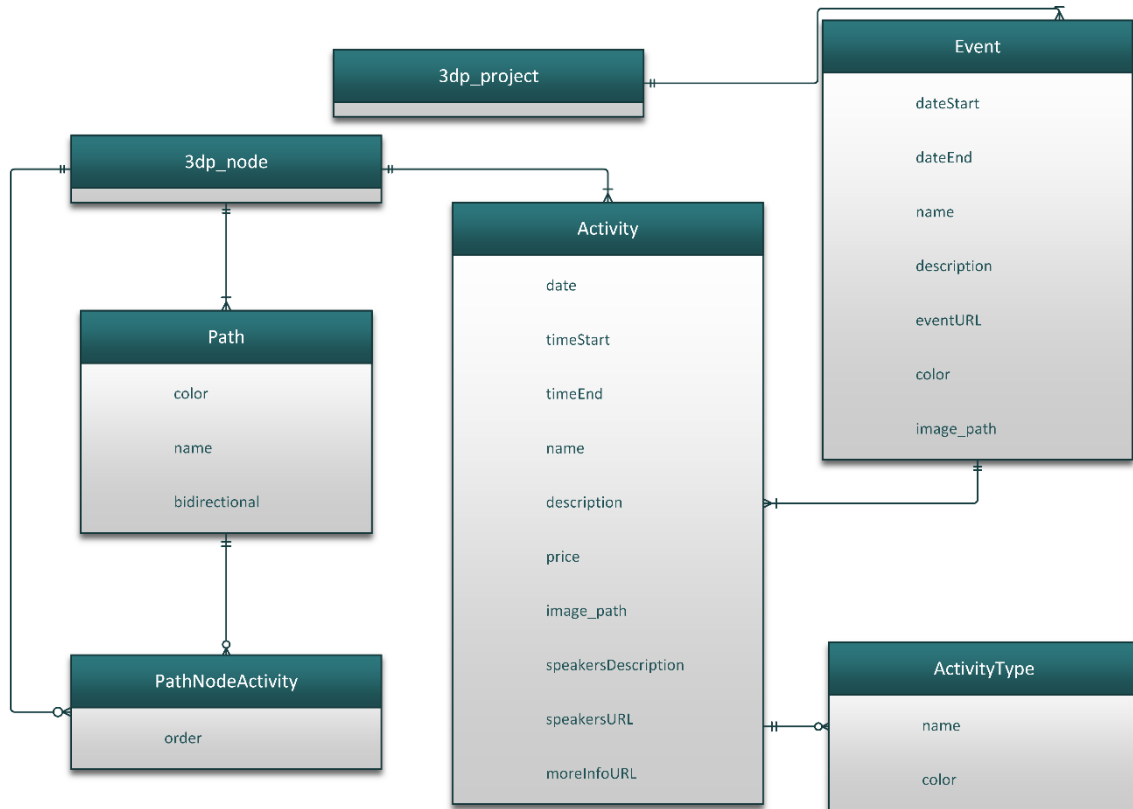


Figura 21 - Diagrama de base de dados relativo ao módulo de gestão de eventos

Quanto ao ficheiro JSON resultante da estrutura de base de dados, foi tida em atenção a consistência com o modelo de dados utilizado para o módulo anterior. Com isso em mente, pode ver-se na Listagem 12 um exemplo do formato dos dados a ler pela aplicação.

Além dos dados gerais de cada tabela, podem ser vistos campos que representam a relação entre tabelas, nomeadamente a lista dos *ids* das atividades de cada evento e também os *ids* de evento, tipo de atividade e espaço de uma atividade. A redundância na existência dos *ids* das atividades no evento e vice-versa prende-se com a eficiência na consulta destes dados a partir de acessos com diferente origem. Caso existisse uma atividade não associada a um evento e a relação entre estas tabelas se encontrasse apenas do lado dos eventos, iria ser gerada uma pesquisa

IMPLEMENTAÇÃO

prescindível com gastos de processamento dispensáveis. Por fim, é visível também uma lista ordenada composta pelos *ids* dos espaços que compõem cada percurso.

```
14. "events":{
15.   21":{
16.     "dateStart":"2013-06-10 10:00:00",
17.     "dateEnd":"2013-06-16 23:30:00",
18.     "name":"OPO Art Week",
19.     "activity_list":[
20.       "55",
21.       "56"
22.     ]
23.   }, "activities":{
24.     "56":{
25.       "timeStart":"2013-06-10 16:00:00",
26.       "timeEnd":"2013-06-10 18:00:00",
27.       "name":"Fotografia de Paisagem - O Douro entre Pontes",
28.       "price":"15",
29.       "speakersDescription":"Tiago Garcia",
30.       "event_id":"21",
31.       "activityType_id":"7",
32.       "node_id":"693"
33.     }
34.   }, "paths":{
35.     "2": {
36.       "color":"#fdca50",
37.       "name":"Rota B",
38.       "nodes": [
39.         "696",
40.         "693",
41.         "688",
42.         "686",
43.         "682",
44.         "678",
45.         "697"
46.       ]
47.     } }
```

Listagem 12 - Estrutura JSON criada para o módulo de gestão de eventos

4.3.1.2 Visual

Foi também analisada a melhor abordagem a tomar na implementação visual dos elementos informativos dos eventos, de forma a estes entrarem em conformidade com a restante interface visual do módulo de informação de espaços e de toda a cena 3D.

IMPLEMENTAÇÃO



Figura 22 - Mockup da página web em relação ao módulo de gestão de eventos

Na *mockup* da Figura 22 é possível observar que foi criado um separador que irá substituir no painel direito a listagem das imagens dos espaços por conteúdo informativo relativo a eventos. Decidiu-se optar pelo separador de forma a não ocupar mais espaço do ecrã com elementos informativos adicionais, mantendo como componente principal a cena 3D.

Para apresentar os detalhes da atividade selecionada de momento faz-se uso do painel inferior deslizante mantendo de igual forma a consistência com o módulo de informação de espaços.

4.3.2 Painéis informativos de Eventos

Depois da conceção, quanto ao seu carácter visual e informativo, do módulo de eventos, passou-se à fase de implementação dos seus componentes. Aqui temos também, além do painel deslizante de detalhes, já presente no módulo de informação de espaços, um painel lateral que neste módulo é constituído por um calendário e por uma listagem de acontecimentos.

4.3.2.1 Calendário

Este é um dos componentes principais do módulo, e permite ao utilizador uma interação simples com as atividades e eventos a partir de uma representação temporal familiar.

IMPLEMENTAÇÃO

Aquando da construção da interface para o módulo de eventos, o calendário surgiu como o elemento mais óbvio a incluir dado ser um elemento comum e de simples usabilidade. Para a sua implementação foi utilizado o componente *DatePicker* da popular biblioteca *jQuery*.

A partir do calendário o utilizador poderá navegar temporalmente e aceder de forma imediata à listagem de atividades a decorrer no dia selecionado, caso existam. É também possível ter um *feedback* visual dos dias em que decorre determinado evento quando o utilizador se encontra a navegar no mesmo.

4.3.2.2 Listagem de Dados

O componente da listagem de dados vai possibilitar ao utilizador uma interação rápida e simples sobre os diferentes elementos, apresentando-lhe imediatamente uma descrição curta das atividades e/ou eventos sobre os quais este poderá ter interesse em saber mais detalhes.

Na Listagem 13 pode observar-se o uso de diversas diretivas da AngularJS para apresentação de uma parte do painel de eventos. Na linha 1 e 2 tem-se o uso das diretivas *ng-show* e *ng-hide* para trocar a mensagem visível em situação de existir, ou não, eventos a decorrer. Após isso é utilizada novamente a diretiva *ng:repeat* para criar a listagem de eventos de forma dinâmica.

```
1. <span ng-show="(events|eventsToCome).length"> Eventos: </span>
2. <span ng-hide="(events|eventsToCome).length"> Não existem eventos a
   decorrer </span>
3. <tr ng:repeat="event in events|eventsToCome" id="event_{{event.id}}">
4.   <td class="span12">
5.     <div> <strong>{{ event['name'] }}</strong> </div>
6.     <div style="margin-left:10px;">
7.       <small>De {{ event['dateStart'] | date:'dd/MM' }}
8.         até {{ event['dateEnd'] | date:'dd/MM' }} </small>
9.     </div>
10.   </td>
11. </tr>
```

Listagem 13 - Exemplo de uso de diretivas da AngularJS [index.html]

É possível ver também o uso de outra funcionalidade da biblioteca, os *filters*. Estes permitem, como o nome indica, filtrar campos de informação e podem ser criados pelo programador. Um exemplo de um *filter* personalizado é o ‘eventsToCome’ que é responsável por apenas devolver os eventos que ainda não terminaram. Contudo, existem outros filtros predefinidos como o ‘date’ que aceita como argumento o formato da data a retornar.

Como já explicado, esta listagem comporta-se de forma dinâmica, o que significa que o utilizador poderá ver diferentes tipos de listagem. Um desses foi o já mencionado na secção anterior limitando as atividades visíveis ao dia selecionado, outra opção é visualizar as atividades

a decorrer no espaço selecionado após o utilizador interagir com a cena 3D. Quando não existem atividades a apresentar como consequência da ação que o utilizador realizou, surge uma mensagem de aviso, e o utilizador poderá optar por navegar noutra listagem, a de eventos ainda não terminados. Se o utilizador entrar num determinado evento, temos a última opção, a listagem de atividades apenas do evento que foi selecionado.

4.3.2.3 *Slider de Detalhes*

O painel deslizante de detalhes oferece ao utilizador um conjunto de informações mais particulares sobre uma atividade selecionada. Este *slider* apresenta um carácter ocultável, de forma a não se tornar um obstáculo na experiência de usabilidade da aplicação, funcionando apenas como uma extensão opcional de informação.

No módulo de informação de espaços, explicado em 4.1, este painel continha dados da sala selecionada tal como foto, *website*, descrição e área, não sendo estes dados fixos, pois o utilizador pode, no *back-office*, customizar os campos que deseja colocar, assim como a sua *label*.

Neste módulo a informação a ser apresentada diz respeito à atividade selecionada, contendo campos como o nome, o horário, preço, descrição e mesmo o(s) orador(es) com a possibilidade de associar também hiperligações externas tanto aos oradores como à descrição.

4.3.3 Percursos

O componente de percursos prende-se com a visualização de um trajeto composto por um número variável de espaços que fazem a ligação entre um local de origem e outro de destino.

A implementação desta funcionalidade foi decidida com o intuito de apresentar ao utilizador informação de difícil transmissão quando o espaço do evento a divulgar é desconhecido ao seu público. Especificamente, informação relativa a como se dirigir dum sala onde o utilizador frequentará alguma atividade até qualquer outro ponto-chave, seja este um ponto de refeição, de saída ou mesmo a sala de uma outra atividade.

Desta forma, a partir do *back-office* poderão ser criados diferentes tipos de percursos. O utilizador poderá visualizar uma rota selecionando uma opção dum *combobox* existente no elemento da atividade listado.

4.3.3.1 Visualização

Após a seleção de um percurso, este ficará visível na cena 3D de forma a oferecer ao utilizador uma descrição visual e realista do caminho a percorrer para ir do ponto de origem ao ponto de destino.

IMPLEMENTAÇÃO

Para criar esta visualização foi criada uma função que a partir do *id* do percurso faz uma pesquisa das *picking-boxes* dos espaços desse percurso. Depois coloca esses espaços com a cor do percurso definida no *back-office*, de forma ao utilizador distinguir claramente o percurso dos restantes espaços.

Para realçar as salas de origem e destino foram também colocados *billboards*, isto é, simples imagens indicativas centradas acima do espaço a que se referem, e que se encontram sempre voltadas para a câmara. A implementação destes *billboards* é realizada através da criação de um plano que tem como textura a imagem desejada e posteriormente, no ciclo de *renderização*, a utilização da função *lookAt*, já referida anteriormente, para manter o *billboard* voltado para o utilizador.

4.3.3.2 Animação

Além da visualização do percurso de forma estática, outra funcionalidade disponível ao utilizador é a de ver a animação de um percurso. Quando o utilizador pressiona o botão com este objetivo, ser-lhe-á visível uma animação que percorre ordenadamente os espaços do trajeto selecionado.

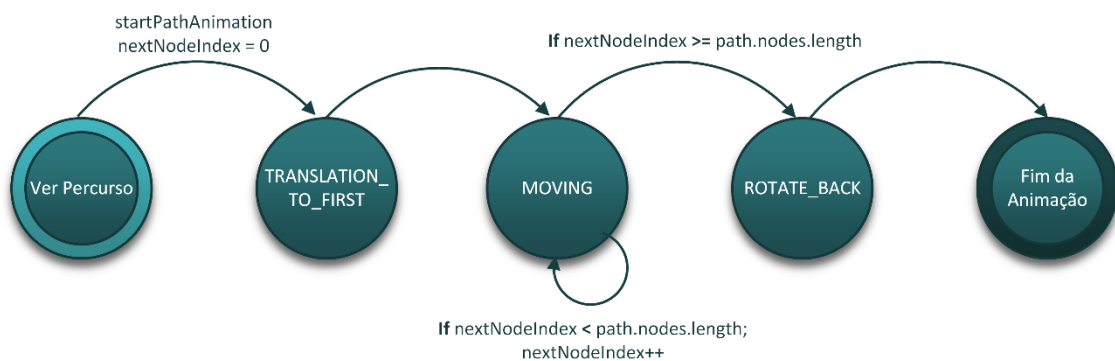


Figura 23 - Máquina de estados da animação do percurso

Para criar a animação foi desenvolvida uma implementação por estados. Quando o utilizador pressiona o botão de ‘Ver Percurso’ é chamado o método *startPathAnimation*. Esta função irá ser chamada repetidamente com um conjunto de parâmetros até ser atingido o estado final.

Inicialmente, é feita uma primeira animação desde a posição atual da câmara até a um posicionamento sobre o espaço inicial. De forma a seguidamente ser chamada a mesma função com o estado MOVING, foi configurada uma função *listener* para o evento *onComplete* do *tween* da animação. O estado de movimento entre os vários espaços é chamado de forma cíclica acrescentando o índice do próximo nó, até que este já não exista.

IMPLEMENTAÇÃO

Quando isso se verificar passa-se então ao estado final. Aqui é criado um *tween* que irá fazer a animação entre a posição atual e a posição da câmara inicial previamente guardada.

Tendo conta que os *tweens* da animação foram criados de forma semelhante aos explicados no ponto 4.1.4, em que 2 *tweens* simultâneos são iniciados - um para o movimento da câmara e outro que efetua a alteração do ponto de observação - foi evidenciado um problema na animação quando esta se encontrava no estado de movimento entre espaços. O problema consistia numa alteração da orientação da câmara em 180° cada vez que esta passava imediatamente acima do centro de um espaço.

Após investigação sobre a origem do problema, foi encontrado um antigo *issue* da biblioteca Three.js onde era explicada a razão deste comportamento. Tentando representá-lo na vida real, imagine uma pessoa em pé a olhar diretamente para o chão e uma caixa á sua frente. Tenha em atenção que neste exemplo a câmara é a cabeça da pessoa, e o seu vetor de *Up* (existente nas câmaras da Three.js) está a sair pelo topo da sua cabeça. Assim, quando a pessoa passa o local imediatamente acima da caixa, para continuar a olhar para esta, o seu vetor de *Up* passa a estar a apontar para baixo. De forma a evitar que o vetor ‘cima’ esteja orientado para baixo, é necessário realizar-se uma rotação de 180°.

Depois de compreendido o problema, a solução encontrada foi, durante o tempo da animação, alterar no ciclo de *renderização*, o vetor de *Up* da câmara para não se verificar o comportamento explicado. A solução mostrou-se funcional, no entanto podem ocorrer saltos de câmara quando se volta a configurar o vetor de *Up* com os valores padrão no final da animação.

4.4 Resumo e Conclusões

O processo de desenvolvimento, numa etapa inicial, baseou-se numa metodologia experimental, funcionando muito à base de tentativa-erro à medida que se adquiriam mais conhecimentos e à vontade no uso das tecnologias selecionadas. Especialmente em relação à biblioteca Three.js, que se trata duma biblioteca em atualização constante e sem muita documentação oficial – no entanto com muita não oficial -, o desenvolvimento foi apoiado em grande parte por respostas a dúvidas em *sites* apropriados ou mesmo na secção oficial de *issues* do projeto da biblioteca no *GitHub*.

Numa segunda fase, a implementação da componente mais informativa, junto com a página Web e a API, consistiu numa etapa menos experimental e mais fluída à medida que se incrementava o leque de funcionalidades. No estágio inicial do protótipo foi dedicado algum tempo a uma boa estruturação e planeamento do protótipo.

IMPLEMENTAÇÃO

Já na fase final de todo o projeto voltou-se a um desenvolvimento mais experimental, nomeadamente em relação à construção de animações e correção de problemas de câmara que surgiram associados às mesmas.

Após todas estas fases experimentais e todas as melhorias e correções de problemas foi possível obter um projeto final que vai de encontro aos requisitos e casos de utilização definidos no Capítulo 3

Capítulo 5

Análise e Avaliação de Resultados

De forma a analisar e a avaliar os resultados do projeto desenvolvido, neste capítulo será apresentada a interface final da aplicação assim como os resultados obtidos tanto dos testes de performance como dos testes de usabilidade realizados a um grupo de utilizadores.

5.1 Interfaces

As interfaces da aplicação foram construídas de acordo com as *mockups* criadas e apresentadas. Todavia, algumas alterações foram surgindo após pequenos testes de usabilidade com a equipa, adequando pequenos detalhes à medida que o desenvolvimento evoluía. A certo momento do desenvolvimento surgiu também um novo *template* para o produto aLIVE Places, o qual foi tido em conta de forma a manter consistência no que diz respeito aos elementos comuns entre os 2 projetos. Desta forma a interface final resultante é a seguinte:

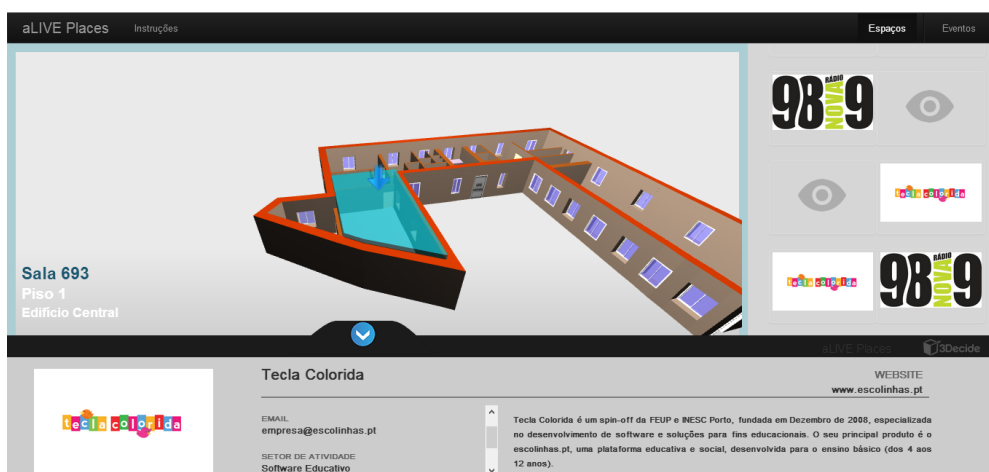


Figura 24 - Interface final relativa à cena 3D conjuntamente com a informação de espaços

ANÁLISE E AVALIAÇÃO DE RESULTADOS

Na Figura 24, é possível visualizar uma imagem da interface composta por uma barra de menus em cima, pela cena 3D, no caso apresentando um piso de um edifício com uma sala selecionada e associado à cena (no canto inferior esquerdo) uma pequena caixa de informação relativa à localização atual. Já do lado direito está o painel de imagens relativas a cada uma das salas do piso atual, e em baixo o painel de detalhes da sala atualmente selecionada.

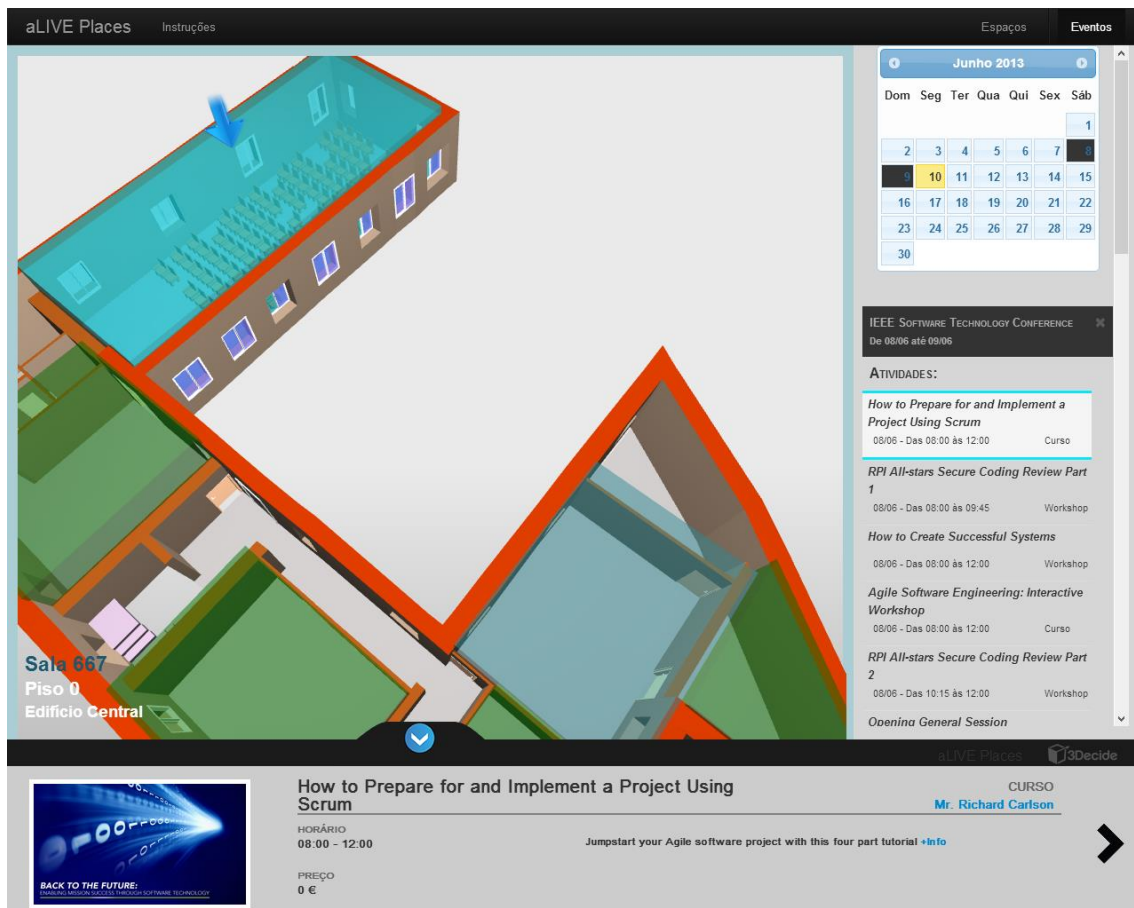


Figura 25 - Interface final relativa à cena 3D conjuntamente com a informação de eventos e atividades

Na Figura 25 podemos ver selecionado na barra de menus o módulo de eventos. Em consequência disso temos no painel da direita o calendário, onde se pode ver um evento selecionado, representado a preto no calendário e com o elemento descritivo preto antes da listagem das suas atividades. Na cena 3D é visível outro piso, novamente com uma sala selecionada, e com outras salas onde decorrem atividades do evento ativo a verde. No painel inferior estão apresentados os detalhes da atividade selecionada.

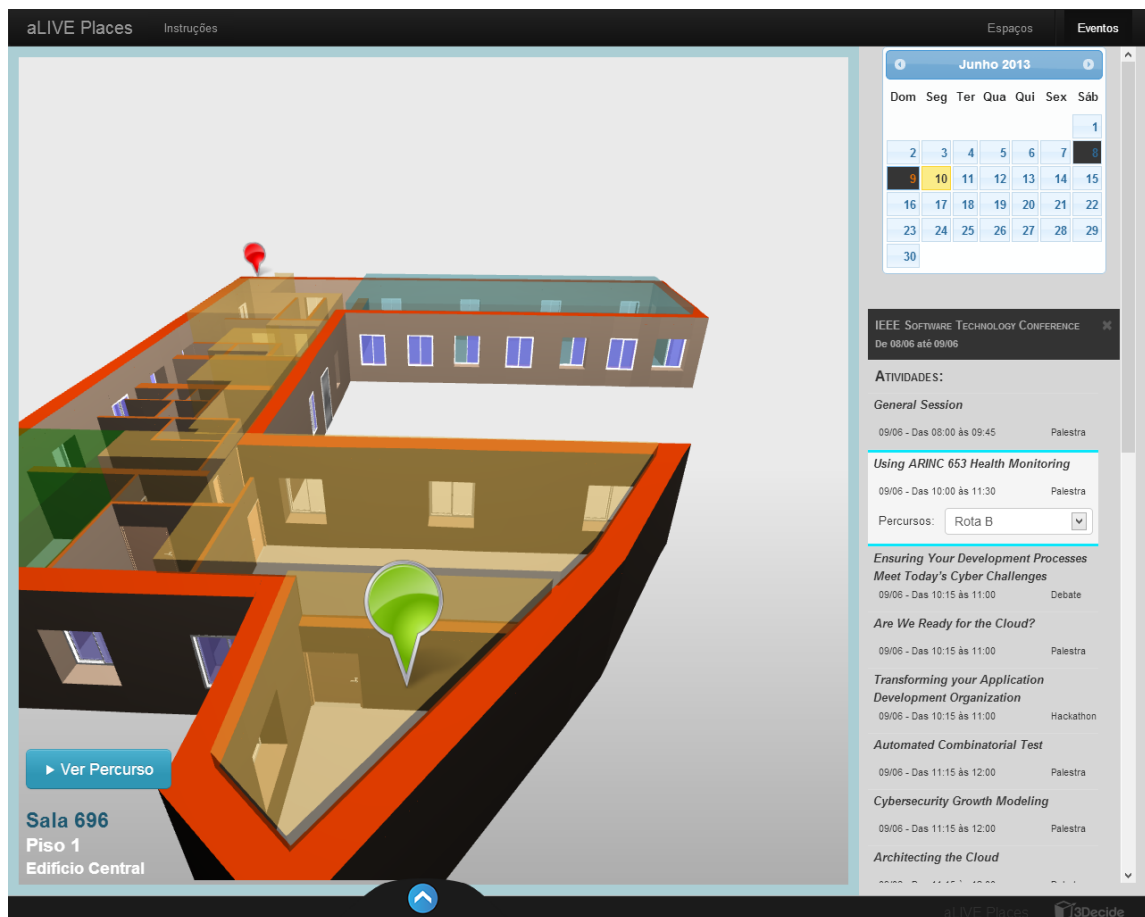


Figura 26 - Interface final relativa à cena 3D conjuntamente com percursos da atividade selecionada

Por último, temos na Figura 26 o mesmo módulo de eventos mas desta vez com um percurso selecionado a partir do elemento descritivo de uma atividade. Pode-se ver os indicadores do ponto de origem - a verde-, e de destino - a vermelho-, sobre as respetivas salas e junto à caixa de informações no canto inferior esquerdo um botão para iniciar a animação do caminho.

5.2 Testes

Ao longo do processo de desenvolvimento foi continuamente testada a aplicação quanto à sua compatibilidade com sistemas operativos e *browsers*. Na fase final, e de forma a avaliar a qualidade do sistema construído, foram realizados testes a 11 pessoas analisando a usabilidade da aplicação assim como a opinião dos utilizadores quanto à sua experiência de utilização.

5.2.1 Testes de compatibilidade

Dado que durante a fase de desenvolvimento do sistema foram utilizados métodos e bibliotecas direcionadas para a construção de páginas web consistentes na diversidade de *browsers* e resoluções, o processo de teste da aplicação em diversos sistemas operativos e *browsers* não levantou problemas.

Dessa forma a aplicação foi testada e encontra-se funcional nas últimas versões dos sistemas operativos Windows, Mac OS e Ubuntu e também nas versões mais recentes dos *browsers* Google Chrome, Mozilla Firefox, Opera e Safari. Na Tabela 4 estão representados os requisitos essenciais quanto à versão do *browser* para o funcionamento do presente sistema.

Browsers	Versão mínima requerida	Notas
Firefox	4.0	Pode ser necessário colocar a variável "webgl.force-enabled" com o valor 'true' para o funcionamento do WebGL
Chrome/ Chromium	8.0	O WebGL pode ser ativado no Chrome para Android a partir da versão 25. Já no Chrome para iOS o WebGL ainda não está disponível devido a restrições da plataforma.
Safari	5.1	O WebGL pode ser ativado a partir da opção "Enable WebGL" no menu Develop
Opera	12.0	Para ativar o WebGL deverá definir-se, na secção User Prefs, as variáveis "Enable Hardware Acceleration" e "Enable WebGL" com o valor 1
Internet Explorer	11.0	Anunciado o suporte a WebGL na próxima versão do IE.

Tabela 4 - Versões mínimas de browser para o funcionamento da aplicação

A aplicação foi também pensada para ser acedida a partir de dispositivos móveis, tal como *tablets*, no entanto não foram realizados testes nestas plataformas devido a não existir nenhum *tablet* disponível para o efeito. De mencionar que em iPads, ainda não é possível visualizar a página da aplicação dado ao acesso à tecnologia WebGL estar atualmente bloqueado nestes dispositivos. Já em *tablets* com o sistema operativo Android é possível aceder a este sistema, pelo menos a partir da versão 25 ou superior do *browser* Chrome para Android. Contudo, como foi referido, não foi possível realizar testes em nenhuma plataforma com estas características portanto a operabilidade da aplicação não é assegurada.

5.2.2 Testes de Usabilidade

Nesta secção serão descritos os procedimentos tomados, a metodologia seguida e posteriormente os resultados obtidos e a sua avaliação.

5.2.2.1 Descrição dos procedimentos

Para a realização dos testes de usabilidade foi utilizada a metodologia ‘Field Observation’. Esta consiste numa avaliação experimental que observa o utilizador a interagir com o sistema, num contexto natural, deixando-o à vontade. A pessoa tem plena consciência de que está a ser observada e os seus passos e comportamentos estão a ser registados, avaliados e testados.

Para o presente teste foram avaliadas 11 pessoas com idades compreendidas entre os 22 e os 26 anos. Nenhuma das pessoas tinha conhecimento da aplicação ou do espaço apresentado mas todas tinham experiência e à vontade na navegação na Internet. Foi-lhes pedido que realizassem um conjunto de tarefas e que testassem a simplicidade com que a conseguiam efetuar. Enquanto isso, todas as suas ações foram registadas para posterior análise.

A métrica utilizada para a avaliação foi a seguinte:

1. Número de cliques que cada utilizador dá até atingir o objetivo;
2. Tempo aproximado que o utilizador demora a atingir o objetivo;
3. Quantidade de vezes que o utilizador erra: por cada 5 cliques desviados do objetivo que o utilizador der é considerado um erro e o utilizador é colocado no passo que tentava encontrar, sendo-lhe pedido que continue a pesquisa a partir desse ponto.

Com o objetivo de abranger a generalidade das funcionalidades e componentes da aplicação foi criada uma lista de 7 tarefas a realizar pelos utilizadores. A lista é a seguinte:

Nº	Tarefa
1	Obter o número de telefone da empresa CulturePrint
2	Descobrir o piso em que se encontram as salas da empresa Tecla Colorida. Ver o número da sala que corresponde às escadas no piso descoberto;
3	Descobrir em que dias ocorre o evento IEEE Software Technology Conference;
4	Ver que atividades decorrem no dia 11/06/2013. Ver a que evento(s) pertencem;
5	Descobrir o preço da atividade Fotografia de Paisagem - O Douro entre Pontes, inserida no evento OPO Art Week;
6	Obter o link do orador da palestra intitulada Implementing Security in the Cloud, que decorre no Auditório (sala 667) do rés-do-chão;

- 7 Encontrar a palestra que decorre no dia 09/06/2013 das 10h às 11h30. Ver a animação do percurso ‘Rota B’;

Tabela 5 - Tarefas realizadas pelos utilizadores no teste de usabilidade

Após a avaliação experimental da usabilidade, foi pedido aos utilizadores que respondessem a um questionário que abordava nas suas perguntas duas componentes: a aquisição de informação sobre o espaço apresentado durante a realização das tarefas e a opinião pessoal dos utilizadores quanto à simplicidade de uso, qualidade da experiência e utilidade do sistema. O questionário era constituído, na primeira parte, por 6 perguntas de escolha múltipla e na segunda por 9 perguntas de escolha múltipla e uma de resposta aberta.

5.2.2.2 Análise e avaliação das Observações

Logo no decurso das avaliações foram facilmente observados pontos críticos na realização de determinadas tarefas que foram confirmados durante a análise final após todos os testes.

O primeiro entrave verificou-se aquando da realização da tarefa número 2. Esta tarefa consiste primeiro em localizar as salas de uma empresa e encontrar o número do piso e numa segunda parte localizar as escadas do piso e ver o número de sala. O caminho mais eficaz era composto por 3 passos apresentados na Tabela 6.

Nº	Descrição do passo
1	Secção Espaços (e localizar-se em vista de edifício – já por defeito no início da aplicação)
2	Seleção da imagem da empresa – Ver nº do piso
3	Selecionar escadas – Ver número da sala

Tabela 6 - Percorso mais eficaz para realizar a tarefa nº2

	Tarefa 2		Passos			Tempo
			-1-	-2-	-3-	
Utilizadores	A01	Cliques	1	1	3	50s
		Erros				
	A02	Cliques	1	1	2	25s
		Erros				
	A03	Cliques	1	1	5	1m40s
		Erros			x	
	A04	Cliques	1	1	5	1m30s
		Erros				

ANÁLISE E AVALIAÇÃO DE RESULTADOS

	Erros				x
A05	Cliques	1	1	3	45s
	Erros				
A06	Cliques	1	2	5	1m55s
	Erros				x
A07	Cliques	1	2	2	45s
	Erros				
B01	Cliques	1	3	4	1m
	Erros				
B02	Cliques	1	1	1	25s
	Erros				
B03	Cliques	1	1	5	1m
	Erros				x
C01	Cliques	5	1	2	50s
	Erros	x			
Média		1,4	1,4	3,4	1min
Desvio Padrão		1,1	0,6	1,4	28segs
Erros		1	0	4	

Tabela 7 - Registo do teste de usabilidade à tarefa nº 2

Na **Tabela 7** estão representados os dados observados no decorrer do teste de usabilidade a todos os utilizadores, sendo que as letras iniciais do *id* do utilizador são meramente indicativas da sessão de testes. No cabeçalho temos o número da tarefa, a listagem dos passos e o tempo aproximado. À esquerda temos os códigos dos utilizadores, e para cada um deles os dados registados, especificamente quanto ao número de cliques por passo, os erros dados e em que passo ocorreram, junto com o tempo aproximado de realização de toda a tarefa. Em baixo temos ainda a média de cliques dados e do tempo despendido assim como o número de erros por passo.

Como podemos visualizar na tabela, nesta tarefa, alguns utilizadores, nomeadamente o utilizador A03, A06 e B03 encontraram dificuldades no 3º passo tendo precisado de ser auxiliados após se afastarem 5 cliques do passo correto. A partir do processo de observação, e tendo em conta que na segunda parte desta tarefa os utilizadores tinham de navegar apenas pela cena 3D de forma a localizar as escadas, conclui-se que alguns deles apresentaram inicialmente algumas dificuldades em utilizar os diferentes comandos do rato (botão esquerdo, direito e scroll) para controlar o ambiente. Esta conclusão foi sustentada ao observar precisamente o oposto, utilizadores com grande à vontade na interação com o ambiente tridimensional, respetivamente os utilizadores A07 e B02, que tinham como característica distintiva o facto de ambos jogarem frequentemente jogos de computador onde utilizavam controlos idênticos.

ANÁLISE E AVALIAÇÃO DE RESULTADOS

Outra tarefa que saltou à vista no decurso das avaliações foi a tarefa nº3. Esta, que seria aparentemente uma tarefa de grau de dificuldade muito baixo, pois o seu percurso ideal é composto por apenas 2 passos, verificou-se problemática. O seu caminho mais eficaz está apresentado na Tabela 8.

Nº	Descrição do passo
1	Selecionar secção Eventos
2	Selecionar evento <i>IEEE Software Technology Conference</i>

Tabela 8 - Percurso mais eficaz para realizar a tarefa nº3

	Tarefa 3		Passos		Tempo
			-1-	-2-	
Utilizadores	A01	Cliques	1	1	15s
		Erros			
	A02	Cliques	5	1	1m05
		Erros	x		
	A03	Cliques	5	1	55s
		Erros	x		
	A04	Cliques	5	1	45s
		Erros	x		
	A05	Cliques	2	1	25s
		Erros			
	A06	Cliques	1	4	30s
		Erros			
	A07	Cliques	1	1	10s
		Erros			
	B01	Cliques	1	1	5s
		Erros			
	B02	Cliques	1	1	10s
		Erros			
	B03	Cliques	2	1	25s
		Erros			
C01	Cliques	1	1	5s	
	Erros				
	Média		2,3	1,3	23segs
	Desvio Padrão		1,7	0,9	15segs
	Erros		3	0	

Tabela 9 - Registo do teste de usabilidade à tarefa nº 3

Como é possível analisar na Tabela 9 os utilizadores A02, A03 e A04 foram incapazes de encontrar por si o botão do separador Eventos. No entanto, a maioria dos outros apresentaram grande facilidade em aceder ao separador necessário. A justificação para este facto prende-se com o tema do *browser* onde os testes foram realizados. Ou seja, os testes dos utilizadores A02, A03 e A04 foram realizados num *browser* que tinha toda a secção superior à página web em tons de preto, os restantes utilizadores realizaram os seus testes em *browsers* com temas de diferentes cores. Com isto conclui-se que efetivamente as cores usadas para os botões dos separadores deverão ser alteradas de forma a distinguirem-se claramente da restante barra. No entanto, de momento essa questão só deverá apresentar problemas em *browsers* configurados com temas de cor muito escura.

Ainda merecedora de atenção foi considerada a tarefa número 6. Esta, junto com a tarefa número 2 foi das que obteve mais erros quando analisada toda a amostra e foi também a tarefa onde foram verificadas mais particularidades. Os passos representando o caminho mais eficaz encontram-se na Tabela 10.

Nº	Descrição do passo
1	Selecionar a secção Eventos
2	Selecionar piso 0
3	Selecionar o Auditório
4	Selecionar a atividade com o nome <i>Implementing Security in the Cloud</i>
5	Abrir painel de detalhes
6	Selecionar orador

Tabela 10 - Percurso mais eficaz para realizar a tarefa nº6

As especificidades começam pelo utilizador A02 que tentou alcançar o objetivo de forma totalmente diferente do ideal. Neste caso ele lembrava-se de ter visto o nome da atividade pedida em tarefas anteriores e tentou chegar até ela dessa forma, em vez de procurar a partir das informações dadas no enunciado.

Outra característica observada em vários utilizadores foi uma alteração na ordenação dos passos, enquanto que o ideal seria selecionar o separador de eventos no 1º passo, as pessoas observadas só tinham tendência a mudar o separador a seguir ao passo 3, quando precisavam de selecionar uma atividade, que não estaria acessível do separador atual. Os outros erros ocorridos no passo 3 relacionaram-se com as dificuldades de interação com o ambiente tridimensional já explicadas acima.

ANÁLISE E AVALIAÇÃO DE RESULTADOS

Quanto aos erros ocorridos no passo 4 foram diretamente relacionados com o facto dos utilizadores não se terem apercebido que a listagem de atividades se alterou após a seleção de um espaço, e portanto não foram à procura da atividade pedida.

Os pontos a retirar da análise desta tarefa relacionam-se com a incrementação da experiência do utilizar em situações de alteração de informação visível. Uma forma de ajudar os utilizadores a corresponder melhor e de forma mais imediata as alterações nos dados listados de forma dinâmica seria ter um efeito de *highlight* e/ou *slow-fade* de forma a alertar o utilizador para a dinamização da informação existente. Também afetado por algum destes efeitos podia existir uma alteração do título de forma a referir de forma mais específica a informação que está atualmente visível. Além disto, seria de valor manter o estado da seleção atual cada vez que o utilizar altera o separador ativo.

Tarefa 6		Passos						Tempo
		-1-	-2-	-3-	-4-	-5-	-6-	
Utilizadores	A01 Cliques	1	2	1	1	3	1	40s
	Erros							
	A02 Cliques	1	1	5	1	1	1	2m10
	Erros			x				
	A03 Cliques	1	1	3	5	1	2	2m05
	Erros				x			
	A04 Cliques	1	1	5	1	1	1	45s
	Erros			x				
	A05 Cliques	2	1	4	1	1	1	1m05
	Erros							
	A06 Cliques	2	3	3	4	1	1	1m20
	Erros							
	A07 Cliques	2	2	1	2	<u>1</u>	1	40s
	Erros							
	B01 Cliques	2	2	1	2	1	1	1m
	Erros							
	B02 Cliques	3	2	3	1	1	1	35s
	Erros							
	B03 Cliques	1	1	5	3	1	1	45s
	Erros			x				
C01 Cliques	1	1	2	5	2	1	1m20	
Erros				x				
Média		1,5	1,5	3,0	2,4	1,3	1,1	1m08s
Desvio Padrão		0,7	0,7	1,5	1,6	0,6	0,3	32secs
Erros		0	0	3	2	0	0	

Tabela 11 - Registo do teste de usabilidade à tarefa nº6

Além daqueles já aqui referidos foram observados mais 5 erros distribuídos pelas restantes tarefas que não pareceram diretamente associados a nenhum problema específico nem recorrente durante os testes à aplicação.

Após esta avaliação de usabilidade foi ainda pedido aos utilizadores que respondessem a um questionário. Na primeira parte, dedicada a perguntas específicas ao edifício apresentado, nomeadamente quanto à forma, número de pisos e localização de certos elementos mencionados nas tarefas previamente realizadas, a reação dos utilizadores foi de surpresa, pois estes não estavam atentos a determinados detalhes que lhes foram perguntados.

Apesar disso os resultados obtidos permitem observar que os utilizadores conseguiram ainda absorver de forma razoável, dado o tempo reduzido do teste e as suas condições, alguma informação relevante sobre o edifício que lhes foi apresentado.

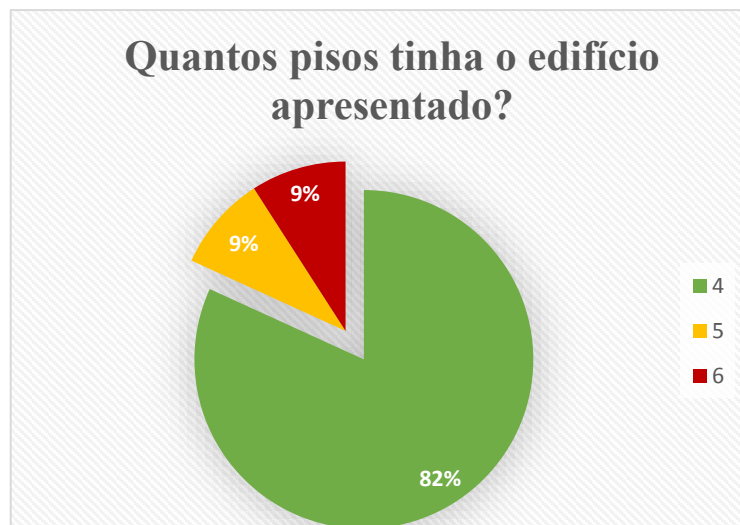


Figura 27 - Gráfico de resultados da questão 'Quantos pisos tinha o edifício apresentado?'

É possível ver no gráfico da Figura 27 a percentagem de respostas dadas a uma das perguntas, tendo em conta que 4 (o valor a verde) era a resposta correta. Ou seja, apesar das pessoas não estarem preparadas para perguntas deste carácter, ainda assim a grande maioria foi capaz de indicar corretamente o número de pisos do edifício. No entanto, como seria esperado dado os diferentes graus de dificuldade das questões realizadas, nem todas as respostas obtiveram resultados tão corretos nem consensuais das pessoas inquiridas.

Uma pergunta que suscitou grandes dúvidas foi, sem qualquer dúvida, a questão 'Quantos eventos decorriam no mês de Junho?'. Os inquiridos deram respostas que variam de 0 a 5 eventos, quando a resposta correcta seria 2. As respostas podem ser analisadas no gráfico da Figura 28, onde estão representadas no eixo horizontal as opções de resposta – o número de eventos, e no

eixo horizontal a contagem de utilizadores a cada resposta. A análise retirada desta questão foi que apesar de algumas respostas erradas terem sido dadas devido a esquecimento, existe também uma relação com a dúvida de entre os elementos listados quais eram atividades e quais seriam eventos. Em consequência disto, e tendo em conta que havia um número muito maior de atividades, nota-se uma alta incidência na resposta '5' dado esta ser a opção com valor mais alto entre as opções.



Figura 28 - Gráfico de resultados da questão 'Quantos eventos decorriam no mês de Junho?'

Falando agora da segunda parte do questionário, de opinião pessoal, pode-se concluir que apesar de algumas dificuldades localizadas, o *feedback* final foi muito positivo. Analisando as primeiras 7 perguntas desta parte que consistiam em avaliar, de 1 a 6, determinadas características da aplicação, podemos verificar que na maioria dos casos a pontuação obtida se concentra apenas nos 50% mais altos de avaliação.

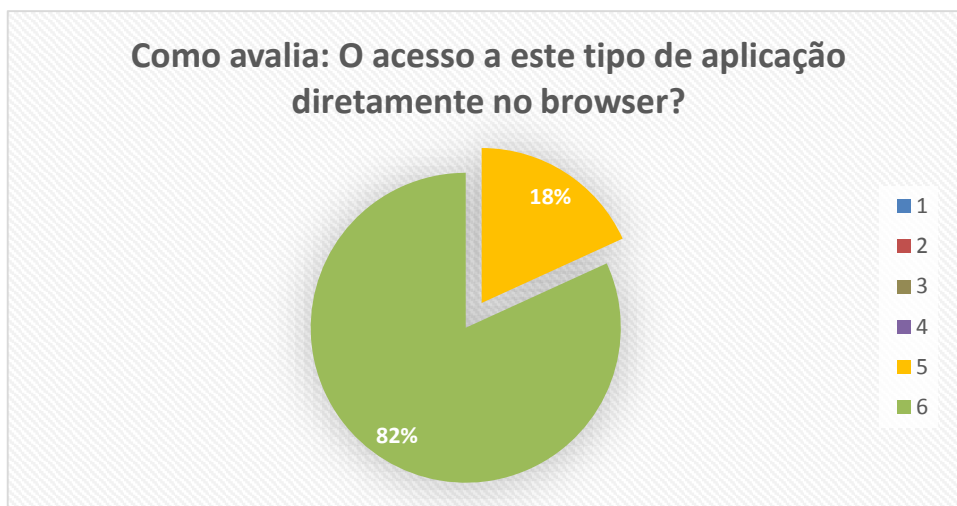


Figura 29 - Gráfico de resultados da questão 'Como avalia: o acesso a este tipo de aplicação diretamente no *browser*?'

Aquela que mais se denotou pela sua avaliação concentrada nos valores mais altos, Figura 29, foi a questão relacionada com a valorização do acesso a uma aplicação deste tipo disponível de forma direta no *browser*. Pode-se concluir que a amostra de utilizadores concorda com um dos ideais básicos do presente projeto - a oferta deste tipo de aplicação sem necessidade de instalação de *software* e de uma forma generalizada no *browser* é uma grande mais-valia.

A componente que obteve respostas com avaliação inferior foi o painel de eventos. Este módulo já foi mencionado anteriormente quanto aos testes de usabilidade, e como já foi referido, apesar das pessoas demonstrarem facilidade em usar e realizar as tarefas que envolviam este painel, algumas delas revelaram alguma dificuldade em distinguir de forma imediata que elementos eram atividades e quais eram os eventos, assim como não acharam notável a mudança dinâmica de conteúdo.



Figura 30 - Gráfico de resultados da questão 'Como avalia: A navegação no painel de eventos?'

Todos os dados registados referentes tanto aos testes de usabilidade como ao inquérito realizado se encontram em anexo no final deste relatório.

5.3 Resumo e Conclusões

Após a análise de todos os resultados obtidos com os testes realizados podemos concluir que a receptividade dos utilizadores à aplicação foi bastante positiva. Os utilizadores apreciam, de

ANÁLISE E AVALIAÇÃO DE RESULTADOS

forma geral, a experiência de utilização, e foram também capazes de realizar a maioria das tarefas sem necessidade de ajuda. Verificam-se ainda algumas falhas, nomeadamente na distinção dos conceitos de atividade e evento, nos botões dos separadores de módulos e possivelmente na falta de auxiliares de interação. No entanto, todos estes problemas não pareceram ser potencialmente críticos da perspetiva do utilizador do sistema.

Quanto a dois pontos cruciais - a utilidade e o grau de inovação duma aplicação deste carácter - a opinião foi consensual, tendo os utilizadores concordado que ambas são de grande valor

Capítulo 6

Conclusões e Trabalho Futuro

Da investigação inicialmente realizada, foi concluído que a utilização de ambientes virtuais tem verificado um crescimento em áreas pouco habituais até ao momento, seja no âmbito da segurança, comércio, saúde ou mesmo simulação de desastres. Dada a crescente qualidade e acessibilidade destes ambientes, a sua aplicação começa a deixar de se cingir aos jogos de vídeo. Nesta dissertação foi criada uma aplicação que visa alcançar o utilizador não especializado para a divulgação de espaços e eventos. Com este sistema pretende-se oferecer ao utilizador uma experiência de fácil usabilidade que seja interessante e apelativa, mas que tem toda uma outra componente focada na utilidade, nomeadamente na divulgação de informação.

O módulo 3D permite ao utilizador uma experiência interativa, com um espaço que o mesmo conheça ou queira conhecer, enquanto os módulos informativos complementares, que consistem na componente de utilidade da aplicação, oferecem um conjunto de informação conexa. É a junção destas duas vertentes que faz desta aplicação algo inovador e com valor para os diferentes utilizadores, tanto aquele que divulga o seu espaço, como o utilizador final que faz uso dessa divulgação. O último ingrediente que pode fazer desta solução uma ideia de sucesso relaciona-se com o alcance de grandes massas, e isso consegue-se com a simplicidade de acesso à aplicação, a sua disponibilização de forma direta no *browser* e a sua compatibilidade com diversas plataformas.

No decurso desta dissertação foram satisfeitos os seguintes objetivos propostos:

- Estudo e investigação do estado da arte quanto a soluções tecnológicas para a *renderização* de ambientes 3D de forma direta no *browser*. Foram analisados projetos com elos de ligação ao sistema a construir e foram estudadas várias tecnologias de *renderização* gráfica tridimensional para a Web. Estudaram-se também técnicas de interação com aplicações no contexto Web.

- Desenvolvimento de um sistema capaz de disponibilizar no *browser* uma cena 3D a partir de modelos exportados da ferramenta CityEngine. Sistema que foi construído de forma integrada com o *back-office* do produto aLIVE Places e que disponibiliza uma interação apelativa ao utilizador de forma compatível com a componente de utilidade das informações igualmente disponíveis.
- Criação de uma nova e inovadora camada no projeto com o intuito de alcançar outro âmbito de utilizadores a partir da divulgação de informações relativas a atividades e eventos a decorrer no espaço promovido. Incremento da experiência do utilizador a partir da adição de funcionalidades essencialmente visuais, tal como distinções de espaços e eventos a partir de cores, e animações entre espaços e percursos.

6.1 Trabalho Futuro

Apesar da satisfação com os objetivos alcançados existem naturalmente possibilidades para novas funcionalidades ou melhorias às já implementadas de forma a incrementar o valor apreciado pelos utilizadores do sistema.

Como foi referido na secção 5.2, uma funcionalidade que não foi desenvolvida devido à falta de dispositivos compatíveis foi a adaptação da página web de forma a garantir a sua acessibilidade a partir de *tablets* Android. Posteriormente, caso o acesso a WebGL em *iOS* passe a estar disponibilizado, este deverá igualmente ser tido em conta. No entanto o sistema foi construído a pensar nesta futura adaptação portanto o grau de dificuldade deste trabalho futuro não deverá ser alto.

Outro ponto que poderá ser tido em conta relaciona-se com a utilização de cache e funcionalidades de *WebStorage* de forma a acelerar o carregamento inicial da aplicação. Esta melhoria deverá ser efetuada tanto do lado da aplicação como do *back-office* de forma a minimizar o tempo de espera do utilizador cada vez que ele abre um projeto.

Por último, seria útil efetuar as correções resultantes dos testes de usabilidade realizados, nomeadamente em relação aos botões dos separadores e à distinção entre elementos no componente de eventos e posteriormente realizar mais testes com outro grupo de utilizadores de forma a verificar o grau de evolução.

Possivelmente também poderiam ser adicionadas outras animações em determinados aspetos da interação mas estas deveriam ser estudadas com algum cuidado de forma a não dar origem a uma perda da simplicidade e clareza na usabilidade do ponto de vista do utilizador final.

Referências

1. Sutherland IE. Sketchpad: A man-machine graphical communication system [Internet]. University of Cambridge, Computer Laboratory; 2003 Set. Report No.: UCAM-CL-TR-574. Obtido de: <http://www.cl.cam.ac.uk/techreports/UCAM-CL-TR-574.pdf>
2. 3Decide - Vision and Mission [Internet]. [citado 29 de Novembro de 2012]. Obtido de: <http://www.3decide.com/content/vision>
3. Jankowski J. A taskonomy of 3D web use. Proc 16th Int Conf 3d Web Technol [Internet]. New York, NY, USA: ACM; 2011 [citado 26 de Janeiro de 2013]. p 93–100. Obtido de: <http://doi.acm.org/10.1145/2010425.2010443>
4. Grupp J. WebGL and other Technologies for hardware accelerated 3D in Browsers. 2012 [citado 28 de Novembro de 2012]; Obtido de: <http://jonathangrupp.com/images/webgl/WebGL.SotA.JonathanGrupp.pdf>
5. Ortiz S. Is 3d finally ready for the web? Computer. 2010;43(1):14–6.
6. Chittaro L, Gatla VK, Venkataraman S. The Interactive 3D BreakAway Map: a navigation and examination aid for multi-floor 3D worlds. Int Conf Cyberworlds 2005. 2005. p 8 pp. –66.
7. Oulasvirta A, Nurminen A, Nivala A, Oulasvirta A, Nurminen A, Nivala A, Oulasvirta A, Nurminen A, Nivala A. Interacting with 3d and 2d Mobile Maps: An Exploratory Study. 2007.
8. Ping S, Aiguo Z, Tao Y. Study and Practice of Virtual Campus Modeling and Touring. 2010 Int Conf Multimed Technol Icmt. 2010. p 1 –3.
9. Oudatzi K. Virtual reality in restoration of historic buildings: 3d model projection of the restoration project of Alaca Imaret C #x00E2;mi with intuitive and interactive application through hyper realism technology. 2010 16th Int Conf Virtual Syst Multimed Vsmm. 2010. p 361 –364.
10. What is Second Life? [Internet]. [citado 3 de Fevereiro de 2013]. Obtido de: <http://secondlife.com/whatis/?lang=en-US>
11. Chen B, Huang F, Fang Y. Integrating virtual environment and GIS for 3D virtual city development and urban planning. Geosci Remote Sens Symp Igarss 2011 Ieee Int. 2011. p 4200 –4203.
12. Al-Humadi R. 3D Virtual Shopping Mall: Towards Transformation of levels from 2D to 3D. LNU; 2012.
13. Chen T, Pan Z, Zheng J-M. EasyMall - An Interactive Virtual Shopping System. Fifth Int Conf Fuzzy Syst Knowl Discov 2008 Fskd 08. 2008. p 669 –673.

14. Xiao L, Wu Z, Zheng Y. Application research on virtual simulation technology in building construction. 2011 Int Conf Electron Commun Control Icecc. 2011. p 162 –165.
15. HCI (human-computer interaction) [Internet]. [citado 3 de Fevereiro de 2013]. Obtido de: <http://searchcio-midmarket.techtarget.com/definition/HCI>
16. HCI [Internet]. Webopedia. [citado 3 de Fevereiro de 2013]. Obtido de: <http://www.webopedia.com/TERM/H/HCI.html>
17. Karray F, Alemzadeh M, Saleh JA, Arab MN. Human-Computer Interaction: Overview on State of the Art. Int J Smart Sens Intell Syst. Março de 2008;1(1):137–59.
18. Pollefeys M, Van Gool L, Akkermans I, De Becker D, Demuynck K. A guided tour to virtual Sagalassos. Proc 2001 Conf Virtual Real Archeol Cult Herit [Internet]. New York, NY, USA: ACM; 2001 [citado 3 de Fevereiro de 2013]. p 213–8. Obtido de: <http://doi.acm.org/10.1145/584993.585027>
19. Bonis B, Stamos J, Vosinakis S, Andreou I, Panayiotopoulos T. A platform for virtual museums with personalized content. Multimed Tools Appl. Abril de 2009;42(2):139–59.
20. Laakso K. Evaluating the use of navigable three-dimensional maps in mobile devices. [Helsinki]: Master's Thesis, Helsinki University of Technology; 2002.
21. Burigat S, Chittaro L. Location-aware visualization of VRML models in GPS-based mobile guides. Proc Tenth Int Conf 3d Web Technol [Internet]. New York, NY, USA: ACM; 2005 [citado 3 de Fevereiro de 2013]. p 57–64. Obtido de: <http://doi.acm.org/10.1145/1050491.1050499>
22. Rakkolainen I, Vainio T. A 3D City Info for mobile users. Comput Graph. 2001;25(4):619 – 625.
23. Walsh AE, Bourges-Sévenier M. Core Web3D. Upper Saddle River, NJ, USA: Prentice Hall PTR; 2001.
24. Web3D Consortium. X3D & VRML, the Most Widely Used Formats [Internet]. X3d Vrml Most Widely Used Formats. [citado 29 de Janeiro de 2013]. Obtido de: <http://www.web3d.org/realtime-3d/x3d-vrml/x3d-vrml-most-widely-used-3d-formats>
25. Web3D Consortium. What is X3D? [Internet]. [citado 31 de Janeiro de 2013]. Obtido de: <http://www.web3d.org/realtime-3d/x3d/what-x3d/>
26. Web3D Consortium. About Web3D Consortium [Internet]. [citado 31 de Janeiro de 2013]. Obtido de: <http://www.web3d.org/realtime-3d/about>
27. Chen JX, Chen C. Foundations of 3D Graphics Programming: Using JOGL and Java3d. Springer; 2008.
28. Xu Z, Yan Y, Chen JX. OpenGL programming in Java. Comput Sci Eng. Fevereiro de 2005;7(1):51 – 55.
29. Gouesse J. Java3D now works with JOGL 2.0 [Internet]. [citado 1 de Fevereiro de 2013]. Obtido de: <http://forum.jogamp.org/Java3D-now-works-with-JOGL-2-0-td3732206.html>
30. Adobe Systems Incorporated. Stage 3D [Internet]. Obtido de: <http://www.adobe.com/devnet/flashplayer/stage3d.html>

31. Microsoft Silverlight [Internet]. Wikipedia Free Encycl. 2013 [citado 4 de Fevereiro de 2013]. Obtido de: http://en.wikipedia.org/w/index.php?title=Microsoft_Silverlight&oldid=536448157
32. Khronos Group. Khronos Details WebGL Initiative to Bring Hardware-Accelerated 3D Graphics to the Internet [Internet]. [citado 2 de Fevereiro de 2013]. Obtido de: <http://www.khronos.org/news/press/khronos-webgl-initiative-hardware-accelerated-3d-graphics-internet>
33. Parisi T. AJAX3D - The Open Platform For Rich 3D Web Applications. Ajaxworld Mag [Internet]. 26 de Janeiro de 2007 [citado 2 de Fevereiro de 2013]; Obtido de: <http://ajax.syscon.com/node/327939>
34. Roth SD. Ray casting for modeling solids. Comput Graph Image Process. 1982;18(2):109 – 144.

Anexo A – Testes de Usabilidade

A.1 – Tarefa nº1: Lista de Passos e Resultados

Nº	Descrição do passo
1	Secção Espaços
2	Selecionar imagem da empresa <i>CulturePrint</i>
3	Abrir painel de detalhes
4	<i>Scroll down</i> até ao nº de telefone

	Tarefa 1	Passos				Tempo	
		-1-	-2-	-3-	-4-		
Utilizadores	A01	Cliques	2	1	1	1	1m10s
		Erros					
	A02	Cliques	1	5	3	1	1m15s
		Erros		x			
	A03	Cliques	1	1	1	1	15s
		Erros					
	A04	Cliques	1	1	1	1	10s
		Erros					
	A05	Cliques	1	2	1	1	25s
		Erros					
	A06	Cliques	1	1	1	1	40s
		Erros					
	A07	Cliques	1	1	1	1	20s
		Erros					
	B01	Cliques	1	1	1	1	30s
		Erros					
	B02	Cliques	1	3	1	1	45s
		Erros					
	B03	Cliques	1	1	1	1	10s
		Erros					
C01	Cliques	1	1	1	1	20s	
	Erros						
	Média		1,1	1,6	1,2	1,0	33segs
	Desvio Padrão		0,3	1,2	0,6	0,0	22segs
	Erros		0	1	0	0	

A.2 – Tarefa nº2: Lista de Passos e Resultados

Nº	Descrição do passo
1	Secção Espaços (e localizar-se em vista de edifício – já por defeito no início da aplicação)
2	Seleção da imagem da empresa – Ver nº do piso
3	Selecionar escadas – Ver número da sala

	Tarefa 2	Passos			Tempo	
		-1-	-2-	-3-		
Utilizadores	A01	Cliques	1	1	3	50s
		Erros				
	A02	Cliques	1	1	2	25s
		Erros				
	A03	Cliques	1	1	5	1m40s
		Erros			x	
	A04	Cliques	1	1	5	1m30s
		Erros			x	
	A05	Cliques	1	1	3	45s
		Erros				
	A06	Cliques	1	2	5	1m55s
		Erros			x	
	A07	Cliques	1	2	2	45s
		Erros				
	B01	Cliques	1	3	4	1m
		Erros				
	B02	Cliques	1	1	1	25s
		Erros				
	B03	Cliques	1	1	5	1m
		Erros			x	
	C01	Cliques	5	1	2	50s
	Erros	x				
	Média	1,4	1,4	3,4	1min	
	Desvio Padrão	1,1	0,6	1,4	28segs	
	Erros	1	0	4		

A.3 – Tarefa nº3: Lista de Passos e Resultados

Nº	Descrição do passo
1	Selecionar secção Eventos
2	Selecionar evento <i>IEEE Software Technology Conference</i>

	Tarefa 3	Passos		Tempo	
		-1-	-2-		
Utilizadores	A01	Cliques	1	1	15s
		Erros			
	A02	Cliques	5	1	1m05
		Erros	x		
	A03	Cliques	5	1	55s
		Erros	x		
	A04	Cliques	5	1	45s
		Erros	x		
	A05	Cliques	2	1	25s
		Erros			
	A06	Cliques	1	4	30s
		Erros			
	A07	Cliques	1	1	10s
		Erros			
	B01	Cliques	1	1	5s
		Erros			
	B02	Cliques	1	1	10s
		Erros			
	B03	Cliques	2	1	25s
		Erros			
C01	Cliques	1	1	5s	
	Erros				
	Média		2,3	1,3	23segs
	Desvio Padrão		1,7	0,9	15segs
	Erros		3	0	

A.4 – Tarefa nº4: Lista de Passos e Resultados

Nº	Descrição do passo
1	Selecionar secção Eventos
2	Selecionar evento <i>IEEE Software Technology Conference</i>

	Tarefa 4	Passos		Tempo	
		-1-	-2-		
Utilizadores	A01	Cliques	1	1	10s
		Erros			
	A02	Cliques	1	2	15s
		Erros			
	A03	Cliques	1	2	15s
		Erros			
	A04	Cliques	1	3	20s
		Erros			
	A05	Cliques	1	1	20s
		Erros			
	A06	Cliques	1	4	20s
		Erros			
	A07	Cliques	2	2	30s
		Erros			
	B01	Cliques	1	2	20s
		Erros			
	B02	Cliques	1	2	20s
		Erros			
	B03	Cliques	1	2	15s
		Erros			
C01	Cliques	1	2	20s	
	Erros				
	Média		1,1	2,1	17secs
	Desvio Padrão		0,3	0,8	3,4secs
	Erros		0	0	

A.5 – Tarefa nº5: Lista de Passos e Resultados

Nº	Descrição do passo
1	Secção Eventos
2	Entrar no evento <i>OPO ART Week</i>
3	Selecionar a atividade <i>Fotografia de Paisagem - O Douro entre Pontes</i>
4	Abrir o painel de detalhes

	Tarefa 5		Passos				Tempo
			-1-	-2-	-3-	-4-	
Utilizadores	A01	Cliques	1	1	3	1	35s
		Erros					
	A02	Cliques	1	5	1	1	50s
		Erros		x			
	A03	Cliques	1	2	4	1	30s
		Erros					
	A04	Cliques	1	2	1	1	20s
		Erros					
	A05	Cliques	1	3	1	1	25s
		Erros					
	A06	Cliques	1	5	1	2	1m20
		Erros		x			
	A07	Cliques	1	3	1	1	20s
		Erros					
	B01	Cliques	1	2	1	1	10s
		Erros					
	B02	Cliques	1	2	1	1	15s
		Erros					
	B03	Cliques	1	1	2	1	30s
		Erros					
C01	Cliques	1	1	2	2	20s	
	Erros						
	Média		1,0	2,5	1,6	1,2	30segs
	Desvio Padrão		0,0	1,4	1,0	0,4	19segs
	Erros		0	2	0	0	

A.6 – Tarefa nº6: Lista de Passos e Resultados

Nº	Descrição do passo
1	Selecionar a secção Eventos
2	Selecionar piso 0
3	Selecionar o Auditório
4	Selecionar a atividade com o nome <i>Implementing Security in the Cloud</i>
5	Abrir painel de detalhes
6	Selecionar orador

	Tarefa 6	Passos						Tempo
		-1-	-2-	-3-	-4-	-5-	-6-	
Utilizadores	A01 Cliques	1	2	1	1	3	1	40s
	Erros							
	A02 Cliques	1	1	5	1	1	1	2m10
	Erros			x				
	A03 Cliques	1	1	3	5	1	2	2m05
	Erros				x			
	A04 Cliques	1	1	5	1	1	1	45s
	Erros			x				
	A05 Cliques	2	1	4	1	1	1	1m05
	Erros							
	A06 Cliques	2	3	3	4	1	1	1m20
	Erros							
	A07 Cliques	2	2	1	2	<u>1</u>	1	40s
	Erros							
	B01 Cliques	2	2	1	2	1	1	1m
	Erros							
	B02 Cliques	3	2	3	1	1	1	35s
	Erros							
	B03 Cliques	1	1	5	3	1	1	45s
	Erros			x				
C01 Cliques	1	1	2	5	2	1	1m20	
Erros				x				
Média		1,5	1,5	3,0	2,4	1,3	1,1	1m08s
Desvio Padrão		0,7	0,7	1,5	1,6	0,6	0,3	32secs
Erros		0	0	3	2	0	0	

A.7 – Tarefa nº7: Lista de Passos e Resultados

Nº	Descrição do passo
1	Selecionar a secção Eventos
2	Selecionar dia 09/06/2013 no calendário
3	Encontrar atividade a decorrer das 10h às 11h30
4	Abrir combobox de percursos
5	Selecionar percurso ‘Rota B’
6	Executar um clique no botão ‘Ver Percurso’

	Tarefa 7	Passos						Tempo
		-1-	-2-	-3-	-4-	-5-	-6-	
Utilizadores	A01 Cliques	1	2	4	1	1	1	1m15s
	Erros							
	A02 Cliques	1	2	1	1	1	1	35s
	Erros							
	A03 Cliques	1	2	3	1	1	1	1m10s
	Erros							
	A04 Cliques	1	2	1	2	1	1	50s
	Erros							
	A05 Cliques	1	2	1	1	1	1	25s
	Erros							
	A06 Cliques	1	2	1	1	1	1	25s
	Erros							
	A07 Cliques	1	2	1	4	<u>1</u>	1	55s
	Erros							
	B01 Cliques	1	2	5	2	1	4	1m
	Erros			x				
	B02 Cliques	1	2	1	1	1	1	25s
	Erros							
	B03 Cliques	1	2	1	1	1	1	25s
	Erros							
C01 Cliques	1	2	1	1	1	5	1m10s	
Erros						x		
Média		1,0	2,0	1,8	1,5	1,0	1,6	47segs
Desvio Padrão		0,0	0,0	1,4	0,9	0,0	1,4	20segs
Erros		0	0	1	0	0	1	

Anexo B – Questionários

B.0.1 – Enunciado do questionário realizado: Parte I

Tendo em conta o conjunto de tarefas que acabou de executar, por favor responda às seguintes questões:

Quantos pisos tinha o edifício apresentado? *

1 2 3 4 5 6

Qual era a forma do edifício apresentado? *

(Aproximadamente e considere-se no exterior, de frente para a porta de entrada)

- I
- Γ
- L
- Π
- U

Em que piso se encontrava o auditório?

0 1 2 3 4 5

Em que lado do edifício se encontrava o auditório? *

Considere-se no exterior, de frente para a porta de entrada

- Esquerda
- Direita
- Frente
- Trás

Em que piso se encontravam as salas da empresa Tecla Colorida? *

0 1 2 3 4 5

Quantos eventos iam decorrer no mês de Junho? *

0 1 2 3 4 5

[Continue »](#)

B.0.2 - Enunciado do questionário realizado: Parte II

Agora tendo em conta a sua opinião pessoal sobre a utilização da aplicação, por favor responda ao seguinte:

Numa escala de 1 a 6, como avalia os seguintes pontos:

1 – Muito má, 2 – Fraca, 3 – Abaixo da média, 4 – Acima da média, 5 – Boa, 6 – Excelente

A visualização da cena 3D? *

1 2 3 4 5 6

A navegação entre os vários pisos do modelo do edifício? *

1 2 3 4 5 6

A navegação no painel de eventos (Painel à direita com calendário e informação de atividades)? *

1 2 3 4 5 6

A acessibilidade do painel de detalhes (Painel slider inferior)? *

1 2 3 4 5 6

O acesso a informações de espaços e eventos associados à própria representação tridimensional? *

1 2 3 4 5 6

O acesso a este tipo de aplicação diretamente no browser? *

1 2 3 4 5 6

De forma geral, a experiência de utilização da aplicação? *

1 2 3 4 5 6

B.0.3 – Enunciado do questionário realizado: Parte III

Acha que uma aplicação deste género tem utilidade? *

- Nenhuma
- Pouca
- Bastante
- Muita

9. Qual seria a sua avaliação se o seu espaço de eventos favorito (Casa da Música, Alfândega do Porto, Coliseu do Porto, etc) passasse a utilizar esta aplicação para divulgar as suas atividades e os seus espaços? *

- Negativa
- Neutra
- Positiva
- Muito Positiva

Tem algum comentário ou sugestão que gostaria de partilhar sobre esta aplicação? *

Qualquer feedback que ajude a melhorar este produto é bem-vindo. Obrigado!

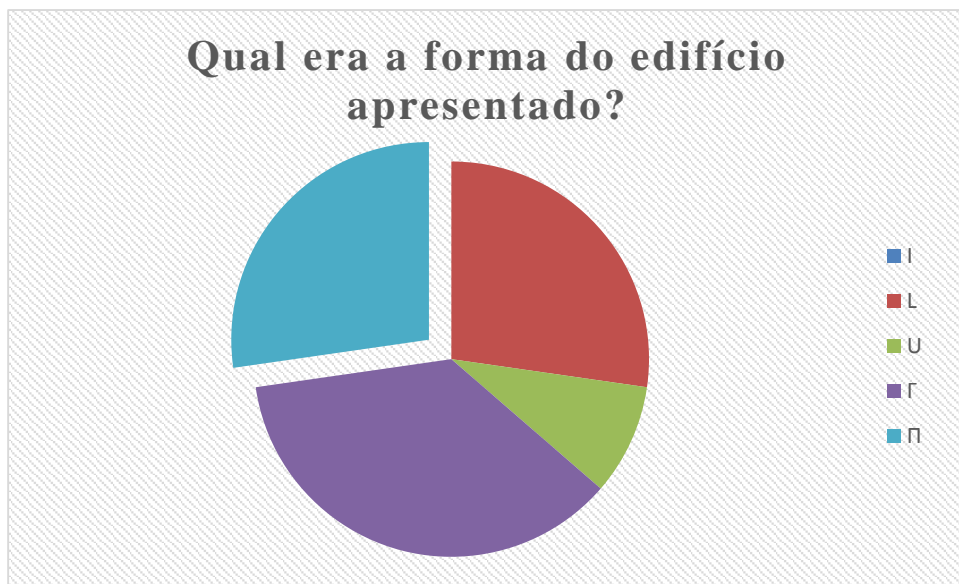
B.1.1 – Parte I: Quantos pisos tinha o edifício apresentado?

Opções	Utilizadores
1	0
2	0
3	0
4	9
5	1
6	1



B.1.1 – Parte I: Qual era a forma do edifício apresentado?

Opções	Utilizadores
I	0
L	3
U	1
Γ	4
Π	3



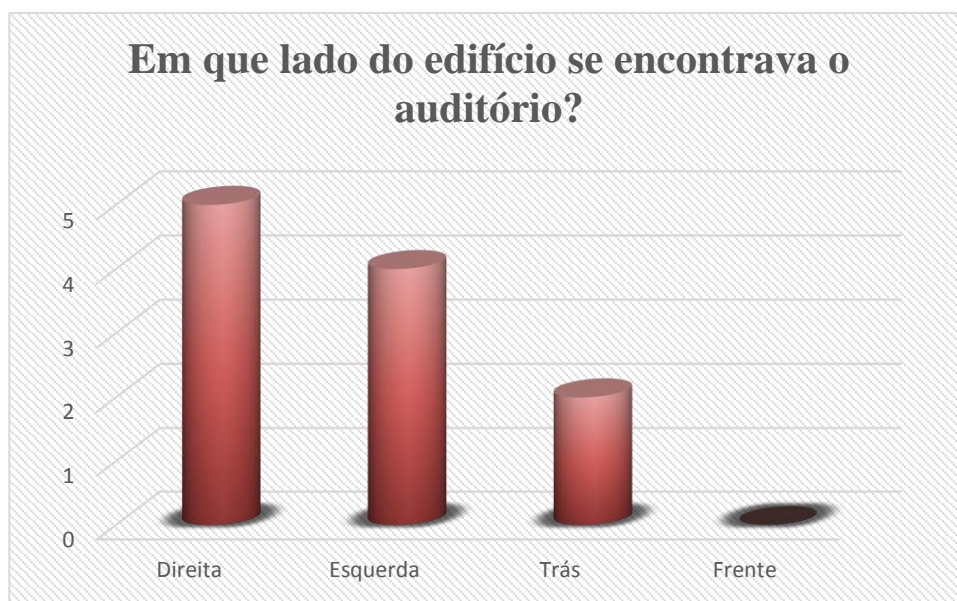
B.1.3 – Parte I: Em que piso se encontrava o auditório?

Opções	Utilizadores
0	10
1	0
2	0
3	1
4	0
5	0



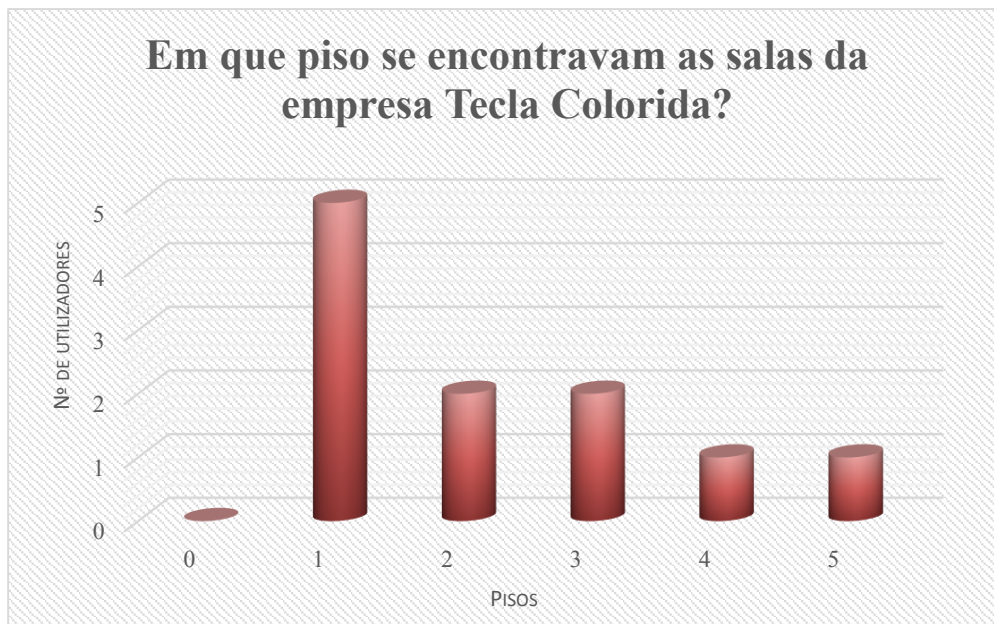
B.1.4 – Parte I: Em que lado do edifício se encontrava o auditório?

Opções	Utilizadores
Direita	5
Esquerda	4
Trás	2
Frente	0



B.1.5 – Parte I: Em que piso se encontravam as salas da empresa Tecla Colorida?

Opções	Utilizadores
0	0
1	5
2	2
3	2
4	1
5	1



B.1.6 – Parte I: Quantos eventos decorriam no mês de Junho?

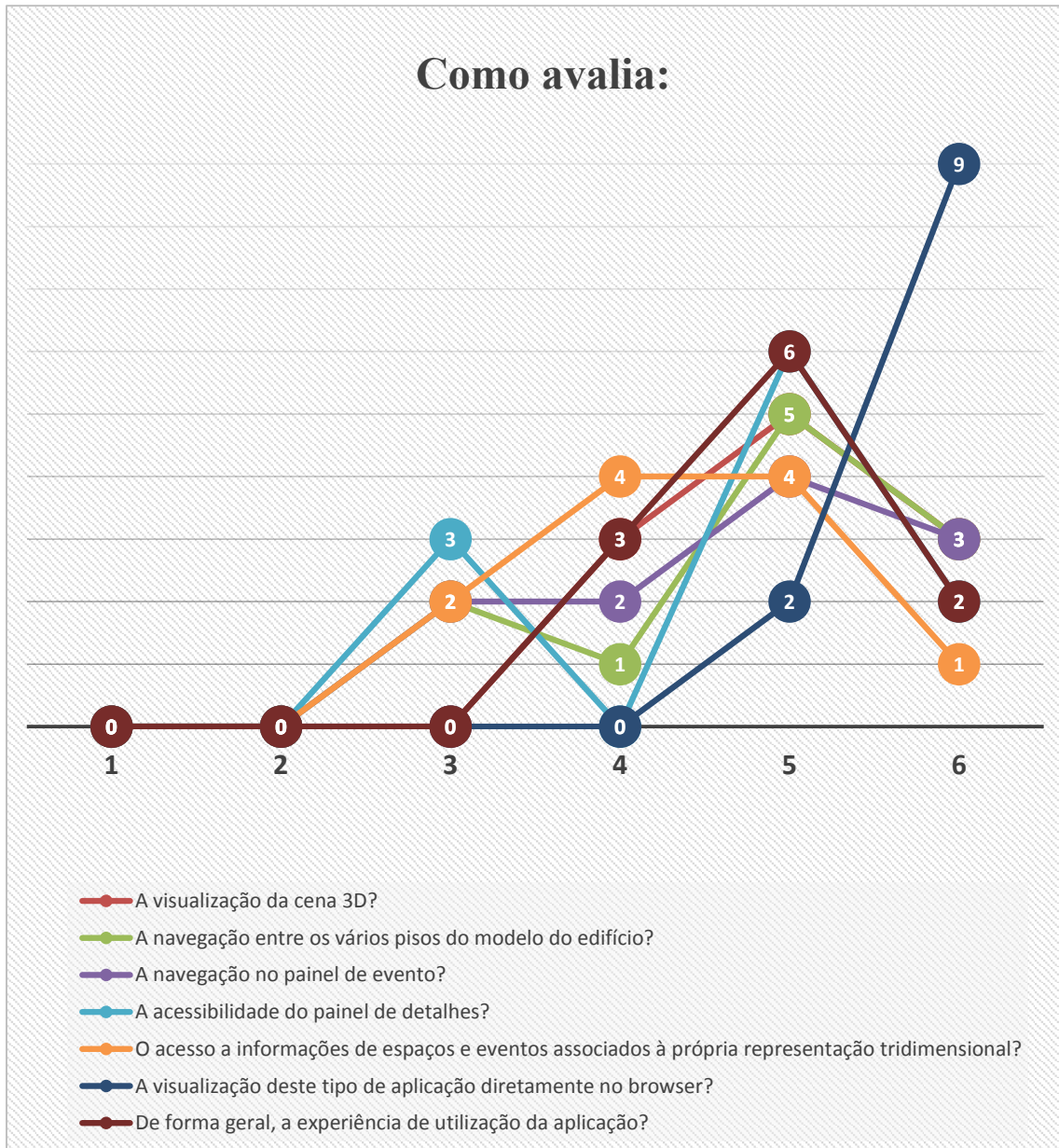
Opções	Utilizadores
0	0
1	1
2	4
3	1
4	1
5	4



B.2 – Parte II: Numa escala de 1 a 6, como avalia os seguintes pontos:

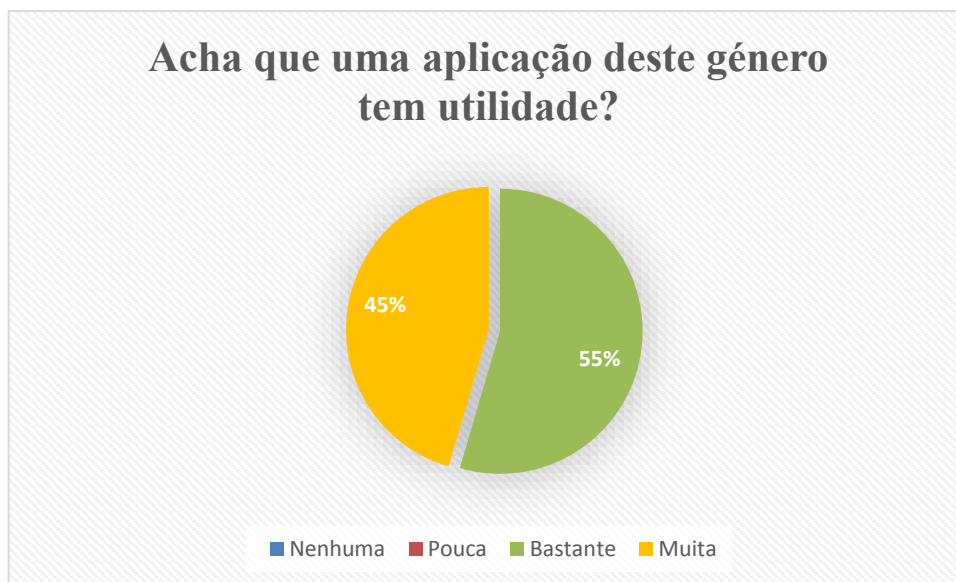
Avaliação	A visualização da cena 3D?	A navegação entre os vários pisos do modelo do edifício?	A navegação no painel de evento?	A acessibilidade do painel de detalhes?
1	0	0	0	0
2	0	0	0	0
3	0	2	2	3
4	3	1	2	0
5	5	5	4	6
6	3	3	3	2

Avaliação	O acesso a informações de espaços e eventos associados à própria representação tridimensional?	A visualização deste tipo de aplicação diretamente no browser?	De forma geral, a experiência de utilização da aplicação?
1	0	0	0
2	0	0	0
3	2	0	0
4	4	0	3
5	4	2	6
6	1	9	2



B.3.1 – Parte III: Acha que uma aplicação deste género tem utilidade?

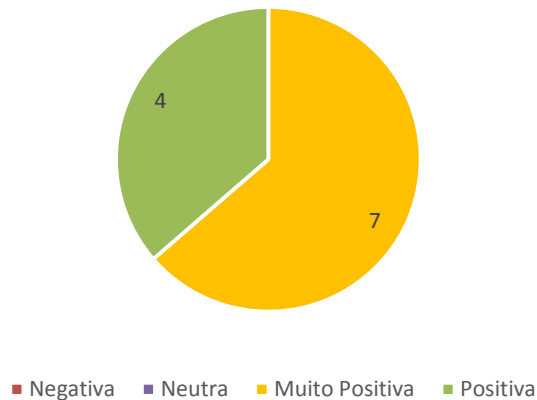
Opções	Utilizadores
Nenhuma	0
Pouca	0
Bastante	6
Muita	5



B.3.2 – Parte III: Qual seria a sua avaliação se o seu espaço de eventos favorito (Casa da Música, Alfândega do Porto, Coliseu do Porto, etc) passasse a utilizar esta aplicação para divulgar as suas atividades e os seus espaços?

Opções	Utilizadores
Negativa	0
Neutra	0
Muito Positiva	7
Positiva	4

Qual seria a sua avaliação se o seu espaço de eventos favorito passasse a utilizar esta aplicação para divulgar as suas atividades e os seus espaços?



B.3.3 – Parte III: Tem algum comentário ou sugestão que gostaria de partilhar sobre esta aplicação?

Lista de sugestões

Navegação entre pisos poderia não necessitar do piso atual;
Identificação dos números de sala diretamente na vista de piso poderia facilitar a perceção;
Colocar no calendário os dias em que existem eventos associados de outra cor por forma a destacar os dias ocupados dos que não estão. Talvez associar uma cor a cada evento para ver no calendário.
Colocar link para eventos no formulário que aparece depois da seleção de uma sala ou então fazer highlight do botão no topo superior direito.

Não encontrei o painel de eventos.
Gostava de um maior feedback visual quando existem alterações no painel da direita enquanto se navega (ex: fadeIn mais demorado da informação para ver que há alteração).
Difícil lembrar qual o uso do duplo clique e um só clique, talvez fosse porreiro haver botão para centrar na imagem.

O botão de eventos está um pouco "escondido". Passa um bocado despercebido na primeira utilização, talvez por causa da informação que existe ao seu redor

Distinguir atividades e eventos foi relativamente confuso.
Por vezes não é muito intuitivo se devo aceder à informação através de duplo clique, pelo painel inferior ou pelo painel lateral

A movimentação entre pisos e mesmo dentro de um piso tem de ser trabalhada para responder melhor (talvez usar atalhos de teclado). Mas de resto está muito bom.