



André João Ribeiro Belo

# ConimbrigAR: Realidade Aumentada Aplicada Sobre Mosaicos Romanos

Relatório de Dissertação/Estágio

Mestrado em Engenharia Informática

orientada por Prof. Jorge C. S. Cardoso  
e apresentada ao Departamento de Engenharia Informática  
da Faculdade de Ciências e Tecnologia da Universidade de Coimbra



## **Agradecimentos**

Esta etapa não teria sido alcançada sem o apoio e dedicação por parte de várias pessoas ao longo de todo o percurso da minha formação. Quero, portanto, agradecer a todos aqueles que, direta ou indiretamente, contribuíram para o meu sucesso.

Em primeiro lugar, quero agradecer ao meu orientador, Professor Jorge Cardoso, pela sua orientação exemplar. Por acreditar nas minhas capacidades. Pela sua disponibilidade, pelos seus conselhos e por toda a ajuda valiosa prestada ao longo desta etapa.

Aos meus pais, porque sem eles, nada disto teria sido possível. Por nunca desistirem de mim. Por apoiarem incondicionalmente os meus sonhos. Pela educação exemplar que me deram. Por me transmitirem valores importantes que levarei comigo para o resto da minha vida. Pelos sacrifícios. Obrigado pai. Obrigado mãe. A ambos, a minha eterna gratidão.

À minha namorada, Vanessa, pelo seu apoio crucial neste longo percurso. Pela sua paciência ilimitada. Por estar sempre presente, sobretudo nos momentos mais difíceis. Por nunca me deixar desistir e me dar forças para continuar. Por acreditar sempre em mim. Obrigado pelo espaço que ocupo no seu coração.

Aos meus irmãos, João Pedro e Regina, por serem os melhores irmãos que alguma vez podia pedir. Pelo carinho que tiveram para comigo desde sempre. Por apoiarem os meus sonhos. Por permitirem contar com eles para qualquer coisa e a qualquer altura.

Ao Dr. Virgílio Correia, ao Dr. Humberto Ferreira e a todas as entidades responsáveis por Conímbriga, pela iniciativa deste projeto e por me permitirem auxiliar na sua concretização. Pela disponibilidade, hospitalidade e apoio excecionais.

Ao meu sogro, João, pela sua paciência e disponibilidade no auxílio de diversas tarefas, e à minha sogra, Cláudia, por me acolher durante esta etapa e fazer-me sempre sentir em casa.

Por último, agradeço a toda a minha família e amigos que sempre acreditaram em mim e me apoiaram de uma maneira ou de outra.

A todos vós, o meu sincero OBRIGADO!

## Resumo

A Realidade Aumentada é uma tecnologia que, apesar de existir há várias décadas, só presentemente é que manifesta o seu enorme potencial. Esta revelação surge numa era de ouro para as novas tecnologias, nomeadamente os *smartphones*, que têm apresentado uma progressão considerável ao longo dos últimos anos. Atualmente, estes dispositivos móveis são praticamente indispensáveis para nós. Além de funcionalidades básicas como telefonemas, mensagens de texto e acesso à Internet, os *smartphones* são capazes de executar as mais diversas aplicações computacionais, graças à robustez do seu software e do seu hardware. Paralelamente a esta evolução, a tecnologia de Realidade Aumentada tem crescido significativamente nos últimos anos e a sua presença em aplicações móveis é cada vez mais uma realidade. O futuro parece bastante promissor, mas com ele surgem alguns obstáculos que necessitam de ser ultrapassados.

Atualmente, existe uma grande variedade de SDKs concebidos para a integração de Realidade Aumentada em aplicações móveis, tanto à base de marcadores artificiais como à base de características naturais. Esta última técnica, no entanto, ainda apresenta algumas limitações, uma vez que a tecnologia de Realidade Aumentada continua otimizada essencialmente para ambientes fechados e/ou controlados. Já em espaços abertos, o desempenho desta tecnologia poderá ser afetado por fatores externos como a luminosidade irregular e a presença de sombras. Tendo isso em conta, o local de estudo definido para este projeto foi Conímbriga, e os seus mosaicos romanos foram escolhidos como alvos da tecnologia de Realidade Aumentada. Um dos objetivos principais desta dissertação passa, portanto, pela pesquisa, recolha e análise dos SDKs de Realidade Aumentada existentes atualmente, de forma a encontrar o(s) mais adequado(s) para utilizar sobre os mosaicos romanos de Conímbriga.

No âmbito do projeto MosaicoLab, foi proposta a criação de um protótipo de uma aplicação móvel que aplicasse a tecnologia de Realidade Aumentada sobre os mosaicos romanos de Conímbriga. Esta proposta visa a preservação e dinamização do Património Cultural constituído por estes mosaicos romanos, que, apesar de serem uma forma de arte antiga, continuam a ser uma expressão de arte magnífica nos dias que correm e, como todo o tipo de Património Cultural, necessitam de ser salvaguardados. Além disso, a Realidade Aumentada vem proporcionar uma experiência visual e culturalmente enriquecedora às ruínas de Conímbriga, e, no futuro, certamente irá marcar a presença, cada vez mais, em espaços de Património Cultural.

Após uma análise feita aos SDKs, o leque de opções ficou reduzido a estes três: *CraftAR*, *PixLive* e *Wikitude*. Após a realização de testes sobre este conjunto de SDKs, o *CraftAR* apresentou melhores resultados. Apesar disso, este acabou por ser descontinuado, sendo que, no final, a escolha recaiu sobre o SDK da *Wikitude* que, por sua vez, é uma ferramenta bastante completa. Ao longo desta dissertação, foi feito o planeamento, calendarização e gestão de todas as tarefas, assim como uma análise do estado da arte relativamente aos conceitos predominantes desta dissertação. Foram ainda definidas as metodologias, os requisitos e a arquitetura, e realizados os testes necessários para a produção de um protótipo sólido da aplicação móvel. Por fim, foi também criada uma interface web, de forma a ser possível gerir os conteúdos da aplicação móvel.

## Palavras-Chave

“Realidade Aumentada”, “SDK”, “Património Cultural”, “Mosaicos Romanos”, “Aplicação Móvel Híbrida”



## Abstract

Augmented Reality is a technology that, although it has existed for several decades, is only now showing its enormous potential. This revelation emerges in a golden age for new technologies, namely smartphones, which have shown a steady progression over the last few years. Nowadays, these mobile devices are practically indispensable for us. In addition to basic features, such as phone calls, text messages and Internet access, smartphones are able to run the most diverse computing applications, thanks to the robustness of their software and hardware. Parallel to this evolution, the technology of Augmented Reality has grown significantly in recent years and its presence in mobile applications is increasingly becoming a reality. The future looks promising, but with it come some obstacles that need to be surpassed.

Currently, there is a wide variety of SDKs designed for the integration of Augmented Reality into mobile applications, based on artificial markers and based on natural features. The latter technique, however, still has some limitations, since Augmented Reality technology remains essentially optimized for closed and/or controlled environments. In open spaces, the performance of this technology may be affected by external factors such as irregular luminosity and the presence of shadows. Taking this into account, the place of study defined for this project was Conímbriga, and its Roman mosaics were chosen as targets of Augmented Reality's technology. Therefore, one of the main goals of this dissertation is to research, collect and analyze existing Augmented Reality SDKs, in order to find the most suitable ones to use on the Roman mosaics of Conímbriga.

Within the scope of the MosaicoLab project, it was proposed the creation of a prototype of a mobile application that would impose the Augmented Reality technology on the Roman mosaics of Conímbriga. This proposal aims at the preservation and dynamization of the Cultural Patrimony, constituted by these Roman mosaics, which, despite being an ancient art form, still remain as a magnificent art expression and, like every type of Cultural Heritage, need to be safeguarded. In addition, the Augmented Reality will provide a visual and culturally enriching experience to the ruins of Conímbriga and, in the future, will certainly and increasingly be part of Cultural Heritage spaces.

After an analysis of the SDKs, the range of options was reduced to three: *CraftAR*, *PixLive* and *Wikitude*. After performing tests on this set of SDKs, *CraftAR* presented better results. Despite this, it was eventually discontinued and, in the end, Wikitude's SDK was chosen which, in turn, is a fairly complete tool. Throughout the course of this dissertation, the planning, scheduling and management of all tasks were done, as well as an analysis of the state of the art, regarding to the predominant concepts of this dissertation. The methodologies, requirements and architecture were also defined, and the necessary tests were carried out in order to produce a solid prototype. Finally, a web interface was also created, enabling the management of the mobile application's contents.

## Keywords

“Augmented Reality”, “SDK”, “Cultural Patrimony”, “Roman Mosaics”, “Hybrid Mobile Application”



# Índice

<b>1. Introdução.....</b>	<b>1</b>
1.1. Enquadramento .....	1
1.2. Desafios .....	2
1.3. Objetivos .....	3
1.4. Estrutura do documento.....	4
<b>2. Análise do Estado da Arte .....</b>	<b>7</b>
2.1. Realidade Aumentada.....	7
2.2. SDKs de Realidade Aumentada.....	12
2.3. Aplicações para Património Cultural .....	17
2.3.1. Aplicações com Realidade Virtual.....	20
2.3.2. Aplicações com Realidade Aumentada.....	24
2.4. Desenvolvimento de Aplicações Multiplataforma.....	28
2.4.1. Estudo Comparativo dos Tipos de Aplicação Móvel .....	29
2.4.2. Frameworks para Aplicações Híbridas .....	31
<b>3. Planeamento e Metodologias .....</b>	<b>33</b>
3.1. Planeamento.....	33
3.2. Metodologia e Ferramentas Utilizadas.....	36
3.3. Riscos .....	40
<b>4. Testes dos SDKs.....</b>	<b>42</b>
4.1. Testes na Calçada Portuguesa .....	42
4.2. Testes em Conímbriga.....	51
<b>5. ConimbrigAR .....</b>	<b>62</b>
5.1. Requisitos .....	62
5.1.1. Atores e User Stories.....	62
5.1.2. Requisitos Funcionais e Não Funcionais .....	64
5.2. Arquitetura do Sistema.....	68
5.3. Modelo de Dados.....	68
5.4. Descrição do Trabalho Desenvolvido .....	72
5.4.1. <i>Backend</i> .....	72
5.4.2. Interface Web.....	75
5.4.3. Aplicação Móvel.....	80
5.5. Protótipo Final.....	82

5.6. Testes de Usabilidade .....	87
<b>6. Conclusão e Trabalho Futuro.....</b>	<b>90</b>
<b>7. Referências.....</b>	<b>91</b>
<b>Anexos.....</b>	<b>96</b>
A: Diagramas de Gantt do segundo semestre .....	96
B: Listagem dos Requisitos Funcionais .....	98
C: Formulários dos testes de usabilidade .....	110
C1: Questões pré-testes.....	110
C2: Questões pós-testes .....	114
D: Vídeos dos testes realizados.....	118
D1: Testes aos SDKs na Calçada Portuguesa.....	118
D2: Testes aos SDKs em Conímbriga.....	118
D3: Testes de Usabilidade.....	118



## Lista de Figuras:

<b>Figura 1:</b> Representação do RVC, com exemplos ilustrativos. (Adaptado) (Noh & Sunar, 2009).....	8
<b>Figura 2:</b> Diagrama representativo das diferentes técnicas de rastreamento. (Adaptado) (Billinghurst et al., 2015) .....	9
<b>Figura 3:</b> Diferentes tipos de marcadores (em cima) suportadas pela framework Studierstube ES, com a respetiva experiência de RA reproduzida (em baixo). (Schmalstieg, Schmalstieg, Wagner, & Wagner, 2008) .....	11
<b>Figura 4:</b> Exemplo ilustrativo da aplicação de RA criada pela IKEA, que deteta as características do ambiente envolvente de modo a posicionar a mobília virtual. (Pardes, 2017) .....	11
<b>Figura 5:</b> Exemplo da aplicação do rastreamento com base num modelo 3D. (Petit, 2013)	12
<b>Figura 6:</b> Estado do anfiteatro romano de Byblos, nos dias que correm. (Monico, 2016)....	20
<b>Figura 7:</b> Exemplo ilustrativo do anfiteatro de Byblos 3D, durante a experiência de RV. (Younes et al., 2017) .....	21
<b>Figura 8:</b> Exemplo ilustrativo da experiência de RV para o Museu de Mosul. (Drum, 2016) .....	22
<b>Figura 9:</b> Pintura “The Angelus” (1857-1859), por Jean-François Millet.....	23
<b>Figura 10:</b> Imagem retirada do trailer da experiência Dreams of Dalí, que mostra o novo mundo pelos olhos do utilizador. (Silverstein, 2016).....	23
<b>Figura 11:</b> Imagem retirada do trailer da experiência Dreams of Dalí, que mostra o utilizador durante a experiência e com esta exibida num monitor ao seu lado. (Silverstein, 2016).....	24
<b>Figura 12:</b> Exemplo ilustrativo da interface do ARCHEOGUIDE, com uma imagem in situ e uma imagem da perspetiva do utilizador. (Archeomaticamagazine, 2013) .....	25
<b>Figura 13:</b> Reconstrução de um monumento (imagem esquerda) através de RA (imagem direita). (Vlahakis et al., 2001) .....	25
<b>Figura 14:</b> Imagem-alvo usada pela aplicação. (News, 2015).....	26
<b>Figura 15:</b> Imagens ilustrativas do funcionamento da aplicação: Imagem-alvo detetada (imagem esquerda); Processo de reconstrução 3D completo (imagem direita). (Scopigno et al., 2015).....	26
<b>Figura 16:</b> Reconstrução de um vaso (imagem esquerda) e projeção de um guia virtual (imagem direita), através de RA. (Martins, 2016) .....	27
<b>Figura 17:</b> Exemplo ilustrativo de uma rota, no modo exterior da aplicação. (Martins, 2016) .....	28
<b>Figura 18:</b> Diagrama de Gantt para o segundo semestre.....	35
<b>Figura 19:</b> Porta Férrea de Coimbra (esquerda) e ilustrações da calçada portuguesa (direita). .....	43

<b>Figura 20:</b> (esquerda para a direita) Targets “Rainha”, “Mocho”, “Pergaminho” e “Guitarra”.	43
<b>Figura 21:</b> Target “Estrela”	43
<b>Figura 22:</b> (esquerda para a direita) Imagens virtuais “Posição”, “Contorno” e “Preenchimento”.	44
<b>Figura 23:</b> Resultados esperados para a sobreposição dos overlays “Posição”, em todos os targets.	45
<b>Figura 24:</b> Resultados esperados para a sobreposição dos overlays “Contorno”, em todos os targets.	45
<b>Figura 25:</b> Resultados esperados para a sobreposição dos overlays “Preenchimento”, em todos os targets.	46
<b>Figura 26:</b> Ecrã inicial do protótipo desenvolvido com o SDK da Wikitude.	46
<b>Figura 27:</b> Exemplo ilustrativo do Teste #1, com o antes (imagem à esquerda) e o depois (imagem à direita).	47
<b>Figura 28:</b> Exemplo ilustrativo do Teste #2, com o antes (imagem à esquerda) e o depois (imagem à direita).	48
<b>Figura 29:</b> Exemplo ilustrativo do Teste #3, com o antes (imagem à esquerda) e o depois (imagem à direita).	48
<b>Figura 30:</b> Exemplo ilustrativo do Teste #4, com o antes (imagem à esquerda) e o depois (imagem à direita).	49
<b>Figura 31:</b> Exemplo ilustrativo do Teste #5, com o antes (imagem à esquerda) e o depois (imagem à direita).	50
<b>Figura 32:</b> Exemplo ilustrativo do Teste #6, com o antes (imagem à esquerda) e o depois (imagem à direita).	50
<b>Figura 33:</b> Aquisição fotográfica dos mosaicos romanos.	52
<b>Figura 34:</b> Exemplo do sistema de avaliação de targets da Wikitude (imagem à esquerda) e do CraftAR (imagem à direita).	53
<b>Figura 35:</b> Targets escolhidos para a realização dos testes aos SDKs em Conímbriga.	53
<b>Figura 36:</b> Exemplo de um overlay (imagem à esquerda) e do respetivo target com esse overlay sobreposto (imagem à direita).	54
<b>Figura 37:</b> Arquitetura do sistema	68
<b>Figura 38:</b> Diagrama de entidade e relacionamentos	69
<b>Figura 39:</b> Exemplo de um objeto do tipo Mosaico	70
<b>Figura 40:</b> Exemplo de um objeto do tipo Grupo_de_Mosaicos.	70
<b>Figura 41:</b> Exemplo de um objeto do tipo Target	71
<b>Figura 42:</b> Exemplo de um objeto do tipo Overlay	71

<b>Figura 43:</b> Exemplo de um objeto do tipo Camada_de_Informacao .....	71
<b>Figura 44:</b> Exemplo de um objeto do tipo Overlay_Transformations .....	72
<b>Figura 45:</b> Diagrama de fluxo do processo de autenticação com JWT (Nascimento, 2018)	75
<b>Figura 46:</b> Interface da "Browsable API" .....	75
<b>Figura 47:</b> Janela de login.....	76
<b>Figura 48:</b> Menu principal da interface web .....	77
<b>Figura 49:</b> Opção Ver Todos - Listagem dos mosaicos (I).....	77
<b>Figura 50:</b> Opção Ver Todos - Listagem dos mosaicos (II) .....	78
<b>Figura 51:</b> Opção Ver Todos - Listagem dos mosaicos (III).....	78
<b>Figura 52:</b> Opção Gerir um Target - Escolha do target a ser gerido .....	79
<b>Figura 53:</b> Opção Gerir um Target - Edição de um target (I).....	79
<b>Figura 54:</b> Opção Gerir um Target - Edição de um target (II) .....	80
<b>Figura 55:</b> Opção Adicionar um novo – Criação de um objeto do tipo Target.....	80
<b>Figura 56:</b> Janela inicial da aplicação móvel.....	83
<b>Figura 57:</b> Janela da listagem de mosaicos (imagem à esquerda), de grupos (imagem central) e de camadas (imagem à direita) .....	83
<b>Figura 58:</b> Janela das informações sobre um mosaico .....	84
<b>Figura 59:</b> Janela das informações sobre um grupo .....	85
<b>Figura 60:</b> Janela das informações sobre uma camada.....	85
<b>Figura 61:</b> Janela da experiência de RA, sem qualquer target detetado .....	86
<b>Figura 62:</b> Janela da experiência de RA, com overlays de diversas camadas .....	86
<b>Figura 63:</b> Janela da experiência de RA, com as mensagens de aviso do sistema. ....	87

## **Lista de Tabelas:**

<b>Tabela 1:</b> Características dos diferentes SDKs de RA existentes atualmente. ....	16
<b>Tabela 2:</b> Comparação entre os diferentes tipos de aplicações móveis. (Abed, 2016; Alexseyenk, 2017; BloomIdea, 2016; Bristowe, 2015; Patil, 2016; Saccomani, 2017).....	30
<b>Tabela 3:</b> SDKs descartados.....	37
<b>Tabela 4:</b> SDKs aceites.....	37
<b>Tabela 5:</b> Risco #1 .....	40
<b>Tabela 6:</b> Risco #2.....	41
<b>Tabela 7:</b> Resultados do Teste #1.....	47
<b>Tabela 8:</b> Resultados do Teste #2.....	48
<b>Tabela 9:</b> Resultados do Teste #3.....	49
<b>Tabela 10:</b> Resultados do Teste #4.....	49
<b>Tabela 11:</b> Resultados do Teste #5.....	50
<b>Tabela 12:</b> Resultados do Teste #6.....	50
<b>Tabela 13:</b> Avaliações dos targets em ambas as interfaces .....	54
<b>Tabela 14:</b> Resultados da capacidade de deteção dos SDKs.....	58
<b>Tabela 15:</b> Resultados do atraso da deteção dos targets .....	59
<b>Tabela 16:</b> Resultados do atraso da deteção dos targets .....	59
<b>Tabela 17:</b> Resultados do atraso da deteção dos targets .....	60
<b>Tabela 18:</b> Resultados do atraso da deteção dos targets .....	60
<b>Tabela 19:</b> Template para os requisitos funcionais.....	65
<b>Tabela 20:</b> Template para os requisitos não funcionais.....	67
<b>Tabela 21:</b> Endpoints da API REST.....	74
<b>Tabela 22:</b> Inquérito pré-testes .....	88



# 1. Introdução

Atualmente, encontramos-nos na era digital, onde surgem tecnologias cada vez mais robustas e versáteis, facilitando o desempenho do ser humano em diversas tarefas do seu quotidiano. Em virtude desta evolução tecnológica, o mundo do trabalho corrente valoriza mais aspetos como a criatividade e a conceção de algo único e proveitoso para a sociedade, pelo que muitas empresas terão que se reinventar, inovar e ajustar-se às tendências do mercado. Um dos atuais surtos tecnológicos, com grande potencial e futuro promissor, é a Realidade Aumentada (RA). Embora a investigação realizada nesta área contenha já mais de 50 anos, é precisamente nesta fase rica em tecnologia e próspera para inovações, que a RA resurge com primazia, desta vez com o foco em aplicações destinadas ao público em geral. Um exemplo claro deste ressurgimento da RA é evidenciado na aplicação móvel Pokémon Go<sup>1</sup> (2016), que cativou milhões de pessoas de todo o mundo, não só pela popularidade da franquia Pokémon e pela devoção que imensos fãs possuem ainda hoje pela mesma, mas também pela experiência de RA atrativa que esta aplicação proporciona. Os milhões de utilizadores são motivados a explorar o mundo real, incluindo pontos de interesse dos locais onde se encontravam, na esperança de encontrar as criaturas virtuais que surgiam espontaneamente no ambiente que os rodeava. Esta aplicação conseguiu projetar a tecnologia de RA a uma escala global, enriquecendo, no processo, a interação entre as pessoas e o mundo real.

A RA envolve o utilizador numa experiência enriquecedora, na medida em que a perceção e interação com o mundo real são ampliadas, devido à incorporação e justaposição de objetos virtuais no mesmo. É de extrema importância que haja um alinhamento preciso, tanto a nível da posição como da orientação, da informação virtual com a informação real, de modo a que o utilizador não distinga à primeira vista os objetos virtuais e dando a sensação de um só ambiente. Para que seja possível a observação dos dois mundos simultaneamente, o utilizador necessita de um dispositivo particular que exponha a informação virtual juntamente com o ambiente real onde se encontra. Existem vários tipos de opções para a visualização de sistemas de RA, desde a projeção por vídeo até ao uso de acessórios particulares como os óculos de RA. No entanto, devido à evolução impetuosa dos *smartphones*, estes apresentam um *hardware* benéfico ao desenvolvimento de aplicações de RA o que, juntando ao fato de serem práticos e intuitivos de utilizar, torna-os a escolha predileta nos dias que correm.

## 1.1. Enquadramento

Após longos anos de investigação e de um notável progresso tecnológico, a RA apresenta um potencial em crescendo, sendo que existe um leque inimaginável de ideias onde esta poderá ser aplicada e eventualmente contribuir para uma melhor qualidade de vida dos seus usufruidores. Inicialmente, a RA era explorada apenas em certas áreas como a medicina, indústria, serviços militares e robótica. Uma vez que o foco do desenvolvimento de aplicações de RA se inseria exclusivamente nestes campos, apenas as pessoas que trabalhavam nos mesmos usufruíam deste tipo de experiências, ou seja, o público não tinha ainda qualquer oportunidade de as vivenciar (Brogni, Avizzano, Evangelista, & Bergamasco, 1999). Desde então, até aos dias que correm, esta tendência inverteu-se. Existe agora um primado interesse de levar a RA ao público e conquistá-lo como consumidor, levando, conseqüentemente, ao surgimento de novos campos de aplicação da RA como a educação, jogos, publicidade, design, etc. Não obstante da popularidade destas áreas

---

<sup>1</sup> <https://www.pokemongo.com/pt-pt/>

recentes, existe uma outra área onde a RA promete ter grande valor e impacto, caso sejam ultrapassadas todas as barreiras que esta ainda apresenta na sua implementação. Essa área denomina-se por Património Cultural e é precisamente uma das principais motivações desta dissertação.

A essência do Património Cultural está na preservação de artefactos, monumentos, locais e qualquer outro tipo de elemento que tenha significância histórica e relevância cultural. Infelizmente, existem imensos marcos impressionantes do nosso passado com um valor histórico incalculável, que se encontram danificados, em risco de desaparecer ou já não existem de todo. Tendo isto em conta, a integridade dos marcos que ainda resistem está em constante ameaça por parte de negligência humana (turismo descomedido, poluição, furto, etc.) ou catástrofes naturais, pelo que surge a necessidade urgente de arranjar soluções para os resguardar. A RA surge como uma solução viável e poderá complementar perfeitamente o Património Cultural, proporcionando inclusive uma experiência diferente aos seus consumidores. Apesar disso, existem ainda fatores como a luminosidade variante ou texturas consideravelmente irregulares, que tornam a implementação de RA uma tarefa algo atribulada em locais de Património Cultural. Na próxima secção, este desafio será descrito mais pormenorizadamente.

O local de interesse que irá ser explorado no âmbito deste projeto será a cidade romana de Conímbriga<sup>2</sup>. Ilustre pelas suas ruínas e artefactos ancestrais, trata-se de um dos maiores sítios arqueológicos em Portugal, sendo inclusive considerado um Monumento Nacional e, assim sendo, encaixa-se perfeitamente no conceito de Património Cultural. Apesar de existir alguma diversidade de artefactos e monumentos neste local, o foco desta investigação recairá apenas sobre os mosaicos romanos. De forma sucinta, este projeto visa a dinamização destes mosaicos, utilizando a tecnologia de RA para colocar imagens virtuais sobre os mesmos. De modo a usufruírem desta experiência, os utilizadores terão apenas que utilizar os seus *smartphones*, executar a respetiva aplicação móvel, e apontar para os mosaicos que desejam observar. Embora existam outros dispositivos de visualização mais robustos, como os óculos de RA, os *smartphones* não implicam o custo elevado que esses equipamentos requerem, não causam um eventual desconforto na sua utilização e, atualmente, praticamente qualquer pessoa tem um em sua posse.

Por fim, importa realçar que existem diversas técnicas de rastreamento que podem ser utilizadas na implementação de sistemas de RA e que serão exploradas no próximo capítulo. A técnica preferencial recai para uma que não interfira fisicamente com o espaço envolvente, pelo que a técnica que envolve a utilização de marcadores artificiais foi excluída *a priori*. Esta decisão e as metodologias adotadas serão discutidas mais à frente neste documento. Na próxima secção serão descritos os principais desafios propostos para esta dissertação.

## 1.2. Desafios

Hoje em dia, ainda persistem alguns obstáculos na implementação de sistemas de RA, impedindo esta tecnologia de atingir o máximo do seu potencial. Um dos principais desafios aquando a implementação de sistemas de RA é o alinhamento preciso, tanto da posição como da orientação, da informação virtual com o ambiente real onde é misturada. Não havendo um registo rigoroso, a experiência de RA não terá sucesso logo à partida. É possível ultrapassar este desafio, desde que a experiência seja destinada a um ambiente fechado e controlado com as condições necessárias (iluminação apropriada, rastreamento minucioso, etc.) ou utilizando a técnica de RA à base de marcadores artificiais, que apresenta um alto grau de eficiência. No entanto, a verdadeira adversidade ocorre em ambientes exteriores (*i.e.*, ao ar livre) e utilizando a técnica de RA baseada nas características naturais dos objetos.

---

<sup>2</sup> <http://www.conimbriga.gov.pt/>

Em espaços exteriores, é bastante difícil controlar as condições e adversidades encontradas, devido à sua irregularidade e imprevisibilidade. Um dos fatores mais delicados, senão o mais, é a luminosidade. Sendo que é possível estimar a posição do sol, dependendo das horas do dia, o mesmo não poderá ser dito acerca do surgimento de nuvens que o bloqueiam, ou até mesmo de nevoeiro. A presença de luminosidade irregular torna-se um entrave significativo nos processos de alinhamento e rastreamento, não existindo ainda uma solução que o resolva na íntegra. Uma vez que o local escolhido para a realização desta investigação se trata precisamente de um espaço ao ar livre, esta adversidade da iluminação foi efetivamente constatada e, portanto, trata-se de um desafio a ter em conta na presente investigação.

Outro desafio considerado para esta investigação, relaciona-se com o tipo de objetos nos quais se pretende aplicar a RA. O desenvolvimento de *Software Development Kits* (SDKs) comerciais de RA foi aprimorado para a aplicação em objetos e superfícies simples e essencialmente planas (*e.g.*, folha de papel, mesa, etc.). Tendo isso em conta, objetos com uma composição mais heterogénea e texturas consideravelmente complexas, como é o caso dos mosaicos romanos, serão uma incógnita no que toca ao nível de responsividade face à tecnologia de RA. A deterioração de alguns destes mosaicos ao longo do tempo só irá elevar o grau de dificuldade de deteção, devido à sua inconsistência e perda de cor. Assim sendo, é também considerado como desafio desta investigação a aplicação de RA em mosaicos romanos, tidos como objetos ou superfícies inortodoxas, *i.e.*, não foram destinados ou preparados para a implementação de sistemas de RA, acrescentando a considerável complexidade da sua composição.

Idealmente e a longo prazo, a implementação e aplicação sistemas de RA deve ser uma realidade em qualquer lado e em qualquer ambiente. Apesar de não existirem ainda soluções para os problemas mencionados anteriormente, existem atualmente técnicas e ferramentas disponíveis que serão úteis na colmatação e minimização desses problemas. Mais à frente neste documento, serão apresentadas algumas destas alternativas.

### 1.3. Objetivos

Esta dissertação tem como objetivos principais a dinamização dos mosaicos romanos de Conímbriga, partindo da iniciativa MosaicoLab<sup>3</sup>, e a investigação dos SDKs de RA existentes atualmente que sejam integrados no desenvolvimento de aplicações móveis híbridas. Juntamente com a avaliação das condições encontradas no local, serão identificados os principais obstáculos detetados e antevistos como desafios. Em suma, é proposta a implementação de RA nos mosaicos romanos, tendo em conta as circunstâncias envolventes e os desafios apresentados, avaliando, no processo, o *feedback* do sistema face à deteção e rastreamento desses mosaicos. Deste modo, são encontradas as melhores soluções existentes para a implementação de RA em aplicações móveis híbridas e será criado um protótipo utilizando a solução mais adequada. É fundamental realçar que existe um interesse acrescido de a aplicação ser de natureza híbrida, de modo a não limitar a sua compatibilidade a apenas uma plataforma (Android, iOS, etc.) e a criar também um processo de desenvolvimento mais simples, uma vez que são apenas utilizadas tecnologias web (JavaScript, HTML, CSS) em vez de várias linguagens distintas, específicas para cada plataforma.

No final desta dissertação, é esperado que o protótipo criado seja utilizado e testado por visitantes das ruínas de Conímbriga, que não tenham qualquer ligação ou conhecimento prévio deste projeto, e aplicado sobre os mosaicos romanos. Tendo isso em conta, é pretendido avaliar o comportamento da aplicação no momento em que esta é executada pelos visitantes, esperando obter a mínima margem de erros possível, e tentar perceber de que forma é que estes visitantes encaram este tipo de experiências, inseridas num contexto cultural. Além disso, é também

---

<sup>3</sup> <http://mosaicolab.pt/pt/>

esperado reunir um conjunto legítimo de SDKs de RA que possam ser utilizados no desenvolvimento de aplicações móveis híbridas.

Concluindo, em termos práticos, o produto final esperado trata-se de um protótipo de uma aplicação móvel híbrida, que irá produzir uma experiência de RA sobre os mosaicos romanos das ruínas de Conímbriga. O público-alvo é definido como qualquer visitante das ruínas de Conímbriga, que possua um *smartphone*. É importante, portanto, que haja uma boa diversidade de experiências de RA nos mosaicos romanos, de forma a cativar utilizadores de diferentes faixas etárias, motivações e expectativas para com a visita, espíritos criativos, etc. Por forma a controlar todos os conteúdos da aplicação, foi proposta também a criação uma interface web.

Resumindo, os objetivos delineados para este projeto são os seguintes:

1. Estudo dos SDKs existentes para a aplicação da tecnologia de RA em aplicações móveis híbridas, escolhendo posteriormente as mais adequadas ao projeto;
2. Teste dos SDKs levantados no objetivo 1, escolhendo posteriormente a que apresenta melhores resultados;
3. Desenvolvimento do protótipo de uma aplicação móvel híbrida que implemente a tecnologia de RA sobre os mosaicos romanos de Conímbriga, tendo em conta o objetivo 1 e o objetivo 2;
4. Criação de uma interface web para a gestão dos conteúdos da aplicação móvel;
5. Avaliação do desempenho do protótipo, através de testes de usabilidade realizados por um conjunto de visitantes às ruínas de Conímbriga.

## 1.4. Estrutura do documento

O presente documento encontra-se dividido nos seguintes capítulos:

### 1. Introdução

É feita uma descrição desta Dissertação, incluindo o seu enquadramento, desafios e objetivos propostos.

### 2. Análise do Estado da Arte

É apresentada a pesquisa relacionada com a tecnologia de Realidade Aumentada e os SDKs de Realidade Aumentada, de forma a avaliar a oferta atual destas ferramentas no mercado. Além disso, serão também apresentados exemplos de aplicações desenvolvidas no âmbito do Património Cultural, integrando as tecnologias de Realidade Virtual ou de Realidade Aumentada. Por fim, é apresentado um breve estudo ao desenvolvimento de aplicações híbridas, apresentado uma comparação entre os diversos tipos de aplicações móveis e exemplos de algumas *frameworks* existentes para o desenvolvimento híbrido.

### 3. Planeamento e Metodologias

É apresentado o planeamento, metodologias, ferramentas e tecnológicas utilizadas ao longo do projeto, dividido em dois semestres. É também apresentada a análise dos riscos considerados.

### 4. Testes aos SDKs

São descritos os testes realizados aos SDKs de RA, no primeiro e no segundo semestre, assim como os seus respectivos resultados.

## **5. ConímbrigAR**

São apresentados os requisitos funcionais e não funcionais do sistema, assim como a sua arquitetura e o modelo de dados criado. É também descrito o funcionamento das componentes de *Frontend* e *Backend* da aplicação móvel e interface web e, por fim, é apresentado o protótipo final e os respetivos testes de usabilidade.

## **6. Conclusão e Trabalho Futuro**

É feita uma retrospectiva de todo o trabalho realizado nesta Dissertação e é proposto o trabalho futuro.

## **7. Referências**

São listadas todas as referências utilizadas nesta Dissertação.

Por fim, são apresentados os anexos do documento.



## 2. Análise do Estado da Arte

### 2.1. Realidade Aumentada

A Realidade Aumentada (RA) pode ser definida como um sistema onde, num ambiente real, são integrados objetos virtuais tridimensionais (3D), de forma coerente e em tempo real (Azuma, 1997). Os objetos virtuais devem misturar-se perfeitamente com os objetos reais, dando a percepção de que coexistem num só ambiente, e interagir com o utilizador e o ambiente real de forma natural. De maneira a que tudo isto seja possível, é essencial que haja um alinhamento adequado e preciso da informação virtual com a informação real (Hoff, Nguyen, & Lyon, 1996; Vallino, 1998). De modo a complementar estas descrições, naquela que é considerada como uma das definições globalmente aceites sobre RA, o investigador Ronald Azuma definiu três requisitos fundamentais para a constituição de um sistema de RA (Azuma, 1997):

- Combinação de informação virtual com informação real;
- Interação em tempo real;
- Alinhamento em 3D.

Com base nesta definição, é possível retirar algumas ilações que elucidam melhor o conceito de RA. Em primeiro lugar, não existem limitações quanto à tecnologia de visualização a ser utilizada, facto que acontecia antes do surgimento desta definição, onde alguns investigadores apenas mencionavam os *Head-Mounted Displays* (HMDs) como ferramenta de visualização das experiências de RA. Em segundo lugar, as experiências de RA não são especificadas somente para estimular a visão do utilizador, podendo, eventual e potencialmente, serem criadas experiências que estimulem também outros sentidos do ser humano, como a audição (Damiani, 2017), o tato e o olfato (Naimark, 2016). Por fim, terá de ser efetuado um alinhamento em 3D dos objetos virtuais com o mundo real. Este último requisito sugere duas possibilidades para a aplicação de RA, sendo uma delas a mais óbvia, *i.e.*, complementar o mundo real através da incorporação de elementos virtuais. De maneira oposta, a outra possibilidade refere-se à remoção de objetos reais ao revesti-los com objetos virtuais (técnica também conhecida por Realidade Diminuída) (Billinghurst, Clark, & Lee, 2015; Krevelen & Poelman, 2010).

A RA pode ser vista ainda como um complemento da Realidade Virtual (RV), sendo que esta última pretende criar um ambiente sintético e imersivo, isto é, o utilizador é totalmente isolado do mundo real, no que diz respeito ao seu campo de visão. Por outro lado, a RA permite ao utilizador ainda visualizar o mundo real, apenas com a sobreposição de objetos virtuais e, desta forma, suplementar a realidade, ao invés de a substituir por completo (Azuma, 1997). A RA apresenta um potencial cada vez mais superior ao da RV, uma vez que aprimora o conhecimento e interação do utilizador com o mundo real, unindo-o a todos os elementos do ambiente que o rodeia (pessoas, locais, objetos) em vez de o remover por completo do mesmo (Azuma, 2017). Estas duas tecnologias podem ser encontradas e definidas também segundo uma taxionomia que representa a diversidade de ambientes possíveis de gerar, através da combinação e variação das informações do tipo real e virtual, denominada de *Reality-Virtuality Continuum* (RVC) (Milgram, Takemura, Utsumi, & Kishino, 1995), ilustrada na Fig. 1.



**Figura 1:** Representação do RVC, com exemplos ilustrativos. (Adaptado) (Noh & Sunar, 2009)

Na extremidade esquerda do RVC (Fig. 1), encontram-se ambientes que contenham apenas objetos reais (Ambiente Real) e todo o tipo de informação também observada num cenário do mundo real, diretamente em pessoa. Já na extremidade direita, o ambiente real será substituído na íntegra por um ambiente sintético (Ambiente Virtual), onde toda a informação é também virtual. Entre estas extremidades, é definida uma área designada por Realidade Mista, onde são encontradas todas as combinações possíveis de informação real com informação virtual, gerando, portanto, ambientes mistos. Perto da extremidade do Ambiente Virtual, apesar de o ambiente ser predominantemente sintético, será também possível observar elementos reais incorporados no mesmo (Virtualidade Aumentada). Finalmente, perto da extremidade do Ambiente Real, encontram-se os ambientes de RA sendo que, dependendo da quantidade de informação virtual adicionada ou removida, será possível percorrer o RVC e encontrar inúmeros e diversificados ambientes com RA (Billinghurst et al., 2015).

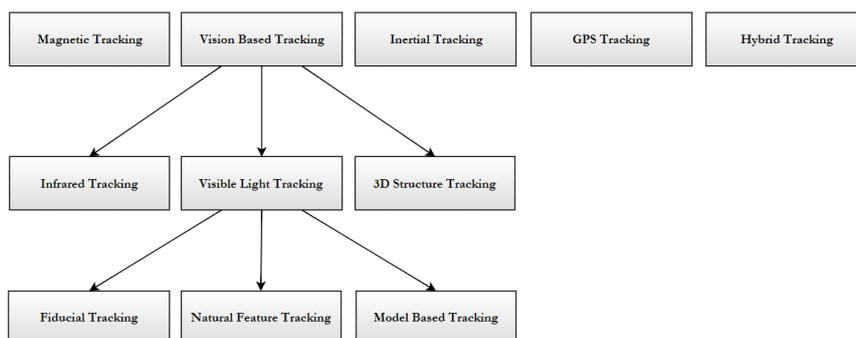
Apesar de o conceito de RA ter sido consolidado e popularizado na década de 1990, este era idealizado em 1965, por Ivan Sutherland, através de um sistema ao qual denominou por *Ultimate Display* (Watson & Luebke, 2005). Este sistema, simbolizado por uma sala, permitia ao utilizador interagir com a matéria (*i.e.*, os objetos virtuais) gerada pelo computador. Esta visão de Sutherland viria a impulsionar as investigações na área da RA pelo que, em 1968, com o auxílio de uma vasta equipa, Sutherland criou o primeiro protótipo de um sistema de RA, denominado por *Head-Mounted Display* (Sutherland, 1968). Este sistema era composto por umas lentes particulares, anexadas a um suporte fixo no teto que faria as ligações ao *hardware* responsável pela execução da experiência de RA. Os gráficos eram gerados por computador e constituídos por *wireframes*. Apesar de se tratar de um sistema primitivo, este já possuía a capacidade de combinar as componentes de visualização, rastreamento e computação, necessárias para a criação e incorporação de objetos 3D no mundo real. Este marco na história da RA embalou o futuro da RA, dando lugar a inúmeras investigações e tecnologias notáveis. Em suma, a história da RA pode ser dividida em quatro períodos (Billinghurst et al., 2015):

- Período da Experimentação (situado antes da década de 1980): Época de experimentação, definição do conceito de Realidade Aumentada e das tecnologias necessárias;
- Período de Investigação Básica (situado entre a década de 1980 e meados da década de 1990): Época de investigação que permitiu o surgimento de tecnologias de rastreamento, visualização e dispositivos para as aplicar;
- Período das Ferramentas e Aplicações (situado entre meados da década de 1990 e 2007): Época de utilização de tecnologias com RA como apoio ao desenvolvimento de outras tecnologias e para explorar técnicas de interação, usabilidade e design;
- Período das Aplicações Comerciais (situado entre 2007 e o presente): Época de difusão global de sistemas de RA, com o foco em aplicações móveis e em áreas como a medicina, jogos e marketing.

Atualmente, encontramos-nos no período das aplicações comerciais e a portabilidade está-se a tornar cada vez uma prioridade no desenvolvimento de aplicações de RA. Apesar de ser possível criar e visualizar experiências de RA em computadores que possuam uma *webcam* (também conhecido por RA baseada em *browser*) (Mullen, 2011), o interesse principal está no desenvolvimento de aplicações de RA para *smartphones*. Além da sua ubiquidade (*i.e.*, no mesmo dispositivo, possuem uma câmara e um ecrã de visualização) e da facilidade de transporte e manuseamento, estes dispositivos móveis possuem um hardware impressionante para o desenvolvimento de sistemas de RA, incluindo sensores (*e.g.*, GPS, acelerómetro, giroscópio) e processadores rápidos (Azuma, Billinghurst, & Klinker, 2011). Por fim, segundo a Statista<sup>4</sup>, existem cerca de 2.32 bilhões de utilizadores de *smartphones*, número que tende a crescer ao longo dos anos, incentivando a criação de aplicações de RA para estas plataformas. Um exemplo expressivo do sucesso desta abordagem encontra-se na aplicação móvel Pokémon Go, abordada anteriormente neste documento.

Apesar dos bons resultados que as aplicações de RA em dispositivos móveis têm apresentado e do seu grande potencial, Azuma afirma que serão os sistemas vestíveis a prevalecer no futuro, devido ao potencial de se obter uma visualização mais ampla, num utensílio compacto e utilizável na cabeça do utilizador (Azuma, 2017). Embora sejam razões plausíveis, o custo considerável e o acesso limitado a estes dispositivos de visualização para o público em geral, assim como o desconforto provocado devido à utilização prolongada ou a outros fatores, dependendo do utilizador, continuam a ser grandes obstáculos neste tipo de sistemas de RA. Enquanto não surgirem melhorias significativas nesses aspetos, no que toca a experiências de RA tendo o público como o consumidor alvo, a aposta irá recair espontaneamente para os *smartphones*.

Voltando à definição de Azuma, é necessário que um sistema de RA tenha a capacidade de inserir e alinhar, tridimensionalmente, a informação virtual com o mundo real (Azuma, 1997). No entanto, este procedimento torna-se impreciso sem o apoio de uma outra técnica, denominada por rastreamento (*tracking*). De modo a incorporar um objeto virtual no ambiente real, é necessário que exista algum sensor ou elemento, no mundo real, que sirva de referência para o sistema detetar e, posteriormente, integrar o respetivo objeto virtual. Contudo, o sistema necessitará também de conhecer e detetar a posição e orientação do utilizador, de modo gerar a informação virtual no mundo real com a sua posição e orientação devidamente ajustadas. O rastreamento será um processo contínuo da experiência de RA, uma vez que a informação virtual terá que antecipar os movimentos do utilizador (*i.e.*, o dispositivo de visualização), detetar a sua posição e orientação no momento e atualizá-la relativamente às anteriores. Idealmente, quanto menor for o número de falhas no processo de rastreamento de um sistema de RA, melhor se tornará a qualidade da interação do utilizador com a informação virtual. Posto isto, existem diversas técnicas de rastreamento, que podem ser devidamente categorizadas (Billinghurst et al., 2015) e visualizadas na Fig. 2.



**Figura 2:** Diagrama representativo das diferentes técnicas de rastreamento. (Adaptado) (Billinghurst et al., 2015)

<sup>4</sup> <https://www.statista.com/>

Embora exista uma grande diversidade de técnicas de rastreamento, atualmente, as mais utilizadas referem-se àquelas baseadas na visão por computador. Este tipo de técnicas determina a posição e orientação da câmera em uso através da informação obtida por sensores óticos e, uma vez que não requer muito *hardware*, e devido ao poder computacional significativo dos dispositivos móveis atuais (*e.g.*, *smartphones* e *tablets*), assim como sua ubiquidade, esta técnica tem adquirido uma popularidade acrescida nos tempos que correm. Dependendo do tipo de sensores óticos utilizados, existem três formas distintas de aplicar esta técnica, como é possível observar na Fig. 2. Tendo isto em conta, atualmente, o tipo de sensores óticos mais comuns refere-se àqueles que utilizam luz visível, pelo que, juntamente com o interesse em estudar e utilizar este tipo de técnicas de rastreamento no âmbito do projeto, estas serão descritas a seguir.

Na técnica de rastreamento baseado em luz visível (*Visible Light Tracking*), são utilizados sensores referentes a luz visível. Estes demonstram ser bastante úteis, na medida que podem ser aplicados simultaneamente no ambiente real e nos objetos virtuais a serem alinhados com o mesmo. Hoje em dia, estes sensores estão incorporados em dispositivos acessíveis a qualquer pessoa, como computadores portáteis, *smartphones* e *tablets*, e em dispositivos vestíveis, com a sua acessibilidade mais limitada ao público. Posto isto, esta técnica de rastreamento contém três vertentes, na qual é aplicada de forma distinta:

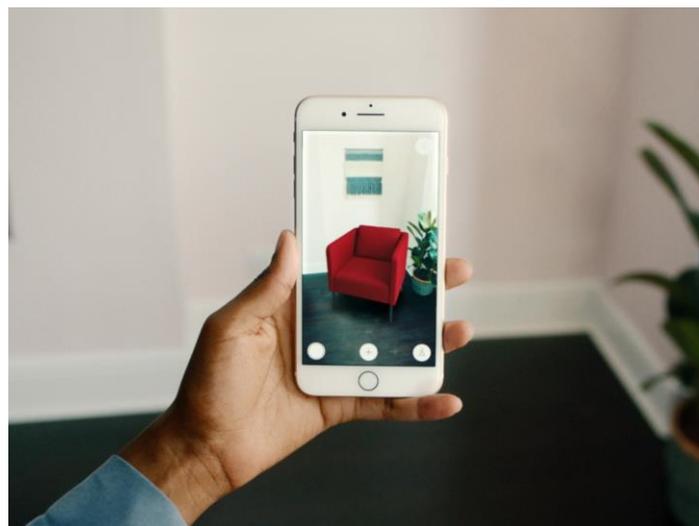
- **Rastreamento baseado em marcadores** (*Fiducial Tracking*)
- **Rastreamento baseado em características naturais** (*Natural Feature Tracking*)
- **Rastreamento baseado em modelos** (*Model Based Tracking*)

A técnica de rastreamento baseado em marcadores é conhecida por se basear no uso de marcadores artificiais (Fig. 3) que, quando são colocados no ambiente real, servem como referências no processo de alinhamento e rastreamento da informação virtual. Estes marcadores tanto podem ser passivos (*i.e.*, constituídos por um padrão facilmente distinguível do resto do ambiente) (*e.g.*, códigos QR) como ativos (*i.e.*, emitem sinais de luz ou magnéticos, captados pelos respetivos sensores) (Bostanci, Kanwal, Ehsan, & Clark, 2013). O tipo de referências utilizadas tem evoluído ao longo dos últimos anos e a complexidade destes marcadores pode variar consideravelmente, dependendo da aplicação ou tecnologia que for utilizada. Inicialmente, eram utilizados como marcadores pequenos LEDs ou pedaços de papel coloridos, que seriam detetados através da correspondência de cores (Neumann & Cho, 1996). Utilizando um número suficiente destes marcadores, a posição e a orientação da câmera eram detetadas, sendo que, durante este processo (*i.e.*, em tempo real), seria possível de adicionar marcadores e estimar a sua posição com base nos marcadores que já se encontravam no cenário (Neumann & Park, 1998). Na presença destes marcadores mais simples, havia a necessidade de estudar o espaço e projetar uma possível distribuição dos marcadores pelo mesmo, de forma a cobrir todas as zonas de interesse de visibilidade. Juntando a isto, seriam necessários, no mínimo, quatro pontos de referência definidos no local, de modo a ser possível calcular a posição e orientação do utilizador (Fischler & Bolles, 1981). Mais tarde, surgia uma alternativa que respeitava este último requisito e, ao mesmo tempo, economizava a utilização de marcadores, através da utilização de um só marcador planar e quadrilátero, com os seus quatro cantos a servirem como pontos de referência (Comport, Marchand, Pressigout, & Chaumette, 2006). Este marcador tinha ainda a capacidade de apreender informações adicionais, codificadas nela, permitindo a incorporação de diversos tipos de marcadores num só. Posto isto, esta técnica de rastreamento apresenta, ainda hoje, grande popularidade devido à simplicidade da sua utilização e à precisão que apresenta.



**Figura 3:** Diferentes tipos de marcadores (em cima) suportadas pela framework Studierstube ES, com a respetiva experiência de RA reproduzida (em baixo). (Schmalstieg, Schmalstieg, Wagner, & Wagner, 2008)

Como alternativa à utilização de marcadores no ambiente real, surge a técnica de rastreamento baseado em características naturais, uma vez que, dependendo das circunstâncias e do valor que o cenário apresenta, poderá não ser válida a aplicação de pontos de referência artificiais. Sem o auxílio de marcadores artificiais, esta técnica foca-se em procurar características naturais (*e.g.*, cantos, arestas, texturas, etc.) presentes nos objetos reais (Fig. 4). No caso das texturas, há ainda que ter em conta fatores como a suavidade, rugosidade e regularidade (Lima, Roberto, Teichrieb, & Marques, 2015). Atualmente, a metodologia mais comum utilizada nesta técnica de rastreamento consiste na captação das imagens ou objetos pretendidos a serem reconhecidos e rastreados (*i.e.*, *targets*). Por fim, através de algoritmos de processamento de imagem (*e.g.*, SIFT, SURF, BRIEF, etc.) estes marcadores naturais podem ser detetados e, posteriormente, registados no sistema através da atribuição de um “descriptor” próprio, que fará com que seja possível reconhecer o objeto correspondente de diferentes distâncias, orientações e, possivelmente, níveis de luminosidade com alguma oclusão (Amin & Govilkar, 2015). Esta técnica apresenta, geralmente, resultados mais incertos que a técnica baseada em marcadores, uma vez que dependerá de vários fatores como a qualidade das imagens (*i.e.*, marcadores naturais), das suas características e de fatores externos como a luminosidade irregular. Apesar disso, a utilização desta técnica de rastreamento não implica qualquer interferência física com o cenário real e abre todo um mundo de ideias inovadoras de aplicações com RA.



**Figura 4:** Exemplo ilustrativo da aplicação de RA criada pela IKEA, que deteta as características do ambiente envolvente de modo a posicionar a mobília virtual. (Pardes, 2017)

Por último, como o próprio nome indica, a técnica de rastreamento baseado em modelos pretende realizar o rastreamento em objetos reais através de modelos 3D criados previamente a partir dos mesmos (Fig. 5). Utilizando estes modelos, será possível manipular a sua posição e orientação, efetuando a correspondência com o objeto original. Embora atualmente os modelos 3D possam ser criados com o auxílio do computador, também denominados por CAD (*Computer-aided design*), no início, os modelos eram desenhados à mão e a sua estrutura consistia maioritariamente numa combinação de primitivas (*e.g.*, linhas, círculos, cilindros, esferas)

(Comport, Marchand, & Chaumette, 2003). De modo a criar modelos 3D com maior precisão, era utilizada uma técnica de detecção de arestas, que retirava essa informação acerca da estrutura dos objetos reais (Wuest, Vial, & Stricker, 2005). Os modelos que viriam a surgir mais tarde já incluíam texturas (Reitmayr & Drummond, 2007) e era aplicado o conceito de *keyframes* (Vacchetti, Lepetit, & Fua, 2003), permitindo assim uma robusteza acrescida em ambientes complexos e irregulares (Billinghurst et al., 2015).



Figura 5: Exemplo da aplicação do rastreamento com base num modelo 3D. (Petit, 2013)

## 2.2. SDKs de Realidade Aumentada

Atualmente, a oferta de SDKs para a integração da tecnologia de RA em aplicações móveis é vasta e bastante diversificada, tornando-se, por isso, difícil de escolher o SDK mais adequado para um projeto em mão. Tendo isso em conta, no âmbito desta dissertação, foi feita uma pesquisa elaborada dos SDKs de RA existentes atualmente. Esta investigação foi feita online e todas as informações acerca dos SDKs foram obtidas a partir de artigos de websites, de comunidades de comparação de ferramentas (*e.g.*: SocialCompare<sup>5</sup>) e dos websites oficiais dos respetivos SDKs. Por forma a conseguir um detalhamento completo de cada SDK, foram consideradas as seguintes características para analisar:

- **Licença:**
  - **Open-Source:** Todo o código-fonte do SDK encontra-se disponível ao público para visualizar, modificar e partilhar.
  - **Gratuita:** O SDK encontra-se disponível para utilizar de forma gratuita, com duração ou funcionalidades limitadas. Em alguns casos, esta licença pode ter que ser requerida à respetiva empresa e, além disso, pode estar destinada apenas para fins educacionais (*e.g.*, estudantes, professores, instituições, etc.).
  - **Comercial:** O SDK encontra-se disponível para utilizar mediante um pagamento prévio, geralmente de forma mensal. Em alguns casos, podem existir vários tipos de licenças comerciais, com diferentes funcionalidades e, conseqüentemente, diferente preços.
- **Plugin para desenvolvimento híbrido:** Existência de uma versão do SDK criada (sob a forma de plugin) para permitir a sua utilização no desenvolvimento

<sup>5</sup> <http://socialcompare.com/en/comparison/augmented-reality-sdks>

de aplicações híbridas. Por norma, estes plugins são criados para aplicações desenvolvidas com a framework Apache Cordova.

- **Compatibilidade (Plataforma de Smartphone):** Sistemas operativos de *smartphones* para os quais é destinada a utilização do SDK.
- **Funcionalidades:**
  - **Rastreamento de Imagens 2D:** O SDK permite aplicar a tecnologia de rastreamento de imagens de duas dimensões (*i.e.*, planas).
  - **Rastreamento de Objetos 3D:** O SDK permite aplicar a tecnologia de rastreamento de imagens/objetos de três dimensões.
  - **Reconhecimento c/ marcadores:** O SDK permite aplicar a tecnologia de reconhecimento de *targets* através da utilização de marcadores artificiais.
  - **Reconhecimento s/ marcadores:** O SDK permite aplicar a tecnologia de reconhecimento de *targets* sem necessidade para a utilização de marcadores artificiais (*i.e.*, através de características naturais).
  - **Geolocalização:** O SDK permite aplicar a tecnologia de geolocalização, permitindo identificar locais e *targets* através da sua localização.
  - **Reconhecimento offline:** O SDK não requer a utilização de internet para reconhecer os *targets*, sendo que estes se encontram incorporados na aplicação.
  - **Reconhecimento online (Cloud):** O SDK permite o reconhecimento dos *targets*, com recurso a um portal online (*i.e.*, Cloud).
  - **Multi-Alvos:** O SDK permite o reconhecimento e rastreamento simultâneo de mais do que um *target*.

Os resultados estão representados na Tab. 1, onde se pode observar a listagem dos SDKs de RA juntamente com as suas características inerentes devidamente assinaladas. Importa realçar que, no decorrer desta pesquisa, foram descartados os SDKs que não especificam ou simplesmente não oferecem de versões compatíveis com qualquer plataforma móvel (*e.g.*, BazAR, DART, FLARToolKit, InstantReality, MagicLens, Minerva, MXR Toolkit, NyARToolKit, TinEye, Total Immersion), assim como aqueles que já foram descontinuados ou adquiridos por outras empresas (*e.g.*, ARPA, ARMES, Metaio, PanicAR) e aqueles que apresentam pouca ou nenhuma informação em relação às suas características (*e.g.* Obvious Engine, PRAugmentedReality, String).

Nome	Licença			Plugin para desenvolvimento híbrido	Compatibilidade (Plataforma de Smartphone)	Funcionalidades							
	Open-Source	Gratuita	Comercial			Rastreamento de Imagens 2D	Rastreamento de Objetos 3D	Rastreamento c/ marcadores	Rastreamento s/ marcadores	Geolocalização	Reconhecimento offline	Reconhecimento online (Cloud)	Multi-Alvos
ALVAR	✓	✓	✓	✗	Android, iOS, Windows Phone	✓	✓	✓	✓	n.e.	✓	n.e.	✓
AndAR	✓	✗	✗	✗	Android	✓	✗	✓	n.e.	n.e.	n.e.	n.e.	n.e.
ARcrowd	✓	✗	✗	✗	Android (HTML5 – browser), iOS (HTML5 browser)	✓	n.e.	✓	✓	n.e.	✗	✓	n.e.
ARLab	✓ (não confirmado – erro do servidor)	✗	✓	✗	Android, iOS	✓	✗	✓	n.e.	✓	✓	n.e.	✓
AR-Media	✓	✓	✓	✗	Android, iOS	✓	✓	✓	✓	✓	n.e.	n.e.	n.e.
ARToolKit	✓	✗	✓	✗	Android, iOS, Windows Phone	✓	✗	✓	✓	✓	✓	n.e.	✓
ArUco	✓	✗	✗	✗	Android, iOS	✓	✗	✓	n.e.	n.e.	n.e.	n.e.	n.e.
Augment	✗	✓	✓	✓	Android, iOS	✓	✗	✓	✓	✗	✓	✓	n.e.
Augmented Pixels	n.e.	n.e.	n.e.	✗	Android, iOS	✗	✓	✓	n.e.	n.e.	n.e.	n.e.	n.e.
Augmented Pro	✗	✓	✓	✗	Android, iOS	✓	✓	n.e.	✓	n.e.	n.e.	n.e.	n.e.
Aurasma	✗	✓	✓	✗	Android, iOS	✓	✗	✓	✗	✓	✗	✓	✗
BeyondAR	✓	✓	✓	✗	Android	n.e.	n.e.	n.e.	✓	✓	n.e.	n.e.	n.e.
BlippAR	✗	✓	✓	✗	Android, iOS	✓	✗	✓	✗	✗	✗	✓	✗

CraftAR	X	✓	✓	✓	Android, iOS	✓	X	X	✓	X	✓	✓	✓
DroidAR	✓	X	✓	X	Android	✓	✓	✓	✓	✓	✓	n.e.	n.e.
EasyAR	X	✓	✓	X	Android, iOS, Windows Phone	✓	✓	✓	✓	X	✓	✓	✓
EzAR	X	✓	✓	✓	Exclusivo para Cordova	n.e.	✓ (Deteção de faces)	X	✓	✓	n.e.	n.e.	n.e.
Goblin XNA	✓	X	X	X	Windows Phone	✓	n.e.	✓	n.e.	n.e.	n.e.	n.e.	n.e.
Here Mobile	X	✓	✓	X	Android, iOS	✓	✓	n.e.	✓	n.e.	✓	✓	n.e.
Holobuilder	X	✓	✓	X	Android, iOS	✓	n.e.	✓	✓	n.e.	n.e.	✓	✓
IN2AR	X	✓	✓	X	Android, iOS	✓	n.e.	n.e.	✓	n.e.	n.e.	n.e.	✓
KudanAR	X	✓	✓	X	Android, iOS	✓	✓	✓	✓	X	✓	X	✓
LayAR	X	✓	✓	✓	Android, iOS, BlackBerry	✓	X	✓	n.e.	✓	X	✓	n.e.
Luxand	X	✓	✓	X	Android, iOS	✓	(Deteção de faces)	n.e.	n.e.	n.e.	✓	X	n.e.
Maxst	X	✓	✓	X	Android, iOS	✓	✓	✓	✓	n.e.	n.e.	n.e.	✓
Mixare	✓	X	X	X	Android, iOS	n.e.	n.e.	n.e.	n.e.	✓	n.e.	n.e.	n.e.
OsgART	✓	X	✓	X	Android, iOS	n.e.	n.e.	✓	✓	n.e.	n.e.	n.e.	n.e.
PixLive	X	✓	✓	✓	Android, iOS	✓	X	✓	n.e.	✓	✓	✓	n.e.
PTAM	✓	✓	✓	X	Android, iOS	n.e.	n.e.	X	✓	n.e.	n.e.	n.e.	n.e.
SLARToolkit	✓	X	X	X	Windows Phone	✓	n.e.	✓	n.e.	n.e.	n.e.	n.e.	✓
SSTT	X	n.e.	n.e.	X	Android, iOS, Windows Phone	✓	n.e.	✓	✓	n.e.	n.e.	n.e.	n.e.

UART	✓	✗	✗	✗	iOS	✓	n.e.	✓	n.e.	n.e.	n.e.	n.e.	✓
Vuforia	✗	✓	✓	✓	Android, iOS, Windows Phone	✓	✓	✓	✓	✓	✓	✓	✓
Wikitude	✗	✓	✓	✓	Android, iOS, Windows Phone, BlackBerry	✓	✓	✓	✓	✓	✓	✓	✓
Xludia	✗	✓	✓	✗	Android, iOS, Windows Phone	✓	✓	n.e.	✓	n.e.	n.e.	✓	n.e.
XZIMG	✗	✓	✓	✗	Android, iOS	✓	✓ (Detecção de faces)	✓	✓	n.e.	n.e.	n.e.	n.e.

**Tabela 1:** Características dos diferentes SDKs de RA existentes atualmente.

**Legenda:**

- ✓ - O SDK contém a respetiva característica
- ✗ - O SDK não contém a respetiva característica
- n.e. – Não Especificado

Observando os resultados da pesquisa, é possível notar a presença acentuada de características que não se sabe ao certo se estão implementadas ou não, devido à pouca documentação existente acerca dos respetivos SDKs. Tendo isso em conta, só será possível de obter estas confirmações e outras informações através da utilização destes SDKs. No entanto, para o projeto em mão, nem todos os SDKs da Tabela 1 foram testados, uma vez que apenas são requeridas as seguintes características:

- Plugin para desenvolvimento híbrido
- Rastreamento s/ marcadores
- Rastreamento de imagens 2D
- Reconhecimento offline
- Multi-Alvos
- Licença Open-Source/Gratuita
- Sobreposição de *Overlays*\*

\* É utilizado o termo *overlay* para descrever a informação virtual representada por imagens 2D ou 3D.

Uma vez que é pretendido o desenvolvimento de uma aplicação móvel de natureza híbrida, é essencial que os SDKs recolhidos contenham uma versão que suporte esse desenvolvimento. Este suporte é refletido através de um plugin criado para esse efeito. Tendo em conta apenas esta característica, é possível seleccionar desde já um conjunto de SDKs para o projeto: *Augment*, *CraftAR*, *EzAR*, *LayAR*, *PixLive*, *Vuforia* e *Wikitude*. Quanto às restantes características, estas serão abordadas e justificadas mais à frente no documento, na seção das metodologias (Capítulo 3). É de notar ainda que o último critério (*i.e.*, Sobreposição de *Overlays*), não foi considerado para a análise dos SDKs, uma vez que a grande maioria não menciona esta funcionalidade nas suas respetivas documentações. Tendo isso em conta, caso um SDK contenha as restantes características, será testado no sentido de averiguar se contém ou não o último critério.

### 2.3. Aplicações para Património Cultural

A RA inicialmente era fundamentalmente utilizada em circunstâncias de trabalho, como auxílio numa operação cirúrgica ou na construção e reparação de máquinas. Devido à serventia prestada pela RA neste tipo de situações, o mundo do trabalho (sobretudo tecnológico) apresentava um grande interesse em aplicar a RA às respetivas profissões, pelo que apenas as pessoas que as desempenhavam seriam as únicas a usufruir desta tecnologia. Entretanto, outra área de interesse em sistemas de RA surgiu e permitiu, não só aos desenvolvedores, mas também ao público, a usufruição deste tipo de experiências. Essa área denomina-se por património (ou herança) cultural (Brogni et al., 1999). A UNESCO definiu, em 1992, este conceito de Património Cultural como sendo todo o conjunto de monumentos, estruturas arquiteturais e locais que apresentam um valor universal excepcional do ponto de vista histórico, estético, etnológico e antropológico (Ibrahim, Mohamad Ali, & Mohd Yatim, 2011).

Existem dois tipos de vertentes, focadas no estudo da atividade humana, que contribuem para o Património Cultural e que importa serem distinguidas. A primeira, relacionada com o ramo da arqueologia, destina-se apenas à recuperação de vestígios e artefactos perdidos. Nesta vertente, tanto os arqueólogos como os historiadores e responsáveis pelos locais de Património Cultural dão ênfase à autenticidade e precisão em relação à visualização e representação dos materiais. Já a outra vertente, representada por tecnólogos, pretende apoiar-se mais nas tradições, narrativas e evidências de arte e cultura das populações. Ao contrário da vertente anterior, esta demonstra menos interesse na autenticidade e precisão do conteúdo histórico, focando-se mais nos aspetos técnicos relacionados com a apresentação da visualização e a forma como esta foi implementada (Roussou, 2000).

Hoje em dia, o Património Cultural e tudo o que o constitui corre perigo de destruição e de decadência, devido sobretudo a poluição, atos de vandalismo, negligência e extravio, e turismo descomedido (Addison, 2000). Para um visitante de um espaço cultural que desconheça *a priori* os artefactos que vai observar, se estes se encontrarem danificados será uma tarefa árdua identificá-los ou imaginar a forma e os detalhes da sua composição original (Gîrbacia, Butnariu, Orman, & Postelnicu, 2013). Em suma, prejudicar o Património Cultural significa prejudicar o nosso passado, algo impossível de recuperar e tendo a recriação como única alternativa. É aqui que a RA apresenta um papel primordial e a sua relação com o Património Cultural é transformada num outro conceito denominado por Património Virtual (PV).

O PV refere-se ao uso de tecnologias interativas e baseadas em computador, com o propósito de registar, preservar ou recriar artefactos, locais e atores de relevância histórica, artística, religiosa ou cultural, e apresentar os resultados abertamente ao público no geral, de maneira a fornecer experiências educacionais formativas através da manipulação eletrónica do tempo e do espaço (Stone & Ojika, 2000). Embora a preservação do passado seja por si só uma razão suficiente para justificar a existência deste ramo, é possível enumerar uma lista plausível de motivações para a prática do mesmo (Sanders, 2007):

- Não há melhor maneira de testar se de facto informações arqueológicas recolhidas resultam na realidade
- Não há melhor maneira de testar teorias complexas espaciais, comportamentais e temporais
- Não há melhor maneira de testar a precisão de interpretações e provas do passado
- Não há melhor maneira de reunir artefactos dispersos a nível global e introduzi-los numa simulação envolvendo os seus contextos originais
- Não há melhor maneira de visualizar mudanças e desenvolvimento sem perturbações

- Não há melhor maneira de absorver conjuntos de dados complexos sobre o passado do que visualmente, interactivamente e em 3D, tal como fazemos na vida real

De modo a complementar as motivações para a prática de PV, é necessário também perceber quais são, na prática, as capacidades e funcionalidades que esta área proporciona (El-Hakim, Beraldin, Picard, & Godin, 2004):

- Documentação de edifícios e objetos históricos para efeitos de reconstrução ou restauro no caso da ocorrência de fogos, terremotos, inundações, guerras, erosões, etc.
- Criação de recursos educacionais para investigadores e estudantes em áreas de história e cultura
- Reconstrução de monumentos históricos que existem parcialmente ou já não existem de todo
- Visualização de cenários de pontos de vista impossíveis de serem visualizados no mundo real, devido a problemas de altura ou acessibilidade
- Interação com objetos sem correr o risco de os danificar
- Fornecimento de turismo virtual e de visitas virtuais a museus

Torna-se bastante evidente o potencial apresentado pelo PV para melhorar vários setores da vida, como a educação e a cultura, e a RA começa a tirar proveito desse potencial e a ganhar bastante importância na reconstrução virtual de monumentos históricos e auxílio a tutores de museus, historiadores e arqueólogos na reprodução de locais históricos, no local original, de como estes eram no seu auge (Fritz, Susperregui, & Linaza, 2005). De modo a avaliar se as reconstruções são fiéis aos elementos originais e distinguir as partes reais das partes virtuais, os sistemas de RA permitem a gestão de texto, imagens e modelos 3D simplificados em tempo real, de modo a complementar a visão real do utilizador (Brogni et al., 1999). A visualização simultânea dos artefactos reais com as imagens virtuais tem um grande impacto no conhecimento sobre a história, sobre os efeitos do tempo sobre os materiais e sobre a preservação dos recursos que tenham eventualmente de ser reconstruídos fisicamente (Fritz et al., 2005).

Após esta exposição do conceito de Património Cultural e da sua variante, o PV, surge agora a necessidade de demonstrar como esta última funciona, na prática, expondo algumas das suas metodologias. Em primeiro lugar, existem três grandes domínios do PV, que podem ser considerados como passos a seguir no desenvolvimento de sistemas neste âmbito (Addison, 2000). Estes são:

- **Documentação:** Neste primeiro domínio, é recolhida toda a informação necessária sobre os objetos de Património Cultural escolhidos, da forma mais precisa possível. Esta recolha inclui também atividades como a análise do local onde se inserem esses objetos e a realização de esboços e medições. Para simplificar este processo, existem diversas técnicas de digitalização 3D, podendo ser divididas consoante o tipo de artefacto a ser digitalizado (Pavlidis, Koutsoudis, Arnaoutoglou, Tsioukas, & Chamzas, 2007):
  - **Digitalização de Objetos:**
    - Digitalização a laser
    - Digitalização baseada em luz estruturada
    - Digitalização baseada em silhuetas
    - Digitalização baseada em fotografias estereoscópicas
    - Digitalização baseada em vídeo
    - Digitalização baseada em sombreamento

- Digitalização baseada em texturas
- Digitalização baseada em fotometria
- Digitalização baseada em focagem
- Digitalização baseada em sistemas de contacto
- **Digitalização de Monumentos:**
  - Técnica empíricas
  - Técnicas topográficas
  - Técnicas de digitalização a laser
  - Fotogrametria

A escolha da(s) técnica(s) de digitalização dependerá das características do artefacto em questão, do orçamento estabelecido e das condições e restrições impostas no local onde se encontra (Ibrahim et al., 2011).

- **Representação:** Após realizada a recolha das informações referentes aos artefactos, o próximo procedimento será replicá-lo tridimensionalmente, caso este se encontre na íntegra ou, caso contrário, criar uma visualização do seu estado atual (ou uma aproximação, caso já não exista de todo). Aqui o foco estará em aspetos técnicos como a iluminação computacional, a eficácia das texturas e da técnica de modelação 3D escolhida, e a rapidez da representação e processamento de informação não tratada proveniente da digitalização 3D (Ibrahim et al., 2011).
- **Disseminação:** O último domínio do PV refere-se à forma como os modelos 3D serão apresentados aos utilizadores. As formas mais populares de visualização destes modelos são através da RV e, claro, da RA. A RA torna-se num meio de apresentação mais interessante na medida em que torna possível a observação do espaço cultural simultaneamente com os modelos 3D incorporados no mesmo. Neste domínio, existe ainda o cuidado de preservar a validade e autenticidade do conteúdo que é apresentado (Ibrahim et al., 2011).

Tendo em conta este último domínio do PV, como já foi mencionado anteriormente, a RA é utilizada como um meio de apresentação do conteúdo virtual no espaço cultural envolvente. Este conteúdo pode tratar-se de uma representação integral do artefacto (caso este já não exista ou como forma de preservação) ou de uma reconstrução do mesmo (caso este se encontre danificado ou incompleto). Para tal, de modo a criar uma experiência singular de RA no contexto de Património Cultural, foram definidos os seguintes requisitos (El-Hakim et al., 2004):

- Alta precisão geométrica
- Captura de todos os detalhes
- Fotorrealismo
- Nível alto de automação
- Baixo custo
- Portabilidade
- Flexibilidade da aplicação
- Eficiência do tamanho do modelo

Além destes oito requisitos, é também importante que a aplicação forneça algum valor cultural e que, do ponto de vista do utilizador, seja atrativa e fácil de usar. Por fim, a reutilização dos dados e disponibilidade a nível global são outros requisitos a ter em conta no desenvolvimento deste tipo de experiências (Fritz et al., 2005). Nas duas próximas seções, serão demonstrados alguns exemplos significativos de aplicações criadas no âmbito do Património Cultural. De modo a existir um termo de comparação, serão abordadas aplicações feitas sem RA e outras com RA, sendo estas últimas de maior relevância para o projeto atual.

### 2.3.1. Aplicações com Realidade Virtual

#### Anfiteatro Romano de Byblos

Hoje em dia, a apreciação plena de um espaço de Património Cultural é comprometida pelo estado degradado apresentado pela maioria, sobretudo os mais antigos. Para os visitantes, torna-se bastante difícil conseguir interpretar um espaço que apresente falhas na sua composição e consequentemente perceber o contexto ou as funcionalidades de certos elementos desse espaço, assim como a sua significância do ponto de vista arquitetural e histórico. Este é o caso do anfiteatro romano situado nas ruínas de Byblos, uma das cidades mais antigas e continuamente inabitadas do mundo, sendo inclusivé considerada como património cultural da UNESCO<sup>6</sup>. Infelizmente resta muito pouco deste anfiteatro histórico e, o que resta, apresenta-se num estado lastimável e pouco apelativo para a sua visita por parte de turistas (Fig. 6). De modo a preservar o valor histórico deste património e cativar o interesse dos visitantes, foi criado um sistema de RV que fornece uma experiência imersiva na qual é possível observar, navegar e interagir com uma reconstrução virtual do anfiteatro, elaborada de forma a se aproximar o mais possível do original no que toca à sua aparência e às suas características únicas (Younes et al., 2017).

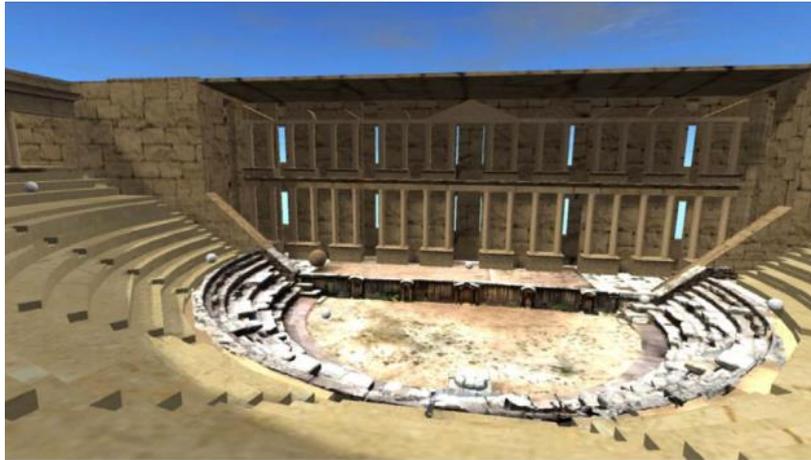


**Figura 6:** Estado do anfiteatro romano de Byblos, nos dias que correm. (Monico, 2016)

A aplicação de RV desenvolvida permite aos utilizadores visualizar e interagir com o anfiteatro de Byblos tridimensional (Fig. 7), através de uma visita guiada. O mundo virtual desta aplicação foi criado através do mecanismo Unity<sup>7</sup>, que é responsável pelo desenvolvimento de jogos para diversas plataformas como o PC, consolas, dispositivos móveis e websites. De modo a ampliar o nível de interação da aplicação e torná-la mais apelativa aos turistas, foi implementado um guia virtual que surge em diversos pontos de interesse do anfiteatro. Este guia virtual estará à espera que o utilizador se aproxime de forma a fornecer informações (sob a forma de áudio) sobre o respetivo ponto de interesse. Concluída uma visita, o guia irá mover-se para outro ponto de interesse, sendo que o utilizador pode optar por não se deslocar de imediato a esse ponto e navegar livremente pelo anfiteatro.

<sup>6</sup> <http://whc.unesco.org/en/list/295>

<sup>7</sup> <https://unity3d.com/>.



**Figura 7:** Exemplo ilustrativo do anfiteatro de Byblos 3D, durante a experiência de RV. (Younes et al., 2017)

Para vivenciar esta experiência, o utilizador terá então que utilizar o sistema visual Oculus Rift, complementado com um comando (*e.g., joystick*) para controlar a sua posição e orientação no anfiteatro virtual. Ainda relativamente ao aspeto da navegação, existe a ilusão de gravidade neste tipo de experiências de RV, *i.e.*, se um utilizador decidir descer de uma superfície alta de forma brusca, este terá a perceção de que realmente está a cair, na vida real. Sendo que esta sensação pode causar algum desconforto, foi criado um sistema de evasão de obstáculos, de forma a impedir que o utilizador atravessasse superfícies com uma diferença de alturas considerável.

Embora seja impossível recuperar fisicamente o anfiteatro romano de Byblos, esta iniciativa permite o seu “renascimento” e, conseqüentemente, a preservação de um património de extrema importância, devido ao seu valor histórico e cultural. Aplicando este tipo de aplicações de RV a outros espaços históricos degradados, será dinamizada a exploração aos mesmos, devido à componente imersiva, e enriquecida a cultura dos seus visitantes, ao incluir informações e detalhes que existiam originalmente nesses espaços e que são atualmente impossíveis de observar fisicamente no local.

## RecoVR Mosul

Em 2014, o grupo militante conhecido como o Estado Islâmico procedeu à destruição de antiguidades pela cidade de Mosul, no Iraque, tendo também danificado estátuas e outros artefatos presentes no Museu de Mosul. Tendo em conta este infortúnio, surgiu uma colaboração entre o jornal *The Economist*<sup>8</sup> e o grupo *Rekrei*<sup>9</sup> (anteriormente reconhecido por Projeto Mosul ou Visualise) de forma a criarem uma experiência de VR que recreasse o museu e os seus artefatos destruídos (Drum, 2016). A essência do projeto *Rekrei* consiste na recolha de imagens e informações acerca de artefatos destruídos, fornecidos pelo público, que são posteriormente combinadas de forma a serem criados modelos 3D e proceder-se então à reconstrução virtual desses artefatos, recorrendo a fotogrametria. Posto isto, este grupo, juntamente com o *The Economist Media Lab*, procederam à criação de uma experiência imersiva de RV, traduzida numa visita guiada pelo Museu de Mosul rejuvenescido (Fig. 8), apoiada pela narração de Tom Standage, um dos editores do *The Economist*, sobre o significado dos objetos e os métodos de reconstrução.

<sup>8</sup> <https://www.economist.com/>

<sup>9</sup> <https://rekrei.org/>



**Figura 8:** Exemplo ilustrativo da experiência de RV para o Museu de Mosul. (Drum, 2016)

Durante o desenvolvimento desta aplicação, foram tidos em contas os bons princípios dos sistemas de RV, como o conforto, a presença e o envolvimento, e, ao mesmo tempo, respeitando as ideias iniciais do projeto e da sua contextualização. Para tal, o ambiente e o espaço do Museu foram completamente remodelados, ao contrário dos modelos 3D originais do artefatos, que não sofreram alterações. No fim, embora o nível de interação não seja o desejado numa visita guiada virtual, esta iniciativa resultou numa experiência de RV agradável para os utilizadores e culturalmente relevante, uma vez que resguarda virtualmente o património histórico que poderá eventualmente, como neste caso, ser fisicamente comprometido.

## Dreams of Dalí

A experiência de RV denominada por Dreams of Dalí<sup>10</sup> foi criada com o intuito de testar uma nova forma de olhar para a arte e de a explorar. A essência deste projeto encontra-se na imersão dos seus utilizadores num novo mundo, constituído pelos detalhes e pela história de uma pintura (Goodby Silverstein & Partners, 2016). Para tal, foi escolhida uma das primeiras pinturas de Salvador Dalí, apelidada por *Archeological Reminiscence of Millet's "Angelus"* (1935), que surgiu a partir do fascínio de Dalí pela pintura *The Angelus*, de Jean-François Millet (Fig. 9). Objetivamente, esta pintura representa um casal de camponeses que interrompera as suas tarefas num campo de batatas para proceder a uma oração (com o mesmo nome da pintura) mas, para Dalí, tratava-se do trabalho mais inquietante e perturbador que ele alguma vez conheceu (Dalí, 1981).

---

<sup>10</sup> <http://www.dreamsofdali.net/>



**Figura 9:** Pintura “The Angelus” (1857-1859), por Jean-François Millet.<sup>11</sup>

Esta experiência foi disponibilizada ao público no Museu Dalí, em São Petersburgo, através do sistema visual Oculus Rift, juntamente com uns auriculares como complemento auditivo da experiência. Apesar de não terem controlo total neste mundo virtual, os utilizadores podem olhar em qualquer direção, deslocando-se desta forma sobre o mesmo (Fig. 10). Além das extensas paisagens criadas para o efeito de imersão completa num mundo surreal, será possível explorar as gigantescas torres e figuras criadas por Dalí, como o elefante de pernas compridas e a rapariga que saltava à corda, assim como outros detalhes interessantes que este terá imaginado aquando a criação da sua pintura. Ao lado do utilizador, encontra-se ainda um monitor que mostra o que este está a visualizar no momento, de forma a partilhar a sua experiência com os restantes visitantes que o rodeiam (Fig. 11).



**Figura 10:** Imagem retirada do trailer da experiência Dreams of Dalí, que mostra o novo mundo pelos olhos do utilizador. (Silverstein, 2016)

<sup>11</sup> <http://www.musee-orsay.fr/en/home.html>



**Figura 11:** Imagem retirada do trailer da experiência Dreams of Dalí, que mostra o utilizador durante a experiência e com esta exibida num monitor ao seu lado. (Silverstein, 2016)

Uma grande vantagem em utilizar RV para este tipo de experiências reside na superação do problema da limitação humana no que toca à sua imaginação fugaz. O aspeto mais importante na contemplação de uma pintura reflete-se precisamente na imaginação que é criada a partir da mesma e, através da RV, esta pode ser enriquecida e vivenciada pelo observador. Tendo isso em conta, além de preservar os atributos da pintura em questão, os sistemas de RV isolam o utilizador dos restantes, de forma a que este se encontre “imerso na pintura” e consiga meditar sobre a experiência sem interferência exterior (Ziamou, 2016).

### 2.3.2. Aplicações com Realidade Aumentada

#### ARCHEOGUIDE

O ARCHEOGUIDE (*Augmented Reality-based Cultural Heritage On-site GUIDE*) (Vlahakis et al., 2001) trata-se de um projeto destinado a fornecer uma visita guiada electrónica aos visitantes de um espaço cultural que, neste caso, refere-se à região arqueológica de Olympia, na Grécia. Através deste sistema, os utilizadores dispõem de ajuda durante a visita, no local e em tempo real, onde a grande peculiaridade reside na capacidade de observação da reconstrução das ruínas antigas, através da tecnologia de RA (Fig. 4). A aplicação deste sistema não impõe qualquer risco de danificação ao espaço cultural nem causa perturbação para com os restantes visitantes e as suas respetivas experiências culturais no local.

Antes de partirem para a experiência, os visitantes devem preencher um questionário relativamente aos seus interesses, dados pessoais (*e.g.*, idade, sexo, idioma) e uma breve história dos seus passados, de maneira a que o sistema (*i.e.*, o roteiro) seja adaptado e configurado com o idioma escolhido, as regiões mais apropriadas, o tempo disponível e o nível de detalhe da informação apresentada (com base na idade e na história dos visitantes). Personalizada a viagem, o visitante terá então que transportar e manusear um pequeno dispositivo móvel (Fig. 12), de modo a interagir com a interface do sistema e controlar a sua viagem, e utilizar óculos de RA desenvolvidos para esta experiência, através dos quais é capaz de observar a reconstrução dos monumentos em 3D.



**Figura 12:** Exemplo ilustrativo da interface do ARCHEOGUIDE, com uma imagem in situ e uma imagem da perspetiva do utilizador. (Archeomaticamagazine, 2013)

A visualização desta informação virtual é baseada na posição e orientação do utilizador, sendo composta e representada em tempo real. Juntamente com o sistema, está incorporada uma base de dados contendo todo o material de teor cultural utilizado nas viagens e que serve também de apoio a investigações científicas e promoção da educação cultural.

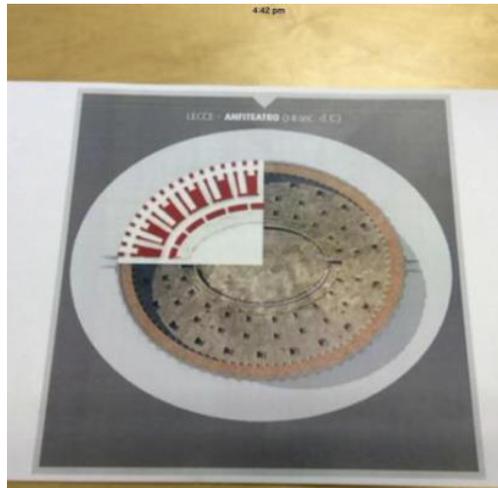


**Figura 13:** Reconstrução de um monumento (imagem esquerda) através de RA (imagem direita). (Vlahakis et al., 2001)

Este sistema teve uma *feedback* positivo por parte dos utilizadores, particularmente pela faixa etária mais nova, sendo considerado como uma mais valia na visita guiada e favorecendo a aprendizagem cultural. No entanto, surgiram eventuais problemas como a limitação na escolha dos idiomas, o contraste fraco do ecrã do dispositivo (dificultando o seu visionamento com a presença de luz solar forte), ao conteúdo virtual limitado (não cobria todos os pontos de interesse do local), ao desconforto causado pelo tamanho e peso dos óculos de RA e o custo considerável de utilização e seguro de todo o equipamento. Estes problemas foram tidos em conta de forma positiva, uma vez que seriam suplantados pelas críticas positivas em relação à experiência no seu todo e pelo interesse demonstrado em vivenciar este tipo de experiências com RA noutros espaços culturais.

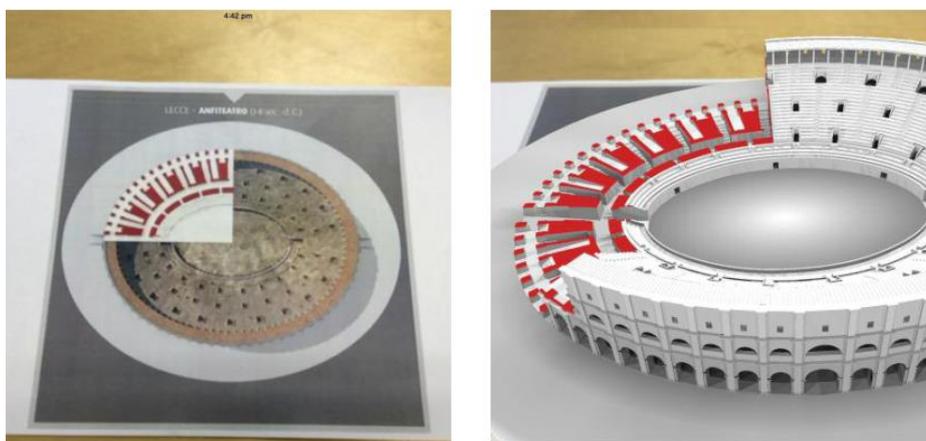
## LecceAR

O LecceAR<sup>12</sup> (Scopigno et al., 2015) é uma aplicação móvel (*e.g.*, *smartphone* ou *tablet*) desenvolvida exclusivamente para iOS e com o intuito de reconstruir tridimensionalmente o anfiteatro romano de Lecce, que se encontra consideravelmente deteriorado e apenas parcialmente visível. Esta reconstrução virtual do anfiteatro é feita através da tecnologia de RA, utilizando técnicas de rastreamento que não implicam o uso de marcadores. Apesar de ser uma ideia mais apelativa, esta reconstrução não será feita *in situ* (*i.e.*, sobre o anfiteatro original), mas sim através de uma imagem-alvo (Fig. 14), onde irá surgir uma maquete 3D do anfiteatro, produzida pela tecnologia de RA.



**Figura 14:** Imagem-alvo usada pela aplicação. (News, 2015)

O processo de funcionamento desta aplicação reflete-se em dois módulos, sendo o primeiro módulo responsável pela deteção e rastreamento de uma imagem-alvo, e o segundo módulo responsável pela composição e representação 3D do objeto virtual. No primeiro módulo, a posição e orientação do dispositivo móvel (*i.e.*, a câmara) são estimadas, em relação à imagem-alvo. O utilizador terá então que apontar a câmara para a imagem-alvo apropriada, de modo a despoletar a criação do objeto virtual, juntamente com o processo de rastreamento, de forma a alinhar o anfiteatro 3D com a imagem-alvo e observá-la de vários pontos de vista em tempo real e de forma estável. Realizado este processo, o segundo módulo será então responsável por compor e representar tridimensionalmente o anfiteatro 3D, sobre a imagem-alvo (Fig. 15).



**Figura 15:** Imagens ilustrativas do funcionamento da aplicação: Imagem-alvo detetada (imagem esquerda); Processo de reconstrução 3D completo (imagem direita). (Scopigno et al., 2015)

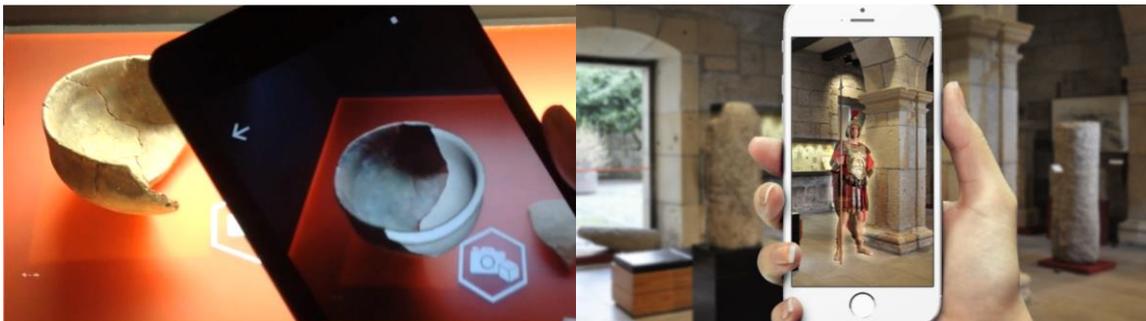
<sup>12</sup> <http://vcg.isti.cnr.it/LecceAR/>

Esta aplicação foi apresentada, em 2015, no museu MUST em Lecce, Itália, no âmbito da exposição Living Lab<sup>13</sup>, que pretende fornecer experiências enriquecedoras culturalmente, aos seus visitantes, através de tecnologias distintas como a RA. Através da reconstrução do anfiteatro romano de Lecce, o visitante terá uma experiência única na medida em que será possível observá-lo na íntegra, próximo daquilo que seria no pico da sua existência e em tempo real, assim como pormenores difíceis de detetar no anfiteatro original ou que já não existem de todo.

## VisitAR

O VisitAR<sup>14</sup> é uma aplicação desenvolvida pela empresa portuguesa IT People Innovation e é um exemplo notável de uma aplicação de RA no âmbito de Património Cultural. Vencedora do concurso de inovação Santa Casa Challenge, na área de Cultura e Património (Rodrigues, 2016), esta aplicação destina-se a engrandecer a ligação entre os visitantes e o património, utilizando a tecnologia de RA para reconstruir virtualmente artefatos destruídos ou incompletos e recriar ambientes semelhantes aos do passado, em tempo real. Como primeiros casos de estudo, a aplicação foi desenvolvida para o Museu da Guarda e o Museu Francisco Tavares Proença Júnior.

Em primeiro lugar, no interior do museu, numa experiência não só visual, mas também audível, os utilizadores podem observar o restauro de elementos históricos como vasos, espadas e estátuas (Fig. 16), descobrir como eram utilizados os utensílios na altura, examinar o interior dos monumentos em exposição (simulando visão Raio X) e, através da projeção virtual de um guia (Fig. 16), análogo a um holograma, são acompanhados ao longo do percurso e presenciam a sua interação com as peças históricas do espaço cultural. Junto aos diversos pontos de interesse do espaço cultural, encontrar-se-ão marcadores que foram colocados estrategicamente (*i.e.*, sem comprometerem a integridade dos artefatos e do espaço envolvente), de forma a acionarem as respetivas experiências de RA (Martins, 2016).



**Figura 16:** Reconstrução de um vaso (imagem esquerda) e projeção de um guia virtual (imagem direita), através de RA. (Martins, 2016)

Um grande benefício prestado por esta aplicação prende-se pelo facto de a experiência fornecida no interior dos espaços culturais ser possível de continuar fora de portas, *i.e.*, na respetiva cidade. Utilizando tecnologias de rastreamento sensorial, o utilizador pode explorar a cidade através de rotas fornecidas pela aplicações (Fig. 17) que, por sua vez, contêm diversos pontos de interesse, representativos de marcos históricos da cidade, com experiências de RA. Desta forma, numa espécie de “caça ao tesouro”, o utilizador passará a conhecer a história da cidade e constatar como esta seria antigamente.

<sup>13</sup> <http://livinglabs.regione.puglia.it/en/home>

<sup>14</sup> <http://visitar.nextreality.com/>

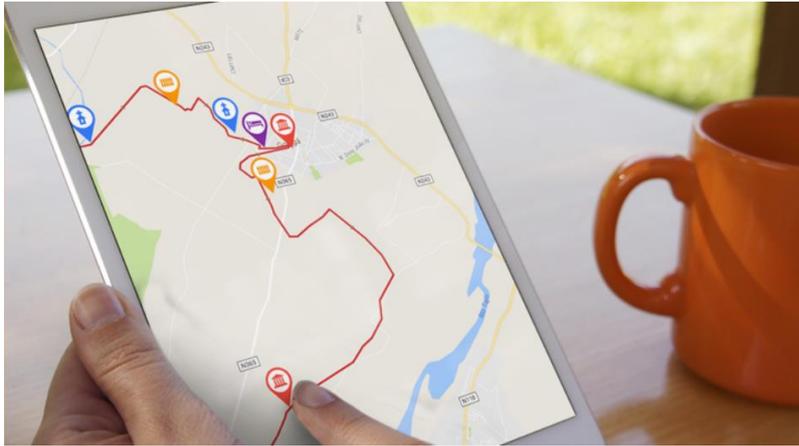


Figura 17: Exemplo ilustrativo de uma rota, no modo exterior da aplicação. (Martins, 2016)

Em suma, através de um conjunto de experiências de RA criadas *a priori*, o VisitAR permite a dinamização de espaços culturais e, conseqüentemente, o enriquecimento das visitas aos mesmos. Apesar de utilizar uma abordagem de rastreamento através de marcadores, que pode não ser aceite ou apropriada em certos espaços ou monumentos, torna-se uma solução mais ágil no desenvolvimento das experiências de RA em espaços interiores, como é o caso dos museus.

## 2.4. Desenvolvimento de Aplicações Multiplataforma

Nos dias que correm, existe um interesse acérrimo em desenvolver aplicações móveis destinadas a mais do que um sistema operativo (*e.g.*, Android, iOS, etc.), uma vez que existe uma quantidade considerável de utilizadores em cada uma destas plataformas. Tendo isso em conta, e dependendo das características do projeto em questão, surge o dilema da escolha do tipo de aplicação a ser desenvolvida: nativa ou multiplataforma. Estas são as duas abordagens fundamentais no desenvolvimento móvel e existem claras diferenças entre ambas, sendo que não existe uma que seja, *a priori*, a escolha mais acertada. A escolha da natureza da aplicação móvel prende-se por vários fatores, que serão observados e confrontados na próxima seção. Por sua vez, nesta seção, será apresentada a abordagem multiplataforma, incluindo as suas características e algumas das vantagens e desvantagens mais significativas no seu desenvolvimento.

As aplicações multiplataforma, como o próprio nome indica, são destinadas ao funcionamento em mais do que uma plataforma móvel. A diferença principal deste tipo de aplicações móveis para as nativas prende-se, precisamente, na forma como esse funcionamento é capacitado. As aplicações nativas são construídas utilizando um SDKS específico para cada sistema operativo móvel, que, por sua vez, adota linguagens de programação utilizadas no desenvolvimento para a respetiva plataforma (*e.g.*, Java para Android, Swift e Objective-C para iOS, etc.). Para tal, será necessário produzir o mesmo código, em linguagens diferentes, para cada plataforma. Esta inconveniência é contornada pelas aplicações multiplataforma, uma vez que é utilizado o mesmo código-base para diversas plataformas móveis. No entanto, as linguagens de programação utilizadas para o desenvolvimento de aplicações multiplataforma podem diferir, consoante as características e requisitos do projeto em questão. Visto isto, as aplicações multiplataforma podem ser ainda divididas em aplicações multiplataforma nativas e aplicações híbridas (Patil, 2016).

As aplicações multiplataforma nativas servem-se de APIs (*Application Programming Interfaces*) fornecidos pelos SDKS de cada sistema operativo móvel. Através destas APIs, são utilizadas linguagens de programação diferentes daquelas usadas particularmente em cada plataforma (*i.e.*, nativas). Apesar disto, continua a ser possível aceder e utilizar as funcionalidades (*e.g.*, câmara, acelerómetro, contactos, etc.) e componentes (*i.e.*, botões, menus, etc.) nativas do sistema

operativo em questão. Tendo isso em conta, o desempenho da UI (*User Interface*) em aplicações multiplataforma nativas aproxima-se bastante daquele apresentado pelas aplicações nativas (Rami A., 2017). Para o desenvolvimento deste tipo de aplicações, existem frameworks como o React Native, NativeScript, Xamarin e Appcelerator Titanium (Saccomani, 2017).

Embora o conceito seja semelhante, as aplicações híbridas diferem significativamente das aplicações multiplataforma nativas no que toca à sua arquitetura e metodologia. O termo “híbridas” refere-se ao facto de este tipo de aplicações ser desenvolvido utilizando tecnologias web (HTML5, CSS e Javascript) e, ao mesmo tempo, poderem utilizar recursos nativos provenientes do dispositivo móvel. (Kanzig, 2015). Este acesso às funcionalidades nativas do sistema operativo móvel é conseguido através da utilização de frameworks criadas esse efeito (*e.g.*, Apache Cordova, Intel XDK, Ionic, etc.) que serão vistas mais à frente neste documento. Estas frameworks servem, de certa forma, como ponte entre as funcionalidades nativas do sistema operativo e a componente onde a aplicação é corrida. Esta componente trata-se de um browser minimalista, denominado por WebView, e é responsável por exibir todo o conteúdo UI e controlar a componente lógica da aplicação (*i.e.*, Javascript), pelo que o desempenho da aplicação híbrida dependerá substancialmente da WebView. Por fim, existem algumas diferenças entre os vários tipos de Webviews disponíveis, dependendo da framework escolhida (Looper, 2015; Saccomani, 2017; Walsh, 2015).

Existem boas razões para o desenvolvimento de aplicações multiplataforma, que as tornam como a opção predileta. Contudo, este tipo de aplicações também apresenta algumas desvantagens notáveis, o que poderá dificultar a escolha do tipo de aplicação a ser desenvolvida. Esta escolha deve apoiar-se numa reflexão prudente, tendo em conta diversos fatores como os objetivos e público-alvo, as plataformas-alvo, as capacidades e conhecimentos técnicos dos programadores e a quantidade e tipo de funcionalidades nativas necessárias ao desenvolvimento da aplicação. Na próxima seção, são confrontados os diferentes tipos de aplicações móveis, expondo os seus pontos fortes e fracos, por forma a auxiliar o processo de decisão e demonstrar de que forma a abordagem multiplataforma se pode apresentar como a opção viável.

#### **2.4.1. Estudo Comparativo dos Tipos de Aplicação Móvel**

Após a conceção de uma ideia para uma aplicação móvel, segue-se o processo crucial e por vezes complexo de decisão da natureza do seu desenvolvimento. Uma vez que existem métodos distintos, é preciso definir *a priori* os requisitos que se pretendem atingir, sobretudo no que toca ao desempenho, tempo de desenvolvimento, custos e relação interação/experiência com o utilizador (UX/UI) (SemaIT, 2015). Será então necessário avaliar as características e implicações dos três métodos de desenvolvimento de aplicações móveis, assim como as suas vantagens e desvantagens, de forma a escolher aquele que melhor se adapta aos objetivos em mão (Tab. 2).

	Aplicações Nativas	Aplicações Web	Aplicações Multiplataforma Nativas	Aplicações Híbridas
Características	<ul style="list-style-type: none"> <li>Desenvolvidas para plataformas específicas, requerendo a utilização de uma linguagem de programação específica para cada um deles (<i>e.g.</i>, Java para Android, Objective-C ou Swift para iOS, C# para Windows Phone)</li> <li>Permitem a utilização de todas as funcionalidades nativas do dispositivo</li> </ul>	<ul style="list-style-type: none"> <li>Desenvolvidas (a maioria) utilizando Javascript, HTML5 e CSS</li> <li>Acesso através de um browser, que se adequa à apresentação de dados mas não permite a utilização das funcionalidades de um dispositivo móvel (<i>e.g.</i>, câmara)</li> <li>Não utilizam SDKS para a sua construção, recorrendo a frameworks e templates</li> </ul>	<ul style="list-style-type: none"> <li>Utilizam APIs criadas a partir de SDKS originais dos sistemas operativos, para construir as aplicações utilizando linguagens diferentes</li> <li>Utiliza as APIs nativas do sistema operativo, tendo acesso às funcionalidades nativas do dispositivo</li> </ul>	<ul style="list-style-type: none"> <li>Possuem um comportamento típico de aplicações nativas e são desenvolvidas utilizando tecnologias web (<i>i.e.</i>, tentam combinar o melhor das aplicações nativas e das aplicações web)</li> <li>Imitação das características nativas de aplicações nativas e acesso às funcionalidades do dispositivo, ainda que limitado, graças a frameworks e plugins apropriados</li> </ul>
Vantagens	<ul style="list-style-type: none"> <li>Melhor desempenho</li> <li>UX/UI superior</li> <li>Maior segurança graças a APIs nativos</li> <li>Suporte e documentação mais desenvolvidos</li> </ul>	<ul style="list-style-type: none"> <li>Custo de desenvolvimento reduzido</li> <li>Tempo de desenvolvimento curto</li> <li>Multiplataforma</li> </ul>	<ul style="list-style-type: none"> <li>Custo de desenvolvimento reduzido</li> <li>Tempo de desenvolvimento curtos</li> <li>Portabilidade (código único, para qualquer plataforma)</li> <li>Manutenção mais simples e comum a todas as plataformas</li> </ul>	<ul style="list-style-type: none"> <li>Custo de desenvolvimento reduzido</li> <li>Tempo de desenvolvimento curtos</li> <li>Portabilidade (código único, para qualquer plataforma)</li> <li>Rápida integração no mercado das aplicações móveis</li> <li>Manutenção mais simples e comum a todas as plataformas</li> </ul>
Desvantagens	<ul style="list-style-type: none"> <li>Custos e tempo de desenvolvimento e manutenção maiores</li> <li>Exige conhecimento específico da plataforma para a qual a aplicação é desenvolvida e da linguagem utilizada</li> <li>As convenções adotadas por cada linguagem de programação podem impedir uma equipa de criar a mesma aplicação para todas as plataformas</li> </ul>	<ul style="list-style-type: none"> <li>Incapaz de utilizar as funcionalidades do dispositivo</li> <li>Não pode ser instalada no dispositivo, nem registada numa App Store</li> <li>Apenas pode ser utilizada quando existir ligação à Internet</li> </ul>	<ul style="list-style-type: none"> <li>É necessário dispensar algum tempo programar a GUI (Graphic User Interface), de forma a que se obtenha um aspeto e sentimento de se estar numa aplicação nativa</li> <li>Desenvolver a componente UX (User Experience) específica para cada plataforma, utilizando apenas um código base, pode ser desafiante</li> </ul>	<ul style="list-style-type: none"> <li>Problemas de desempenho relacionados com alguns tipos de aplicações, sobretudo em casos onde é necessária a utilização intensa do processador</li> <li>Acesso limitado às funcionalidades do dispositivo e menor número de APIs relativamente às aplicações nativas</li> <li>Necessidade de plugins ou técnicas complexas para imitar as funcionalidades nativas de UI dos sistemas operativos</li> </ul>

**Tabela 2:** Comparação entre os diferentes tipos de aplicações móveis. (Abed, 2016; Alexseyenk, 2017; BloomIdea, 2016; Bristowe, 2015; Patil, 2016; Saccomani, 2017).

Como se pode observar pela Tabela 2, não existe um tipo de aplicação dominante em relação aos outros, uma vez que cada um apresenta os seus pontos fortes, que se revelam como pontos fracos nos outros tipos de aplicação, e vice-versa. O desenvolvimento de qualquer um destes três tipos de aplicação poderá ter sucesso e prosperar no mercado, desde que tenha uma direção bem definida e todos os recursos necessários bem estudados. No entanto, a escolha do desenvolvimento de aplicações híbridas será acertada se o ciclo de desenvolvimento pretendido

for curto e barato, tendo em conta que serão necessários menos programadores, reduzindo, conseqüentemente, os custos da aplicação. Apesar do desempenho das aplicações híbridas ficar sempre aquém daquela apresentada tipicamente pelas aplicações nativas, este pode ser satisfatório, tendo em conta que é possível ter acesso ao hardware dos dispositivos móveis e reproduzir a relação UX/UI das aplicações nativas, através de frameworks específicas e dedicação na pormenorização. Atualmente, existe uma grande diversidade destas frameworks, pelo que na próxima seção serão apresentadas as mais notórias e completas.

## 2.4.2. Frameworks para Aplicações Híbridas

### Adobe Cordova (PhoneGap)

Considerada como uma das frameworks mais populares para o desenvolvimento de aplicações multiplataforma, o Cordova permite o desenvolvimento descomplicado de aplicações híbridas utilizando apenas tecnologias web (Javascript, HTML5, CSS3) e suporta algumas das plataformas principais, como o Android, iOS e Windows Phone. O hardware do dispositivo móvel (*e.g.*, câmara, acelerómetro, GPS, etc.) consegue ser acedido utilizando HTML5 mas, em versões mais antigas de sistemas operativos, o acesso aos recursos nativos através de HTML5 pode não ser suportado. Para tal, o código HTML5 é inserido numa WebView nativa do dispositivo móvel, e o acesso aos recursos nativos é conseguido através de uma FFI (*Foreign Function Interface*). De modo a complementar esta framework, existem diversos plug-ins nativos, como aqueles que permitem a integração e aplicação da tecnologia de RA (Donga, 2017; Gajotres, 2017; Jscrambler, 2017).

### Ionic Framework

A Ionic Framework trata-se de uma das frameworks mais completas para o desenvolvimento de aplicações híbridas, utilizando tecnologias Javascript, HTML e CSS. O que a torna única é a integração da framework AngularJS, sendo capaz de fornecer experiências praticamente nativas, e o suporte para SASS, que irá enaltecer o aspeto visual da aplicação. Além disso, possui imensas funcionalidades nativas, sobretudo utilizando o plugin do Cordova e extensões TypeScript, e uma CLI (Command Line Interface) bastante prática para criar, desenvolver e testar as aplicações em qualquer plataforma nativa (Donga, 2017; Gajotres, 2017; Jscrambler, 2017).

### Framework 7

A Framework 7 é uma framework gratuita e “open-source”, destinada a desenvolver aplicações híbridas ou aplicações web, para as plataformas iOS e Android. Utilizando as tecnologias JavaScript, HTML5 e CSS, esta framework é utilizada sobretudo no desenvolvimento de protótipos, uma vez que permite a sua criação de forma rápida. De forma a fornecer a melhor experiência e simplicidade na sua produção, a componente de design apoia-se apenas nos materiais de design da iOS e da Google. Apesar de apresentar níveis de desempenho elevados, a Framework 7 terá que ser complementada com outras frameworks de JavaScript, como o Ionic Framework, para projetos de complexo superior (Donga, 2017; Gajotres, 2017; Jscrambler, 2017).

### Sensha Touch

Esta framework permite a criação de aplicações multiplataforma utilizando HTML5 e JavaScript, sendo a mais conceituada no mundo empresarial. Esta framework diferencia-se das restantes pois, além dos custos consideráveis da sua utilização, fornece um aspeto e ambiente nativo a todas as plataformas que ela suporta. Isto deve-se sobretudo à presença da ExtJS, uma das frameworks de JavaScript mais populares no mercado, que, através do alto desempenho e

experiência praticamente nativa que possibilita na criação das aplicações, complementa perfeitamente a Sencha Touch. Além disso, esta framework contém widgets, para implementar o visual e sensação quase nativa nas principais plataformas móveis, e uma ferramenta de “drag-and-drop” baseado em HTML5, com imensos templates prontos a serem aplicados (Donga, 2017; Gajotres, 2017; Jscrambler, 2017).

## **Kendo UI**

Baseada em HTML5, esta framework permite a criação de aplicações multiplataforma através da utilização considerável de jQuery, apesar de suportar JavaScript, HTML5 e CSS3. Além disso, caso seja necessário, é possível integrar frameworks como o AngularJS e o Bootstrap UI. Apesar de a sua utilização ser gratuita, não é o caso para algumas das widgets existentes. No entanto, existe uma boa documentação para esta framework, é relativamente fácil de utilizar para os utilizadores familiarizados com jQuery, e o desenvolvimento da componente UI é facilitado para qualquer plataforma (Donga, 2017; Gajotres, 2017; Jscrambler, 2017).

### 3. Planeamento e Metodologias

Este projeto foi dividido em duas partes, separadas pela defesa intermédia do mesmo. Nesta defesa, deu-se lugar à apresentação da proposta de dissertação e de todo o trabalho desenvolvido até ao momento, assim como aquele que seria realizado posteriormente. Tendo isto em conta, no início deste projeto, foi delineado um plano de trabalhos que seria realizado ao longo de toda uma primeira parte, que ocorreu durante o primeiro semestre, e de uma segunda parte, que teve lugar durante o segundo semestre.

#### 3.1. Planeamento

##### Primeiro Semestre

A primeira parte da dissertação, decorrida entre setembro e janeiro, consistiu, essencialmente, em investigação e na preparação para a fase de desenvolvimento, que teria lugar na segunda parte. Inicialmente, o plano de trabalhos proposto compreendia a análise das *frameworks* de RA existentes e, mediante uma recolha das mais promissoras, uma avaliação das mesmas nos mosaicos romanos de Conímbriga. Além destes encargos, seria também definida a arquitetura do sistema representativo da aplicação móvel e da interface web. Atendendo a este plano de trabalhos, o primeiro semestre foi devidamente planeado e dividido pelas seguintes tarefas:

##### 1. Revisão da literatura

Pesquisa, recolha e leitura de artigos online, sobre o conceito de Realidade Aumentada, a sua história, a sua evolução, as suas técnicas, as suas ferramentas e a sua aplicação em diversos contextos, com foco no Património Cultural.

##### 2. Análise do Estado da Arte

Escrita baseada nas leituras realizadas na primeira tarefa, de forma a abordar a temática da Realidade Aumentada desde o seu início até ao presente, com ênfase nas técnicas de rastreamento, nas áreas de aplicação e nos SDKs que permitem a sua integração em aplicações móveis. Por fim, foram comparados os diferentes tipos de aplicações móveis, de forma a justificar a escolha da natureza híbrida para este projeto.

##### 3. Investigação sobre os SDKs de Realidade Aumentada

Pesquisa e recolha de todos os SDKs existentes atualmente que permitem a integração da tecnologia de Realidade Aumentada no desenvolvimento de aplicações móveis, sendo feita, posteriormente, uma avaliação e comparação das suas características e funcionalidades.

##### 4. Escolha das tecnologias a serem utilizadas

Escolha da *framework* para o desenvolvido híbrido e dos SDKs que reuniram condições para a realização do projeto em mão.

##### 5. Teste dos SDKs de Realidade Aumentada

Elaboração de um protocolo de testes e execução dos mesmos sobre os SDKs escolhidos na tarefa anterior, em objetos semelhantes aos mosaicos romanos e num espaço aberto.

## 6. Análise dos resultados dos testes

Observação e avaliação dos resultados dos testes realizados na tarefa anterior.

## 7. Definição da arquitetura e dos requisitos do sistema

Especificação de alto nível da arquitetura do sistema a desenvolver e dos requisitos funcionais e não funcionais que este terá que cumprir.

## 8. Escrita do relatório da dissertação

Estruturação do documento da dissertação e escrita do seu conteúdo, mediante o ponto de situação das tarefas realizadas ao longo do 1º semestre.

Nesta primeira parte desta dissertação, não houve qualquer calendarização ou estimativas definidas para as tarefas planeadas, sendo que, embora seja uma mais valia, não prejudicou o fluxo de trabalho até à defesa intermédia, altura em que terminaria esta primeira fase. No entanto, o segundo semestre já teria um diagrama de Gantt (Fig. 32) com todas as tarefas representadas e devidamente agendadas.

## Segundo Semestre

A segunda parte da dissertação, decorrida entre fevereiro e julho, seria então dedicada ao desenvolvimento da aplicação móvel e da interface web, como havia sido proposto no plano de trabalhos inicial. As tarefas realizadas foram as seguintes:

### 1. Teste dos SDKs de Realidade Aumentada em Conímbriga

Elaboração de um protocolo de testes e execução dos mesmos sobre os SDKs *Wikitude*, *CraftAR* e *PixLive*, nos mosaicos romanos de Conímbriga.

### 2. Análise dos resultados dos testes anteriores

Análise dos vídeos dos testes realizados aos SDKs e extração dos seus resultados.

### 3. Definição do Modelo de Dados

Delineação dos dados necessários e elaboração do modelo de dados, através dos diagramas adequados.

### 4. Revisão das funcionalidades da aplicação móvel

Retificação das funcionalidades que a aplicação móvel deverá ou não possuir.

### 5. Redefinição dos requisitos e da arquitetura do sistema

Realização das alterações necessárias aos requisitos funcionais e não funcionais, assim como à arquitetura do sistema.

### 6. Implementação do *Backend*

#### a. Criação da base de dados

Criação da base de dados local.

#### b. Criação e configuração da API REST

Implementação da API REST, para consumo do *frontend*.

### 7. Desenvolvimento da aplicação móvel

**a. Implementação das Listagens**

Implementação da funcionalidade da aplicação móvel correspondente à listagem dos dados.

**b. Implementação da Realidade Aumentada**

Implementação da funcionalidade da aplicação móvel correspondente à experiência de RA sobre os mosaicos romanos de Conímbriga

**c. Implementação dos Pontos de Interesse**

Implementação da funcionalidade da aplicação móvel correspondente à geolocalização dos mosaicos romanos de Conímbriga, através dos Pontos de Interesse.

**8. Desenvolvimento da interface web**

Implementação da interface web para gestão dos conteúdos da aplicação móvel.

**9. Teste do protótipo final em Conímbriga**

Realização de testes de usabilidade, ao protótipo final da aplicação móvel, em Conímbriga.

**10. Análise dos resultados dos testes anteriores**

Avaliação do *feedback* obtido nos testes de usabilidade e dos vídeos gravados nessa sessão, de forma a extrair os resultados.

**11. Escrita do relatório final da dissertação**

Escrita do documento final desta dissertação.

Esta fase do projeto exigiu um planeamento mais rigoroso e um acompanhamento constante do estado atual do trabalho, razões pelas quais todas as tarefas foram devidamente projetadas e calendarizadas através de um diagrama de Gantt (Fig. 18).

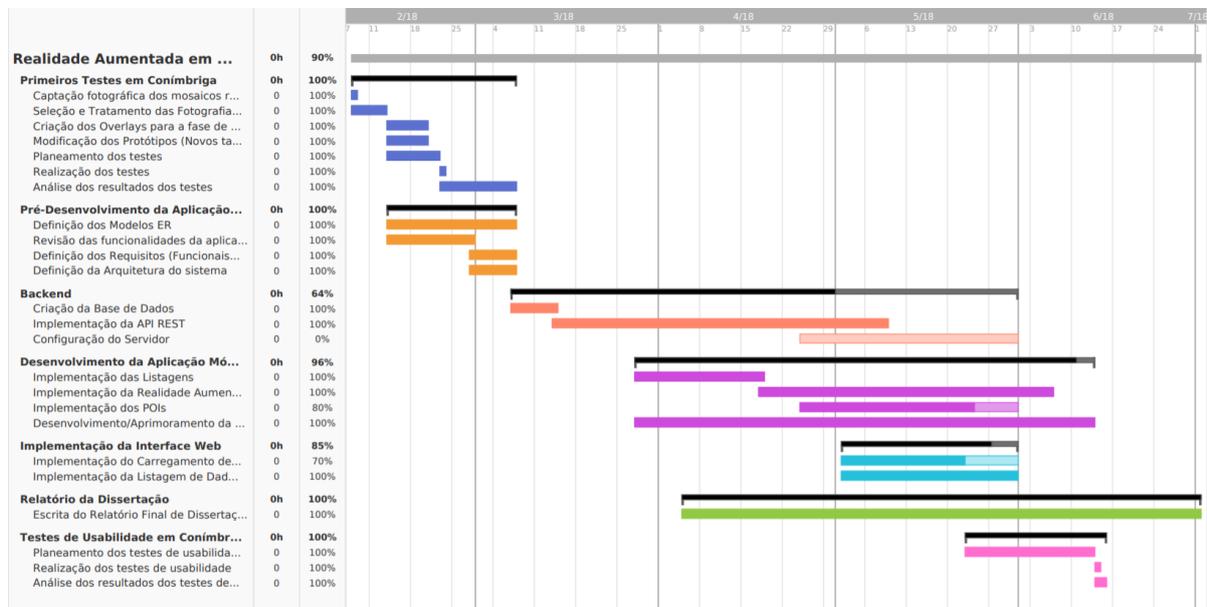


Figura 18: Diagrama de Gantt para o segundo semestre

Ao longo de toda a segunda parte do projeto, foram efetuadas as alterações necessárias no diagrama de Gantt, principalmente no reagendamento de algumas tarefas. De modo a observar essas modificações e estudar o seu impacto no desenvolvimento do trabalho, foi registado o progresso mensal (Anexo A). As tarefas que levaram mais tempo dizem respeito ao desenvolvimento da aplicação móvel e da interface web. A implementação do *Backend* levou mais tempo que o esperado, devido principalmente ao tempo de ambientação das ferramentas utilizadas. A implementação da RA na aplicação móvel levou também mais tempo a ser finalizada, uma vez que o SDK utilizado no momento teria sido descontinuado e houve a necessidade de escolher e utilizar um novo SDK de RA.

## 3.2. Metodologia e Ferramentas Utilizadas

### Primeiro Semestre

Como já foi mencionado anteriormente, no primeiro semestre foi realizada uma investigação acerca dos SDKs de RA existentes atualmente. Além de demonstrar o leque de opções disponíveis para a utilização em aplicações móveis, incluindo as características e funcionalidades de cada um, essa análise permitiu selecionar os SDKs mais adequados para este projeto. Para serem selecionados, os SDKs investigados (Tab. 1) teriam que respeitar os seguintes critérios:

1. **Plugin para desenvolvimento híbrido**

Foi proposto, para este projeto, o desenvolvimento de uma aplicação móvel híbrida, pelo que os SDKs terão que disponibilizar suporte nesse sentido.

2. **Rastreamento s/ marcadores**

Uma vez que não é pretendido interferir fisicamente com os mosaicos romanos de Conímbriga, através da utilização de marcadores artificiais, foi optada a técnica de RA baseada em características naturais. Um dos desafios propostos para esta dissertação prende-se precisamente com a aplicação da tecnologia de RA neste tipo de locais e objetos, sem a ajuda de marcadores, de forma a avaliar o seu desempenho.

3. **Rastreamento de imagens 2D**

Os mosaicos romanos de Conímbriga, alvos da tecnologia de RA, serão registados através de fotografias (imagens de duas dimensões). Tendo isso em conta, é essencial que o SDK seja capaz de detetar e rastrear este tipo de *targets*.

4. **Reconhecimento offline**

Este critério é bastante importante, na medida em que os visitantes de Conímbriga apenas necessitarão de utilizar a Internet uma vez, para descarregar os dados da aplicação. A partir desse momento, a experiência de RA pode decorrer normalmente em modo offline.

5. **Multi-Alvos**

Uma vez que é pretendido dinamizar várias partes de um mosaico, é essencial que o SDK permita a deteção e rastreamento simultâneo de vários *targets*, de forma a que seja possível observar e interagir com diversos *overlays* em tempo real.

6. **Licença Open-Source/Gratuita**

Uma vez que se está a desenvolver apenas o protótipo da aplicação móvel, o ideal será utilizar as versões gratuitas oferecidas pelos SDKs. Desta forma, será também

possível verificar a quantidade e diversidade de funcionalidades que estes oferecem neste tipo de licenças.

## 7. Sobreposição de *Overlays*

A dinamização dos mosaicos romanos de Conímbriga não fará sentido sem a possibilidade de serem sobrepostos por informação virtual (*i.e.*, *overlays*).

Primeiramente, os SDKs foram filtrados de acordo com o primeiro critério, restando apenas os seguintes: *Augment*, *CraftAR*, *EzAR*, *LayAR*, *PixLive*, *Vuforia* e *Wikitude*. De seguida, foi verificado se este conjunto de SDKs cumpria com os restantes critérios.

O SDK *EzAR* foi excluído de imediato uma vez que apenas se dedica à deteção de faces, não se adequando a este projeto. Quanto aos restantes SDKs, não foi possível confirmar o critério #5 (*i.e.*, Multi-Alvos) para os SDKs *Augment*, *LayAR* e *PixLive*, observando apenas os resultados. Tendo isso em conta, estes SDKs teriam que ser testados nesse sentido. Além disso, como foi mencionado na seção 2.2. deste documento, o último critério não foi tido em conta na análise dos SDKs, devido à incerteza da sua presença em grande parte dos SDKs. No entanto, os 3 SDKs restantes, *CraftAR*, *Vuforia* e *Wikitude*, confirmam esta funcionalidade através das suas respetivas documentações. Apesar disso, estes SDKs também teriam que ser testados nesse sentido, de modo a confirmarem esse critério.

Após alguma experimentação, foram descartados os SDKs *Augment*, *Layar*, *PixLive* e *Vuforia* (Tab. 3), e aceites os SDKs *Wikitude* e *CraftAR* (Tab. 4).

SDK	Razão para descarte	Fonte
PhoneGap Layar Plugin	Impossível de instalar, devido a erros nos seus servidores	<a href="https://www.layar.com/documentation/layar-sdk/phonegap-layar-plugin/">https://www.layar.com/documentation/layar-sdk/phonegap-layar-plugin/</a>
Augment Player SDK Cordova Plugin	Este plugin é limitado à utilização de objetos ( <i>i.e.</i> , <i>overlays</i> ) 3D, fornecidos pelo mesmo	<a href="https://developers.augment.com/cordova/">https://developers.augment.com/cordova/</a>
PixLive SDK Cordova Plugin	Não foi possível aplicar a sobreposição de imagens 2D, devido a dificuldades na implementação	<a href="https://vidinoti.github.io/cordova-plugin-PixLive/">https://vidinoti.github.io/cordova-plugin-PixLive/</a>
Cordova Vuforia SDK Plugin	Não foi possível aplicar a sobreposição de imagens 2D, uma vez que este plugin não contém a respetiva funcionalidade	<a href="https://github.com/mattrayner/cordova-plugin-vuforia/">https://github.com/mattrayner/cordova-plugin-vuforia/</a>

Tabela 3: SDKs descartados

SDK	Fonte
CraftAR Pro SDK Cordova Plugin	<a href="https://catchoom.com/documentation/pro-sdk/craftar-pro-cordova-plugin-for-ios-and-android/">https://catchoom.com/documentation/pro-sdk/craftar-pro-cordova-plugin-for-ios-and-android/</a>
Wikitude SDK Cordova Plugin	<a href="https://www.wikitude.com/external/doc/documentation/latest/phonegap/">https://www.wikitude.com/external/doc/documentation/latest/phonegap/</a>

Tabela 4: SDKs aceites

Realizada esta etapa, o próximo passo seria então a realização de um conjunto de testes bem definidos, de modo a avaliar o desempenho destes dois SDKs. Para a realização desses testes, foram utilizadas as seguintes ferramentas:

- **Hardware**
  - Smartphone Redmi Note 4
    - Android 7.0
    - Bússola, acelerómetro, giroscópio, sensor de proximidade
    - Câmera traseira (13 MP, f/2.0)
    - Vídeo (1080p@30fps)
- **Software**
  - Apache Cordova (v 7.1.0)
  - jQuery (v. 2.2.4, minified)
  - jQuery Mobile (v 1.4.0, minified)
  - Chrome DevTools
  - Android USB Debugging
- **SDKs**
  - Wikitude SDK Cordova Plugin (v. 7.1.0)
  - CraftAR Pro SDK Cordova Plugin (v. 2.4.3)

O smartphone utilizado para estes testes foi disponibilizado pelo orientador e viria a ser utilizado até ao fim do projeto. Este dispositivo móvel reúne todas as condições necessárias para o desenvolvimento de uma experiência de RA, contendo todos os sensores exigidos e uma câmara de boa qualidade. Quanto ao software, uma vez que a aplicação terá de ser de natureza híbrida, o Cordova foi a framework de eleição para os testes do primeiro semestre. Foram utilizadas as linguagens web (Javascript, HTML5 e CSS3), juntamente com o auxílio da biblioteca JQuery para facilitar o processo de desenvolvimento. De modo a depurar e testar este protótipo inicial de forma rápida e prática, foi ativada a opção de depuração no dispositivo móvel Android e foi, em conjunto, utilizada a ferramenta de depuração remota Chrome DevTools. É de realçar que, apesar de a aplicação ter que funcionar em diversas plataformas, nesta fase será testada apenas no sistema operativo Android. Por fim, os SDKs de RA que serão utilizados e testados estão também acima listados, com as respetivas versões utilizadas.

## Segundo Semestre

O plano de trabalhos do segundo semestre envolvia essencialmente o desenvolvimento da aplicação móvel e da interface web. No entanto, ainda não haviam sido realizados testes dos SDKs sobre os mosaicos romanos de Conímbriga. Esta seria então a primeira tarefa a ser realizada neste semestre e, dependendo dos seus resultados, seria escolhido apenas um SDK para utilizar no protótipo da aplicação móvel. Apesar de terem sido considerados apenas dois SDKs nos testes realizados no primeiro semestre, foi aceite mais um para a realização destes testes: o *PixLive*. A razão para a consideração deste SDK prende-se com o facto de já ter sido possível implementar com sucesso a funcionalidade que diz respeito à sobreposição de imagens 2D. Dito isto, após a realização dos testes dos SDKs em Conímbriga, seria então escolhido aquele que obtivesse os resultados mais satisfatórios, para utilizar e integrar na aplicação móvel.

No que toca às ferramentas e tecnologias que viriam a ser utilizadas no segundo semestre, elas são as seguintes:

- **Hardware**
  - Smartphone Redmi Note 4
- **Software**
  - Ionic Framework (v. 3.20.0)

- Cordova-SQLite-Storage (v. 2.3.1)
  - Angular (v. 5.2.9)
  - jQuery (v. 2.1.0, minified)
  - jQuery Mobile (v. 1.4.5, minified)
  - Django (v. 2.0.3)
    - Django REST Framework (v. 3.7.7)
  - MySQL (v. 5.7.21)
  - Chrome DevTools
  - Android USB Debugging
- SDKs
  - Wikitude SDK Cordova Plugin (v. 7.1.0)
  - CraftAR Pro SDK Cordova Plugin (v. 2.4.3)

Em termos de *hardware*, o *smartphone* utilizado o desenvolvimento da aplicação da móvel e realização dos testes manteve-se o mesmo. Para a elaboração do protótipo da aplicação móvel, teriam que ser utilizadas linguagens web (*i.e.*, HTML, CSS e JavaScript), uma vez que o desenvolvimento híbrido assim o exigia. Tendo isso em conta, primeiramente, foi feita uma análise às *frameworks* para o desenvolvimento de aplicações híbridas (Tab. X). A escolha acabou por recair sobre o **Ionic**, por diversas razões. Esta *framework* é *open source* e gratuita, disponibilizando um conjunto vasto de elementos UI que podem ser facilmente aplicados e modificados, poupando o tempo que se gastaria a programa-los de raiz. Outra grande vantagem desta é a integração da *framework* de *frontend* Angular, que é extremamente orientada aos componentes, torna o código fácil de ler e reutilizar e, de modo geral, fornece robustez à aplicação móvel. Além disso, o Ionic integra também a *framework* Cordova, permitindo assim o funcionamento da aplicação em diversas plataformas e a possibilidade de utilizar os seus plugins. Por fim, por ser das *frameworks* de desenvolvimento de aplicações híbridas mais populares há já algum tempo, contém bastante documentação e apoio online (INKONIQ, 2017).

Foi também feita uma pesquisa sobre *frameworks* para o desenvolvimento *web*, do lado do servidor (*backend*), de modo a facilitar o desenvolvimento nessa componente do sistema e, ao mesmo, aprender uma nova *framework*. As opções mais populares, no momento desta pesquisa, eram as *frameworks* Django e Ruby On Rails. A *framework* Ruby on Rails foi desde logo descartada, uma vez que o autor desta dissertação estava familiarizado com a linguagem Python (utilizada pelo Django) e não seria viável estar a aprender uma nova linguagem durante a fase de desenvolvimento da aplicação móvel. Dito isto, a *framework web Django* é também *open source* e gratuita, e possibilita um desenvolvimento rápido e pragmático do lado do servidor, permitindo mudar o foco para o *frontend*. Além de escalabilidade e segurança, esta *framework* inclui a sua própria biblioteca de ORM (Object-Relational Mapping), facilitando as ações relacionadas com a base de dados. Esta é das *frameworks web* mais populares do momento e, por isso, também oferece bastante documentação e suporte online (Hansen, 2017).

De maneira a construir a API REST, foi aproveitado o facto de o Django possuir uma biblioteca para o efeito: a **Django REST Framework**. Esta biblioteca permite a criação e manutenção de forma simples e rápida de uma API REST, fornecendo funcionalidades como autenticação (básica, baseada em JWT, etc.). Novamente, esta *framework* apresenta uma documentação basta completa online e é popular entre os utilizadores de Django. Como será explicado mais à frente neste documento, é graças a esta *framework* que a interface web consegue estar protegida, através da autenticação baseada em *JSON Web Tokens* (JWT). Relativamente à base de dados, foi utilizada a linguagem **MySQL**, por ser aquela com que o autor mais se familiariza. Além disso, esta é uma linguagem simples de utilizar, facilmente configurável, compatível com o Django e contém bastante documentação e apoio online.

Para a integração da tecnologia de RA, foi utilizado o SDK da **Wikitude**. Apesar de, após os testes aos SDKS realizados no segundo semestre, a escolha do SDK ter recaído para o CraftAR, este deixou de ser opção a partir do momento que foi descontinuado, durante o desenvolvimento da aplicação. O plano de mitigação para este risco sugere que seja escolhido o segundo SDK com melhores resultados nos testes do segundo semestre que, neste caso, seria o PixLive. No entanto, este SDK é bastante limitado em termos de funcionalidades e de documentação, não correspondendo aos requisitos e às expectativas do projeto. Sendo assim, a escolha do SDK recaiu para a Wikitude, que por sua vez é um SDK bastante completo e popular no mercado, fornecendo as mais diversas características (Tab.1) e funcionalidades para a criação de uma experiência de RA interessante.

Por fim, para o desenvolvimento da interface web, foi escolhida a *framework* **Angular** para o *frontend*, uma vez que esta já estaria a ser utilizada através da *framework* Ionic, no desenvolvimento da aplicação móvel. O *backend* é o mesmo que aquele utilizado na aplicação móvel, no sentido em que ambos consomem a API REST integrada no Django. Na próxima seção, será descrito o processo de trabalho, incluindo as metodologias e técnicas aplicada, levado a cabo no desenvolvimento da aplicação móvel e para a interface web.

### 3.3. Riscos

Apesar de, no início desta dissertação, não terem sido previstos os possíveis riscos, é importante tê-los em conta, uma vez que poderão vir a comprometer a concretização da mesma, com sucesso. No início do segundo semestre, e após algumas recomendações tidas em conta na defesa intermédia, foram definidos os seguintes riscos:

<b>Nome do Risco</b>	Estimação errada no planeamento de tarefas
<b>Descrição</b>	Ao definir e calendarizar certas tarefas, há a possibilidade de atribuir menos tempo do que ela realmente necessita.
<b>Consequência</b>	Atraso de outras tarefas
<b>Probabilidade</b>	Alta
<b>Impacto</b>	Alto
<b>Plano de Mitigação</b>	Atribuir prioridades a determinados requisitos, de modo a que aqueles que não sejam essenciais possam ser adiantados ou até mesmo não implementados

**Tabela 5:** Risco #1

<b>Nome do Risco</b>	Problema com o SDK
<b>Descrição</b>	O SDK utilizado na aplicação móvel poderá vir a ter problemas como descontinuidade ou falta de pequenas funcionalidades que fazem a diferença no projeto.
<b>Consequência</b>	Caso ocorra este risco, pode afetar significativamente o desenvolvimento da aplicação móvel, quer em termos de planeamento e calendarização, quer em termos de desempenho, uma vez que terá que ser escolhido outro SDK.
<b>Probabilidade</b>	Média
<b>Impacto</b>	Alto
<b>Plano de Mitigação</b>	Caso o SDK seja descontinuado ou não consiga satisfazer todos os requisitos essenciais do projeto, será escolhido o segundo SDK que obteve resultados mais satisfatórios nos testes

	aos SDKs. Se este segundo também apresentar problemas, será escolhido o terceiro SDK nas condições mencionadas anteriormente.
--	---

**Tabela 6:** Risco #2

O risco #1 (Tab. 5), como era esperado, ocorreu com alguma frequência, devido sobretudo à curva de aprendizagem de algumas ferramentas e funcionalidades utilizadas neste projeto, que apresentou ser maior do que o esperado. Por esta razão, e devido também a algumas estimativas desacertadas, determinadas tarefas tiveram que ser prolongadas, causando o atraso de outras. Já o risco #2 (Tab. 6) não era tão expectável e, no entanto, aconteceu numa fase delicada do projeto. Durante a utilização do SDK *CraftAR* no desenvolvimento da aplicação móvel, este foi descontinuado. Já não sendo possível utilizar este SDK no projeto, foi então aplicado o plano de mitigação e escolhido o SDK *Wikitude*.

## 4. Testes dos SDKs

Neste capítulo encontram-se descritos os testes que foram realizados aos SDKs de RA, de modo a avaliar os seus desempenhos e optar, no fim, pelo mais adequado para o projeto em mão. A metodologia utilizada para a seleção dos SDKs a serem testados pode ser consultada na seção 3.2. deste documento. Primeiramente, serão apresentados os testes decorridos no primeiro semestre, incluindo os respetivos resultados, onde foram testados os SDKs *Wikitude* e *CraftAR*, sobre a calçada portuguesa. Além disso, serão também apresentados os testes decorridos no início do segundo semestre, incluindo os respetivos resultados, onde foram testados os SDKs *Wikitude*, *CraftAR* e *PixLive*, sobre os mosaicos romanos de Conímbriga.

### 4.1. Testes na Calçada Portuguesa

Terminada a recolha dos SDKs com as características e funcionalidades exigidas para este projeto (Tab. 4), seria então necessário testá-los, de forma a avaliar e comparar o desempenho de cada um e, no fim, escolher aquele que reunisse os melhores resultados. Para estes testes, foram considerados os SDKs da *Wikitude* e do *CraftAR*. Uma vez que se tratava ainda de uma fase inicial da dissertação, a ideia seria de testar primeiramente num ambiente, condições e objetos minimamente semelhantes às de Conímbriga, o local de estudo deste projeto. Para tal, o local de testes teria que apresentar as seguintes características:

- Aberto (i.e., ao ar livre)
- Luminosidade variante
  - Forte (céu limpo)
  - Fraca (céu coberto)
  - Irregular (presença de sombras)

Quanto aos objetos presentes nesse local, nos quais iria ser aplicada a tecnologia de Realidade Aumentada, teriam que seguir os seguintes requisitos:

- Situados no solo
- Mosaicos ou elementos com constituição semelhante à dos mosaicos
- Texturas e padrões minimamente complexos

Felizmente, não foi difícil encontrar um local que reunisse todas estas condições, uma vez que existe um tipo de revestimento de piso extremamente comum (e originário) no nosso país: a calçada portuguesa. A colocação estratégica de pedras de cores diferentes origina padrões decorativos e um efeito visual análogo ao dos mosaicos romanos de Conímbriga. Tendo isto em conta, o local escolhido para a realização dos primeiros testes aos SDKs foi a Porta Férrea de Coimbra (Fig. 18). A entrada principal para a Universidade de Coimbra encontra-se num espaço aberto, onde a luminosidade é condicionada pelas condições climatéricas e pela envolvência de edifícios, que podem causar a presença de sombras sobre o solo. Este último, por sua vez, é constituído por calçada portuguesa com uma ilustração composta por elementos distintos, onde se evidencia a Rainha Santa Isabel (Fig. 19).



Figura 19: Porta Férrea de Coimbra (esquerda) e ilustrações da calçada portuguesa (direita).

## Variáveis

Além do local dos testes, foram também definidas as variáveis que iriam ser manipuladas (**independentes**) assim como aquelas que seriam consequências dessas manipulações (**dependentes**). Seguem-se então, primeiramente, as variáveis **independentes** que foram utilizadas nestes testes:

- **SDKs**
  - CraftAR Pro SDK Cordova Plugin
  - Wikitude SDK Cordova Plugin

- **Targets**

Foram escolhidos 5 *targets* (Fig. 20) para a realização destes testes, todos eles fotografados na vertical e juntos uns dos outros, à exceção de um *target* representado por uma estrela (Fig. 21), que se situava ligeiramente distante dos restantes *targets*.



Figura 20: (esquerda para a direita) Targets “Rainha”, “Mocho”, “Pergaminho” e “Guitarra”.



Figura 21: Target “Estrela”.

- **Overlays**

Os *overlays* utilizados (Fig. 22) utilizados nestes testes tratam-se de imagens 2D criadas no Photoshop e podem ser descritas da seguinte forma:

1. **Nome da imagem:** Posição  
**Constituição:** Imagem “sólida” (100% opaca), que surge em posições estratégicas do *target*.
2. **Nome da imagem:** Contorno  
**Constituição:** Imagem representada por uma linha azul, que contorna o *target*.
3. **Nome da imagem:** Preenchimento  
**Constituição:** Imagem com 50% de opacidade, que surge em posições estratégicas do *target*, de modo a dar um efeito de preenchimento.



Figura 22: (esquerda para a direita) Imagens virtuais “Posição”, “Contorno” e “Preenchimento”.

- **Inclinação**

O smartphone pode ser manipulado de modo a que fique com diferentes inclinações, relativamente aos *targets*:

- Paralelo
- Inclinado para cima/baixo
- Inclinado para o lado esquerdo/lado direito

- **Distância**

O smartphone pode encontrar-se a diferentes distâncias, relativamente aos *targets*:

- Baixa (*i.e.*, junto ao *target*)
- Média (*i.e.*, 1 a 3 passos à retaguarda)
- Alta (*i.e.*, 4 ou mais passos à retaguarda)

- **Luminosidade**

Apesar de, num espaço aberto, a luminosidade não ser diretamente manipulável pelo utilizador, esta pode afetar consideravelmente o desempenho dos SDKs e, consequentemente, o valor das variáveis dependentes. Foram, portanto, consideradas três condições de luminosidade:

- Luminosidade forte (céu limpo)
- Luminosidade fraca (céu coberto)
- Luminosidade irregular (presença de sombras)

Além das variáveis independentes descritas anteriormente, foram definidas as seguintes variáveis **dependentes**:

- **Deteção**

Esta variável dependente refere-se à capacidade de deteção de um *target*. Um *target* pode ser:

- Detetado
  - Surge a imagem sobreposta correspondente.
- Não Detetado
  - Nada acontece.

- **Precisão do alinhamento**

Esta variável dependente refere-se à precisão com que uma imagem sobreposta surge em relação ao respetivo *target*. A imagem pode:

- Surgir no local esperado
  - Observar Figuras 23-25.
- Surgir fora do local esperado

- **Estabilidade**

Esta variável dependente refere-se ao desempenho de rastreamento de um *target*, face à imagem sobreposta no mesmo. O *target* pode apresentar-se como:

- Estável
  - A imagem sobreposta correspondente não desaparece.
- Instável
  - A imagem sobreposta correspondente desaparece ocasionalmente.
- Muito instável
  - A imagem sobreposta correspondente desaparece frequentemente.



Figura 23: Resultados esperados para a sobreposição dos overlays “Posição”, em todos os targets.



Figura 24: Resultados esperados para a sobreposição dos overlays “Contorno”, em todos os targets.



Figura 25: Resultados esperados para a sobreposição dos overlays “Preenchimento”, em todos os targets.

## Descrição dos testes

Para a execução destes testes, a interface do protótipo em questão será simples e prática, uma vez que o objetivo será somente avaliar o desempenho da tecnologia de RA fornecida pelos SDKs. Sendo que foram escolhidos dois SDKs, foram criadas duas aplicações, ambas com a mesma interface: uma janela com o título respetivo do SDK a ser testado, e três botões, que representam os três tipos de *overlays* criados para estes testes (Fig. 26). Após a seleção da opção desejada, será despoletada uma WebView onde decorre a experiência de RA, utilizando-se a câmera traseira para visualizar o cenário real, os *targets* e as respetivas imagens virtuais.



Figura 26: Ecrã inicial do protótipo desenvolvido com o SDK da Wikitude.

Segue-se então a listagem dos testes que foram realizados para a experiência de RA:

1. Apontar a câmera, na vertical, para um *target* de cada vez e verificar se a respetiva imagem sobreposta aparece.
2. Apontar a câmera, na vertical, para um *target* de cada vez e verificar se a respetiva imagem sobreposta aparece no local esperado e com as dimensões esperadas.
  - 2.1. Se não aparecer no local esperado, registar o novo local onde este surgiu.
  - 2.2. Se não aparecer com as dimensões esperadas, registar as novas dimensões com as quais este surgiu.
3. Apontar a câmera para um *target* de cada vez, na vertical, e dar uma volta completo em torno do mesmo, verificando se a imagem sobreposta desaparece no processo.
  - 3.1. Se desaparecer, contar o número de vezes que desaparece.
4. Apontar a câmera para um *target* de cada vez e afastá-la lentamente em qualquer direção, de modo a que a imagem sobreposta desapareça progressivamente do campo de visão da câmera, mas não desapareça na íntegra.

- 4.1. Se desaparecer, registar a porção do target que ainda ficou no campo de visão da câmara.
5. Apontar a câmara para um *target* de cada vez caminhar para trás, afastando lenta e progressivamente a câmara do *target* e verificar se a imagem sobreposta desaparece.
  - 5.1. Se desaparecer, registar o número de passos dados até esse momento.
6. Apontar a câmara para um *target* de cada vez, na vertical, e colocar um objeto entre o *target* e a câmara (e.g., mão), de modo a tapar o *target* e verificar se a imagem sobreposta desaparece.

## Resultados dos Testes

Os resultados dos testes foram descritos de acordo com a listagem de testes definidos, de modo a ser possível confrontar os resultados correspondentes de cada protótipo. Importa mencionar que apenas foi possível realizar os testes em condições de luminosidade fraca (*i.e.*, céu coberto), devido às condições climáticas na altura. Segue-se então uma exposição e análise dos resultados retirados dos testes aos SDKs:

- **Teste #1:** Apontar a câmara, na vertical, para um *target* de cada vez e verificar se a respetiva imagem sobreposta aparece.
  - **Exemplo:**



Figura 27: Exemplo ilustrativo do Teste #1, com o antes (imagem à esquerda) e o depois (imagem à direita).

	Target “Estrela”	Target “Guitarra”	Target “Mocho”	Target “Pergaminho”	Target “Rainha”
<b>Wikitude</b>	A imagem sobreposta nem sempre aparece nos primeiros instantes.	A imagem sobreposta não surge de todo.	A imagem sobreposta pode demorar até 2 segundos a surgir.	A imagem sobreposta surge de imediato.	A imagem sobreposta pode demorar até 2 segundos a surgir.
<b>CraftAR</b>	A imagem sobreposta surge de imediato.	A imagem sobreposta surge de imediato.	A imagem sobreposta surge de imediato.	A imagem sobreposta surge de imediato.	A imagem sobreposta surge de imediato.

Tabela 7: Resultados do Teste #1.

- **Teste #2:** Apontar a câmara, na vertical, para um *target* de cada vez e verificar se a respetiva imagem sobreposta aparece no local esperado e com as dimensões esperadas.
  - **Exemplo:**



Figura 28: Exemplo ilustrativo do Teste #2, com o antes (imagem à esquerda) e o depois (imagem à direita).

	Target “Estrela”	Target “Guitarra”	Target “Mocho”	Target “Pergaminho”	Target “Rainha”
<b>Wikitude</b>	A imagem sobreposta aparece no local esperado e com as dimensões esperadas.	A imagem sobreposta não surge de todo.	A imagem sobreposta aparece no local esperado e com as dimensões esperadas.	A imagem sobreposta aparece no local esperado e com as dimensões esperadas.	A imagem sobreposta aparece no local esperado e com as dimensões esperadas.
<b>CraftAR</b>	A imagem sobreposta aparece no local esperado e com as dimensões esperadas.	A imagem sobreposta “Posição” aparece no local esperado e com as dimensões esperadas. Já a imagem sobreposta “Contorno” aparenta ter as dimensões certas, mas aparece ligeiramente deslocada para baixo. Por fim, a imagem “Preenchimento” aparece com alguns defeitos, sendo incerto confirmar o seu local e dimensões exatas.	A imagem sobreposta aparece no local esperado e com as dimensões esperadas, à exceção da imagem “Preenchimento”, que aparece com alguns defeitos, sendo incerto confirmar o seu local e dimensões exatas.	A imagem sobreposta aparece no local esperado e com as dimensões esperadas, à exceção da imagem “Preenchimento”, que aparece com alguns defeitos, sendo incerto confirmar o seu local e dimensões exatas.	A imagem sobreposta aparece no local esperado e com as dimensões esperadas, à exceção da imagem “Preenchimento”, que aparece com alguns defeitos, sendo incerto confirmar o seu local e dimensões exatas.

Tabela 8: Resultados do Teste #2.

- **Teste #3:** Apontar a câmara, na vertical, para um *target* de cada vez, e dar uma volta completo em torno do mesmo, verificando se a imagem sobreposta desaparece no processo.
  - **Exemplo:**



Figura 29: Exemplo ilustrativo do Teste #3, com o antes (imagem à esquerda) e o depois (imagem à direita).

	Target “Estrela”	Target “Guitarra”	Target “Mocho”	Target “Pergaminho”	Target “Rainha”
<b>Wikitude</b>	A imagem sobreposta desaparece, em média, 7 vezes.	A imagem sobreposta não surge de todo.	A imagem sobreposta desaparece, em média, 4 vezes.	A imagem sobreposta desaparece, em média, 4 vezes.	A imagem sobreposta desaparece a partir do momento que é dado um ângulo de 90° em torno do <i>target</i> .
<b>CraftAR</b>	A imagem sobreposta não desaparece.	A imagem sobreposta não desaparece.	A imagem sobreposta não desaparece.	A imagem sobreposta não desaparece.	A imagem sobreposta desaparece, em média, 2 vezes.

Tabela 9: Resultados do Teste #3.

- **Teste #4:** Apontar a câmera para um *target* de cada vez e afastá-la lentamente em qualquer direção, de modo a que a imagem sobreposta desapareça progressivamente do campo de visão da câmera, mas não desapareça na íntegra.
  - **Exemplo:**



Figura 30: Exemplo ilustrativo do Teste #4, com o antes (imagem à esquerda) e o depois (imagem à direita).

	Target “Estrela”	Target “Guitarra”	Target “Mocho”	Target “Pergaminho”	Target “Rainha”
<b>Wikitude</b>	A imagem sobreposta desaparece apenas quando cerca de metade do <i>target</i> desaparece do campo de visão da câmera.	A imagem sobreposta não surge de todo.	A imagem sobreposta desaparece apenas quando cerca de metade do <i>target</i> desaparece do campo de visão da câmera	A imagem sobreposta desaparece apenas quando cerca de metade do <i>target</i> desaparece do campo de visão da câmera	A imagem sobreposta desaparece apenas quando cerca de metade do <i>target</i> desaparece do campo de visão da câmera
<b>CraftAR</b>	A imagem sobreposta desaparece praticamente no momento em que o <i>target</i> desaparece do campo de visão da câmera.	A imagem sobreposta desaparece praticamente no momento em que o <i>target</i> desaparece do campo de visão da câmera.	A imagem sobreposta desaparece praticamente no momento em que o <i>target</i> desaparece do campo de visão da câmera.	A imagem sobreposta desaparece praticamente no momento em que o <i>target</i> desaparece do campo de visão da câmera.	A imagem sobreposta desaparece apenas quando cerca de metade do <i>target</i> desaparece do campo de visão da câmera.

Tabela 10: Resultados do Teste #4.

- **Teste #5:** Apontar a câmera para um *target* de cada vez caminhar para trás, afastando lenta e progressivamente a câmera do *target* e verificar se a imagem sobreposta desaparece.
  - **Exemplo:**



Figura 31: Exemplo ilustrativo do Teste #5, com o antes (imagem à esquerda) e o depois (imagem à direita).

	Target “Estrela”	Target “Guitarra”	Target “Mocho”	Target “Pergaminho”	Target “Rainha”
<b>Wikitude</b>	Foram dados, em média, 5 passos até à imagem sobreposta desaparecer.	A imagem sobreposta não surge de todo.	Foram dados, em média, 6 passos até à imagem sobreposta desaparecer.	Foram dados, em média, 5 passos até à imagem sobreposta desaparecer.	Foram dados, em média, 6 passos até à imagem sobreposta desaparecer.
<b>CraftAR</b>	Foram dados, em média, 6 passos até à imagem sobreposta desaparecer.	Foram dados, em média, 6 passos até à imagem sobreposta desaparecer.	Foram dados, em média, 6 passos até à imagem sobreposta desaparecer.	Foram dados, em média, 6 passos até à imagem sobreposta desaparecer.	Foram dados, em média, 7 passos até à imagem sobreposta desaparecer.

Tabela 11: Resultados do Teste #5.

- **Teste #6:** Apontar a câmara para um *target* de cada vez, na vertical, e colocar um objeto entre o *target* e a câmara (*e.g.*, mão), de modo a tapar o *target* e verificar se a imagem sobreposta desaparece.
  - **Exemplo:**



Figura 32: Exemplo ilustrativo do Teste #6, com o antes (imagem à esquerda) e o depois (imagem à direita).

	Target “Estrela”	Target “Guitarra”	Target “Mocho”	Target “Pergaminho”	Target “Rainha”
<b>Wikitude</b>	A imagem sobreposta apenas desaparece com movimentos bruscos.	A imagem sobreposta não surge de todo.	A imagem sobreposta apenas desaparece com movimentos bruscos.	A imagem sobreposta apenas desaparece com movimentos bruscos.	A imagem sobreposta apenas desaparece com movimentos bruscos.
<b>CraftAR</b>	A imagem sobreposta suporta movimentos bruscos e não desaparece com facilidade.	A imagem sobreposta suporta movimentos bruscos e não desaparece com facilidade.	A imagem sobreposta suporta movimentos bruscos e não desaparece com facilidade.	A imagem sobreposta suporta movimentos bruscos e não desaparece com facilidade.	A imagem sobreposta suporta movimentos bruscos e não desaparece com facilidade.

Tabela 12: Resultados do Teste #6.

A execução dos testes foi filmada através da aplicação *ScreenRecorder*, sendo que, no final, foi retirada toda a informação dos vídeos resultantes que fosse relevante para os testes em questão. Por fim, para complementar os resultados observados anteriormente, podem ser retiradas as seguintes conclusões:

- O SDK da Wikitude apresenta dificuldade em detetar e rastrear alguns *targets*, nomeadamente aqueles que aparentam ser simétricos (e.g., “Estrela” e “Guitarra”). Enquanto que o *target* “Estrela” ainda é possível de ser detetado, o mesmo não se pode dizer acerca do *target* “Guitarra”. Por outro lado, o SDK do CraftAR apresenta bom desempenho na deteção de todos os *targets*.
- É possível observar defeitos bastante perceptíveis nas imagens “Preenchimento”, utilizando o SDK do CraftAR. Utilizando o SDK da Wikitude, este caso não se verifica, pelo que as imagens sobrepostas aparecem como o esperado.
- O SDK da Wikitude apresenta dificuldade em rastrear os *targets*, sempre que é efetuada uma rotação sobre os mesmos. A imagem sobreposta desaparece mais vezes no *target* “Estrela” e no *target* “Rainha”, sendo que neste último a imagem sobreposta deixa de aparecer depois de se ter rodado 90° sobre o *target*. Por outro lado, o SDK do CraftAR apresenta um ótimo desempenho no rastreamento dos *targets* nesta situação, onde a imagem sobreposta não desaparece (à exceção de *targets* de dimensões grandes, como o “Rainha”).
- À medida que os *targets* desaparecem do campo de visão da câmara do *smartphone*, as respetivas imagens sobrepostas perduram por mais tempo utilizando o SDK do CraftAR.
- À medida que o utilizador (i.e., a câmara) se distancia dos *targets*, as respetivas imagens sobrepostas perduram praticamente na mesma distância final verificada utilizando ambos os SDKs.
- O CraftAR apresenta melhor estabilidade dos *overlays* face a obstáculos estáticos e em movimento, incluindo movimentos bruscos.
- Em *targets* de dimensões consideráveis, como “A Rainha”, surge a necessidade de elevar mais a câmara de forma a que este apareça na totalidade e consiga ser detetado, por ambos os SDKs.

Estes resultados serão tidos em conta aquando a realização de novos testes no segundo semestre, desta feita em Conímbriga e já sobre os mosaicos romanos.

## 4.2. Testes em Conímbriga

No primeiro semestre, foram realizados testes aos seguintes SDKs de RA: *Wikitude* e *CraftAR*. Após a realização dos mesmos, seria agora altura de realizar novos testes, mas, desta vez, no local de estudo (ruínas de Conímbriga) e tendo os seus mosaicos romanos como alvos da tecnologia de RA. O objetivo principal para a realização destes testes prende-se com a escolha do SDK que viria a ser utilizado no desenvolvimento da aplicação móvel. Aquele que obtivesse os resultados mais satisfatórios seria, portanto, o SDK escolhido para este projeto.

Apesar de, nos testes do 1º semestre, terem sido utilizados apenas dois SDKs, nestes testes posteriores já seria incluído um novo SDK: *PixLive*, da *Catchboom*. Este último SDK havia sido incluído no leque de SDKs descartados (Tab. 3), devido à incapacidade, na altura, de implementar *overlays* sobre os *targets*. No entanto, após mais algum tempo de exploração do SDK, e já depois dos testes do 1º semestre, foi possível então sobrepor *overlays* nos *targets* detetados. Tendo isto em conta, para os testes do 2º semestre, foram considerados estes três SDKs de RA: *Wikitude*, *CraftAR* e *PixLive*.

Houve uma primeira deslocação às ruínas de Conímbriga, de modo a recolher registos de todos os mosaicos romanos que estivessem ao alcance visível dos visitantes. No final, foram registados 31 mosaicos. Este registo fotográfico foi obtido a partir da máquina fotográfica Canon EOS 1200D e do smartphone Samsung Galaxy J5. Enquanto que, com a câmara, as fotografias foram tiradas ao nível do visitante, com o *smartphone*, por outro lado, foi utilizado um acessório para suportar o *smartphone* a uma distância e elevação superiores (Fig. 33). Deste modo, o *smartphone* foi colocado por cima dos mosaicos (o mais vertical possível), e os registos seriam obtidos com o auxílio de um temporizador.



Figura 33: Aquisição fotográfica dos mosaicos romanos

## Variáveis

Para estes testes, foram definidas as seguintes variáveis **independentes**:

- **SDKs**
  - CraftAR Pro SDK Cordova Plugin
  - Wikitude SDK Cordova Plugin
  - PixLive SDK Cordova Plugin
  
- **Targets**

A escolha dos targets para estes testes foi mais rigorosa relativamente à dos *targets* dos testes do 1º semestre. Foi tida em conta a funcionalidade de avaliação de *targets* (Fig. 34), fornecida pelas interfaces web da Wikitude e do CraftAR, que prevê a qualidade de rastreamento que os respetivos *targets* irão ter. Esta avaliação, representada por estrelas, é atribuída de 1 a 3 (Wikitude) e de 1 a 5 (CraftAR), sendo que os *targets* que obtiverem 1 estrela dificilmente serão detetados pelo respetivo SDK e, por outro lado, aqueles que obtiverem a avaliação máxima terão grandes hipóteses de serem detetados. Embora tenham sido registados mosaicos com algumas sombras presentes, estes foram considerados na mesma para a realização destes testes, de forma a verificar se estes seriam detetáveis ou não. Apesar de o PixLive também possuir uma interface web, esta não contém qualquer funcionalidade para avaliar os *targets* submetidos.

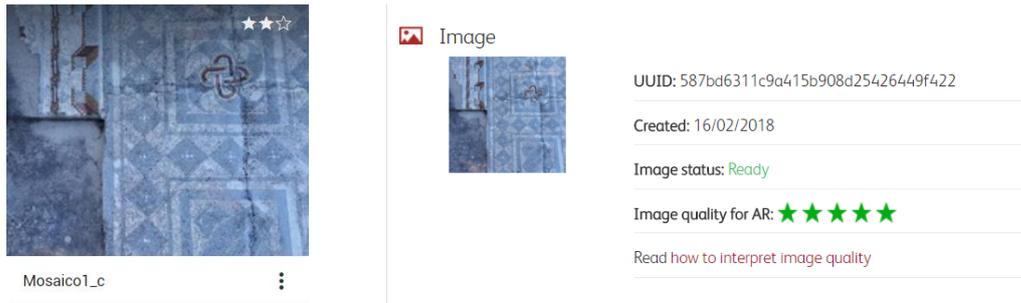


Figura 34: Exemplo do sistema de avaliação de targets da Wikitude (imagem à esquerda) e do CraftAR (imagem à direita)

Após a seleção de *targets* em cada mosaico, o próximo passo seria a sua submissão em ambas as interfaces web. Enquanto que, no plano *Free Trial*, a interface da Wikitude aceita até 1GB de *targets*, a interface do CraftAR apenas aceita 20 *targets* no máximo, por cada coleção. Esta limitação levou à redução do número de *targets* escolhidos, que passaria então a ser 20 (Fig. 35). Foram escolhidos *targets* de mosaicos diferentes (*i.e.*, não haveria mais do que 1 *target* do mesmo mosaico) e com diversas avaliações, incluindo 2 *targets* com 1 estrela (na interface da Wikitude) A listagem completa pode ser observada na Tabela 13.

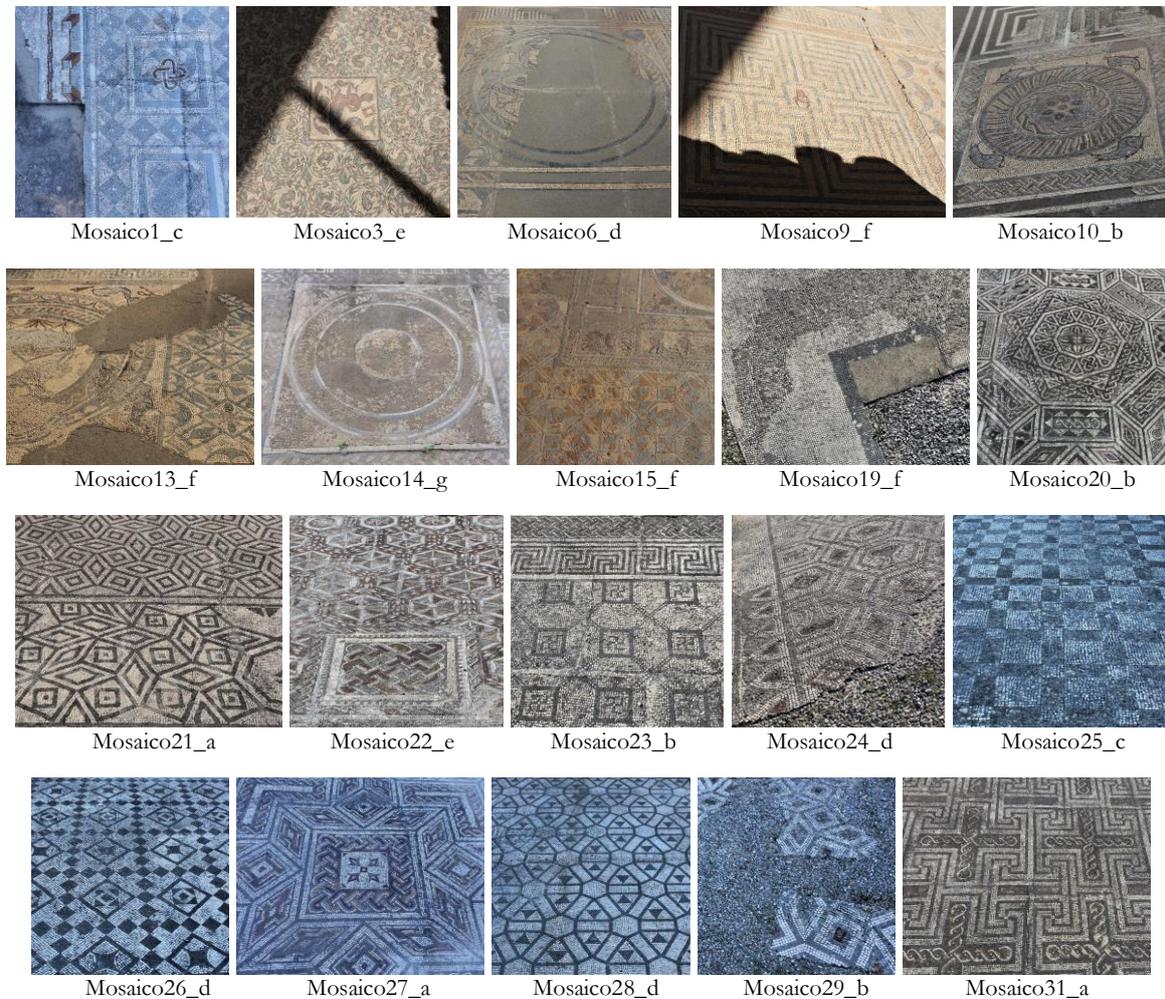


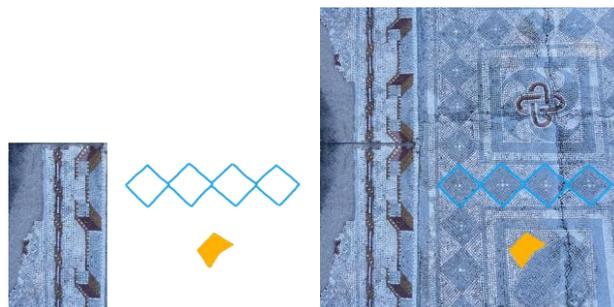
Figura 35: Targets escolhidos para a realização dos testes aos SDKs em Conímbriga

Mosaico	CraftAR	Wikitude
Mosaico 1_c	5/5	2/3
Mosaico 3_e	5/5	2/3
Mosaico 6_d	3/5	1/3
Mosaico 9_f	4/5	2/3
Mosaico 10_b	5/5	2/3
Mosaico 13_f	5/5	2/3
Mosaico 14_g	5/5	2/3
Mosaico 15_f	4/5	1/3
Mosaico 19_a	5/5	3/3
Mosaico 20_b	5/5	3/3
Mosaico 21_a	5/5	3/3
Mosaico 22_e	5/5	3/3
Mosaico 23_b	5/5	3/3
Mosaico 24_d	5/5	3/3
Mosaico 25_c	5/5	3/3
Mosaico 26_d	5/5	3/3
Mosaico 27_a	5/5	2/3
Mosaico 28_d	5/5	3/3
Mosaico 29_b	5/5	3/3
Mosaico 31_a	5/5	2/3

**Tabela 13:** Avaliações dos targets em ambas as interfaces

- **Overlays**

Os *overlays* utilizados (Fig. 36) tratam-se de imagens 2D, criadas no Photoshop, que contêm desenhos do contorno de certos elementos dos mosaicos, de forma a ser possível analisar o alinhamento destes *overlays* ao longo do rastreamento dos respetivos *targets*.



**Figura 36:** Exemplo de um overlay (imagem à esquerda) e do respetivo target com esse overlay sobreposto (imagem à direita)

- **Inclinação**

O smartphone pode ser manipulado de modo a que fique com diferentes inclinações, relativamente aos *targets*:

- Paralelo (*i.e.*, na posição inicial)
- Inclinado para cima/baixo
- Inclinado para o lado esquerdo/lado direito

- **Velocidade da Inclinação**

O smartphone pode ser inclinado com diferentes velocidades:

- Baixa (*i.e.*, lentamente)

- Normal (*i.e.*, normalmente)
- Alta (*i.e.*, rapidamente)

- **Distância**

O smartphone pode encontrar-se a diferentes distâncias, relativamente aos *targets* e a uma posição inicial:

- Baixa (*i.e.*, afastado da posição inicial, mais junto ao *target*)
- Normal (*i.e.*, na posição inicial)
- Alta (*i.e.*, afastado da posição inicial, mais afastado do *target*)

Quanto às variáveis **dependentes**, elas são as seguintes:

- **Atraso no reconhecimento do *target***

O atraso no reconhecimento dos *targets* é representado através do número de *frames* contabilizadas entre a posição inicial e o surgimento do *overlay* (número positivo). Caso o *overlay* apareça antes de se chegar à posição inicial, esta contagem será negativa.

- **Estabilidade**

Desempenho de rastreamento de um *target*, face ao *overlay* sobreposto no mesmo. É atribuído um valor (-1, 0, 1) dependendo do grau de estabilidade. O *overlay* pode apresentar-se como sendo:

- Nada estável (-1)
- Pouco estável (0)
- Estável (1)

- **Precisão do alinhamento**

Precisão com que um *overlay* surge em relação ao respetivo *target*. É atribuído um valor (-1, 0, 1) depende do grau de precisão. A *overlay* pode encontrar-se:

- Muito desalinhado (-1)
- Desalinhado (0)
- Alinhado (1)

- **Número de desaparecimentos do *overlay***

É contabilizado o número de vezes que um *overlay* desaparece, a partir do momento em que surge pela primeira vez.

- **Percentagem de área visível do *target***

Nas inclinações do *smartphone* para a esquerda e para a direita, é calculada a área do *target* que ainda é visível na *frame* seguinte àquela em que o *overlay* desaparece pela última vez. O valor, em percentagem, é relativo à área total do *target*.

## Descrição dos testes

Nestes testes do 2º semestre, a interface também não teria influência no desempenho dos SDKs nem relevância para as tarefas a serem executadas. Tendo isso em conta, a experiência de RA iniciava logo após a abertura da aplicação. Quanto às condições de luminosidade, estas encontravam-se semelhantes àquelas presenciadas na deslocação anterior a Conímbriga, pelo que

o céu se encontrava maioritariamente limpo. Dito isto, segue-se então a listagem dos testes que foram realizados em cada mosaico:

1. Começando com a câmara apontada para baixo (*i.e.*, para os pés do utilizador), incliná-lo para cima até atingir a posição estável (o momento em que o *target* aparece na íntegra no ecrã) e esperar até que o *overlay* apareça.
  - a. Caso apareça, inclinar o *smartphone* rapidamente para a esquerda até que o *target* desapareça do ecrã e, logo após, voltar à posição estável, de forma rápida também. Repetir o mesmo procedimento para a direita.
  - b. Caso não apareça, o *target* é considerado como indetetável pelo respetivo SDK.
2. Começando com a câmara na posição estável (*i.e.*, com o *target* visível na íntegra no ecrã) e com o *overlay* visível, iniciar uma inclinação lenta para a esquerda, até cerca de 30% do *target* ficar visível, e repetir o mesmo procedimento para a direita, para cima e para baixo, nesta ordem.
3. Começando com a câmara na posição estável (*i.e.*, com o *target* visível na íntegra no ecrã) e com o *overlay* visível, iniciar uma aproximação lenta na direção do *target*, até o braço ficar completamente estendido. De seguida, realizar o movimento oposto, até o *smartphone* ficar junto ao utilizador.

## Análise dos vídeos

Os resultados destes testes foram obtidos inteiramente a partir da análise dos respetivos vídeos. Foram, portanto, realizados três vídeos em cada mosaico romano, um para cada teste (descrito anteriormente). Segue-se a metodologia aplicada na análise de cada vídeo e na extração dos seus resultados:

- **Vídeo #1**

O primeiro vídeo, que diz respeito à primeira tarefa, foi analisado tendo em conta duas variáveis: o atraso decorrido durante a fase de reconhecimento do *target* e a percentagem de área visível do *target*. A primeira variável representa o tempo que o SDK demora a reconhecer o *target* e a mostrar o respetivo *overlay*. Este atraso é representado através do número de *frames*. Tendo isto em conta, a contagem é iniciada na *frame* correspondente à posição estável e é terminada na *frame* em que o *overlay* aparece pela primeira vez. Apesar desta contagem ser positiva, existiram casos em que o *overlay* surgia antes de se chegar à posição estável. Neste caso, a contagem teria que ser negativa (*i.e.*, as *frames* seriam contadas a partir do momento que o *overlay* surgisse até à posição estável).

Relativamente à segunda variável, esta representa a área do *target* (em percentagem) que ainda se encontra visível, no momento em que este deixa de ser detetado. Esta área é calculada tanto no movimento para o lado esquerdo como no movimento para o lado direito, sendo que é capturada a *frame* seguinte ao desaparecimento do *overlay*. Esta *frame* é então comparada com a imagem original do *target*, de forma a perceber a área que ainda ficou visível e traçar os contornos necessárias. Por fim, essa área seria calculada e convertida em percentagem, relativamente à área total do *target*.

- **Vídeo #2**

O segundo vídeo, referente à segunda tarefa, foi analisado tendo em conta três variáveis: precisão do alinhamento, estabilidade e número de vezes que o *overlay* desapareceu. Para todas estas variáveis, foi utilizada a mesma escala de valores (-1, 0, 1). A precisão de alinhamento diz respeito à análise subjetiva da qualidade do alinhamento do *overlay* em relação ao *target*, ao longo de todo o vídeo. Caso o *overlay* estivesse sempre perfeitamente alinhado, seria atribuído o valor 1 à sua precisão de alinhamento. Caso o *overlay* não estivesse perfeitamente alinhado, mas não estivesse demasiado desalinhado, seria atribuído o valor 0. Caso o *overlay* tivesse uma péssima precisão de alinhamento com o *target*, seria atribuído o valor -1. A estabilidade refere-se, também, a uma análise subjetiva, que se refere à forma como o *overlay* se encontra sobreposto no *target*, no que toca a movimentos. Caso o *overlay* aparenta estar “parado” no mesmo sítio durante todo o vídeo, este é considerado muito estável e é atribuído um valor 1 a esta avaliação. Caso o *overlay* esteja “parado” grande parte do vídeo, mas seja possível de observar umas movimentações mínimas, o *overlay* é considerado estável e é atribuído o valor 0. Caso o *overlay* se “mexa” significativamente durante o vídeo, é considerado muito instável e é atribuído o valor -1. Por fim, foi contabilizado o número de vezes que o *overlay* desaparece, a partir da *frame* em que o *overlay* surge pela primeira vez e terminando na *frame* em que o *overlay* desaparece pela última vez.

- **Vídeo #3**

O terceiro vídeo, representativo da terceira tarefa, foi analisado tendo em conta as mesmas três variáveis assumidas para o vídeo #2: precisão do alinhamento, estabilidade e número de vezes que o *overlay* desapareceu. É utilizado também a mesma escala de valores (-1,0,1).

## Resultados dos testes

A primeira avaliação feita a estes testes diz respeito à capacidade de deteção por parte dos três SDKs, para o mesmo mosaico romano. Através da Tabela 14, é possível observar a listagem dos mosaicos que foram testados e os resultados relativamente à capacidade de deteção dos três SDKs.

Mosaico	CraftAR	PixLive	Wikitude	
Mosaico 1_c	Detetado	Detetado	Não Detetado	
Mosaico 3_e	Não Detetado	Não Detetado	Não Detetado	
Mosaico 6_d	Detetado	Detetado	Não Detetado	
Mosaico 9_f	Não Detetado	Não Detetado	Não Detetado	
Mosaico 10_b	Detetado	Não Detetado	Não Detetado	
Mosaico 13_f	Detetado	Não Detetado	Não Detetado	
Mosaico 14_g	Detetado*	Detetado	Não Detetado	
Mosaico 15_f	Detetado	Não Detetado	Não Detetado	
Mosaico 19_a	Não Detetado	Não Detetado	Não Detetado	
Mosaico 20_b	Detetado	Não Detetado	Não Detetado	
Mosaico 21_a	Detetado**	Não Detetado	Não Detetado	
Mosaico 22_e	Detetado	Não Detetado	Não Detetado	
Mosaico 23_b	Não Detetado	Não Detetado	Não Detetado	
Mosaico 24_d	Não Detetado	Não Detetado	Não Detetado	
Mosaico 25_c	Detetado	Detetado	Não Detetado	
Mosaico 26_d	Detetado	Detetado	Detetado	
Mosaico 27_a	Detetado	Detetado	Não Detetado	
Mosaico 28_d	Detetado	Detetado	Detetado	
Mosaico 29_b	Detetado	Não Detetado	Não Detetado	
Mosaico 31_a	Detetado***	Detetado***	Não Detetado	
	<b>15</b>	<b>8</b>	<b>2</b>	<b>Total de Mosaicos detetados</b>

**Tabela 14:** Resultados da capacidade de deteção dos SDKs

\* Sem contar a análise feita ao segundo vídeo, devido a problemas ocorridos com o *ScreenRecorder*

\*\* Sem contar a análise feita ao terceiro vídeo, devido a problemas ocorridos com o *ScreenRecorder*

\*\*\* Sem recorrer a qualquer vídeo, devido a problemas ocorridos com o *ScreenRecorder*

Olhando para estes primeiros resultados, é evidente a superioridade do *CraftAR* na capacidade de deteção, em relação aos outros SDKs. O *CraftAR* foi capaz de detetar 15 mosaicos num total de 20 (75%), contra 8 (40%) reconhecidos pelo *PixLive*, e apenas 2 (10%) reconhecidos pela *Wikitude*. Tendo em conta os testes do primeiro semestre, não seriam esperados os resultados pobres mostrados pelo SDK da *Wikitude*, além de que grande parte dos *targets* apresentava uma avaliação positiva de qualidade de deteção, na interface da *Wikitude*. Com base apenas nestes resultados, o SDK da *Wikitude* já não seria tido em conta nas restantes análises realizadas e estaria, desde já, excluído como opção viável para o desenvolvimento da aplicação móvel. Sendo que foram reconhecidos os mesmos 8 mosaicos tanto pelo *CraftAR* como pelo *PixLive*, as restantes análises serão realizadas apenas considerando este conjunto. No entanto, como surgiram problemas com a ferramenta *ScreenRecorder* durante a gravação dos testes no mosaico “Mosaico 31\_a”, este não poderá ser considerado para as restantes análises, pelo que o número de mosaicos considerados passaria para 7.

Relativamente à análise do atraso de deteção dos *targets*, os resultados podem ser observados pela Tabela 15.

Mosaico	Número de Frames até o <i>overlay</i> aparecer		
	CraftAR	PixLive	
Mosaico 1_c	-8	48	
Mosaico 6_d	-6	6	
Mosaico 14_g	-2	16	
Mosaico 25_c	-2	6	
Mosaico 26_d	15	5	
Mosaico 27_a	-4	19	
Mosaico 28_d	-2	11	
	<b>-1.3</b>	<b>15.9</b>	<b>Média</b>

**Tabela 15:** Resultados do atraso da deteção dos targets

Observando estes resultados, o *CraftAR* mostra uma eficiência significativa na deteção dos mosaicos (*i.e.*, dos seus *targets*), apresentando uma média negativa na contagem do número de frames até o aparecimento dos *overlays*. Este valor indica que, em média, os *targets* dos mosaicos estudados são detetados pelo *CraftAR* praticamente no momento em que estes são capturados pela câmara. Já o *PixLive* apresenta um atraso maior na deteção dos *targets*, quer em média, quer em cada mosaico detetado.

A próxima análise consiste na percentagem de área visível dos *targets*, no momento em que os respetivos *overlays* desaparecem. Os resultados podem ser observados na Tabela 16, tanto para o movimento esquerdo como o movimento direito.

Mosaico	Movimento para a esquerda		Movimento para a direita		
	CraftAR	PixLive	CraftAR	PixLive	
Mosaico 1_c	37%	17%	75%	8%	
Mosaico 6_d	40%	51%	56%	8%	
Mosaico 14_g	35%	22%	30%	16%	
Mosaico 25_c	33%	26%	26%*	**	
Mosaico 26_d	41%	18%	49%	19%	
Mosaico 27_a	26%	14%	11%	21%	
Mosaico 28_d	50%	6%	43%	28%	
	<b>37.43 %</b>	<b>22%</b>	<b>44%</b>	<b>16.67%</b>	<b>Média</b>

**Tabela 16:** Resultados do atraso da deteção dos targets

\* Valor não contabilizado para a média, uma vez que este não se encontra definido no *PixLive*, para o mesmo mosaico

\*\* O *overlay* desaparece antes do início do movimento para a direita, não sendo possível medir a área nestas circunstâncias

Neste campo, o PixLive apresentou resultados mais satisfatórios em relação ao CraftAR, na medida em que a deteção dos *targets* persistiu durante mais tempo. No movimento para a esquerda, utilizando o PixLive, em média, restou apenas 22% de área visível dos *targets*, antes do *overlay* desaparecer. Já com o CraftAR, em média, restou 37.43% de área visível dos *targets*, indicando que estes deixam de ser detetados mais cedo. Embora no movimento para a esquerda a diferença não seja significativa, para o lado direito já é possível de verificar uma diferença mais acentuada, verificando-se, em média, 27% a mais de área visível dos *targets*, utilizando o CraftAR. Por fim, resta mostrar os resultados da precisão do alinhamento dos *overlays*, da sua estabilidade e da quantidade de vezes que estes desapareceram. Para tal, foram criadas as Tabelas 17 e 18, onde estão expostos os resultados desta análise extraídos do vídeo #2 e do vídeo #3, respetivamente.

Mosaico	Precisão do alinhamento (-1, 0, 1)		Estabilidade (-1, 0, 1)		Quantidade de vezes que o <i>overlay</i> desaparece		Média
	CraftAR	PixLive	CraftAR	PixLive	CraftAR	PixLive	
Mosaico 1_c	0	1	0	0	4	0	
Mosaico 6_d	-1	0	-1	0	7	0	
Mosaico 14_g	*	0**	*	-1**	*	0**	
Mosaico 25_c	-1	0	0	-1	0	0	
Mosaico 26_d	0	1	0	0	1	0	
Mosaico 27_a	1	1	1	-1	1	0	
Mosaico 28_d	-1	0	0	0	0	0	
	<b>-0.33</b>	<b>0.5</b>	<b>0</b>	<b>-0.5</b>	<b>2.17</b>	<b>0</b>	

**Tabela 17:** Resultados do atraso da deteção dos targets

\* Valor impossível de atribuir, devido a problemas com os vídeos

\*\* Valor não contabilizado para a média, uma vez que este não se encontra atribuído no *CraftAR*, para o mesmo mosaico

Mosaico	Precisão do alinhamento (-1, 0, 1)		Estabilidade (-1, 0, 1)		Quantidade de vezes que o <i>overlay</i> desaparece		Média
	CraftAR	PixLive	CraftAR	PixLive	CraftAR	PixLive	
Mosaico 1_c	0	1	1	0	0	0	
Mosaico 6_d	-1	1	1	-1	0	0	
Mosaico 14_g	-1	1	1	-1	0	0	
Mosaico 25_c	-1	-1	1	-1	0	0	
Mosaico 26_d	0	1	0	0	0	0	
Mosaico 27_a	0	1	1	-1	0	0	
Mosaico 28_d	-1	0	0	0	0	0	
	<b>-0.43</b>	<b>0.57</b>	<b>0.71</b>	<b>-0.57</b>	<b>0</b>	<b>0</b>	

**Tabela 18:** Resultados do atraso da deteção dos targets

Em primeiro lugar, é possível observar que a precisão do alinhamento dos *overlays* em relação aos respetivos *targets*, utilizando o PixLive, é mais satisfatória do que utilizando o CraftAR. Este último SDK apresenta, inclusive, médias negativas (-0.33 e -0.43) em ambos os testes, enquanto que o PixLive apresentou médias (0.5 e 0.57) próximas do valor ideal (*i.e.*, 1). De seguida, relativamente à estabilidade, a situação inverte-se. O CraftAR apresentou, em média, uma estabilidade satisfatória em ambos os testes, sendo que no terceiro teste chega praticamente a alcançar o valor ideal em todos os mosaicos, com uma média (0.71) próxima do valor ideal (*i.e.*, 1). Enquanto isso, o PixLive apresentou médias negativas de estabilidade (-0.5 e -0.57), em ambos os testes, mostrando apresentar bastante instabilidade no rastreamento dos *targets*. Por fim, quanto à quantidade de vezes que os *targets* deixam de ser detetados (*i.e.*, desaparecem os *overlays*), os resultados foram bastante satisfatórios, em ambos os SDKs. À exceção do segundo teste, onde, em média, os *targets* deixaram de ser detetados aproximadamente 2 vezes utilizando o CraftAR, os *targets* seriam detetados do início ao fim, não havendo qualquer desaparecimento aparente dos *overlays*.

Depois de todos estes resultados e respetivas análises, é possível retirar as seguintes conclusões:

- Tanto o CraftAR como o PixLive são opções seguras para serem utilizados nos mosaicos romanos de Conímbriga, sendo que detetaram uma quantidade satisfatória. O CraftAR (15) no entanto, detetou quase o dobro dos mosaicos detetados pelo PixLive (8).
- O CraftAR releva uma maior eficiência na tecnologia de deteção dos *targets*, sendo que estes foram, em média, detetados de forma rápida.
- O PixLive apresentou melhores resultados a nível do rastreamento dos *targets*, na medida em que os *overlays* eram observáveis ainda no momento em que os *targets* praticamente saíram do campo de visão da câmara.
- O PixLive apresentou resultados bastante satisfatórios a nível da precisão do alinhamento dos *overlays*, e melhores que aqueles apresentados pelo CraftAR. No entanto, o PixLive apresenta bastante instabilidade, indicando que, sempre que o *overlay* estabiliza, volta à posição e alinhamento corretos. Por outro lado, o CraftAR apresenta uma estabilidade quase ótima em todos os mosaicos que foi testado.

Terminados estes testes, e estudados os resultados, a escolha do SDK para ser utilizado no desenvolvimento do protótipo final acabou por recair para o CraftAR. As razões principais que levaram a esta decisão basearam-se na quantidade considerável de mosaicos detetados (que excedeu as expectativas) e na rapidez e eficácia com os mosaicos (*i.e.*, *targets*) foram detetados. Ao conseguir detetar bastantes mosaicos, este SDK oferece uma boa margem de manobra para aplicar a experiência de RA. Além disso, ao ser capaz de realizar as deteções rapidamente, tornará a experiência menos frustrante para os visitantes de Conímbriga. Já o PixLive, mediante estes testes, também se apresentou como uma boa alternativa ao CraftAR, ao contrário do Wikitude que, por razões que deverão ser investigadas futuramente, fiquem aquém das expectativas.

## 5. ConimbrigAR

O protótipo proposto a ser desenvolvido no âmbito desta dissertação foi apelidado de **ConimbrigAR**. O seu planeamento foi iniciado no segundo semestre, onde foram definidos os requisitos, a arquitetura do sistema, o modelo de dados, e as ferramentas e tecnologias que viriam a ser utilizadas. Além da aplicação móvel, também foi proposta a criação de uma interface web que permitisse a gestão dos conteúdos da primeira. Dito isto, neste capítulo será exposto o todo o processo de trabalho levado a cabo ao longo do segundo semestre, desde a definição dos requisitos até ao protótipo final.

### 5.1. Requisitos

Nesta seção são apresentados os atores e as *User Stories* definidos para o sistema, os requisitos funcionais delineados, juntamente com os respetivos casos de uso e, por fim, os requisitos de qualidade (não funcionais).

#### 5.1.1. Atores e User Stories

Neste sistema, existem essencialmente dois atores:

- **Visitante das ruínas de Conímbriga**  
Ator responsável pela utilização da aplicação móvel.
- **Gestor de conteúdos**  
Ator responsável pela gestão dos dados da aplicação móvel, através da interface web.

As diversas ações que os atores podem realizar neste sistema são descritas através de *User Stories*. Desta forma, será possível idealizar os requisitos que serão descritos mais à frente neste documento, do ponto de vista dos atores do sistema. Será apresentada então, de seguida, uma listagem com as *User Stories* (US). Estas irão seguir o *template* seguinte:

- Enquanto <ator>
- Posso <ação>
- De forma a <proposta de valor>

Definido o *template*, seguem-se então as *User Stories*:

- **US #1** – Enquanto visitante das ruínas de Conímbriga, posso descarregar a aplicação móvel, de forma a poder utilizá-la nos mosaicos romanos das ruínas de Conímbriga.
- **US #2** – Enquanto visitante das ruínas de Conímbriga, posso utilizar a aplicação móvel sem necessitar de autenticação, de forma a ter acesso aos seus conteúdos.
- **US #3** – Enquanto visitante das ruínas de Conímbriga, posso utilizar a aplicação móvel sem necessitar de ligação à Internet, de forma a poder utilizá-la em modo *offline*.
- **US #4** – Enquanto visitante das ruínas de Conímbriga, posso aceder a uma listagem dos mosaicos romanos presentes nas ruínas de Conímbriga de forma a selecionar um deles.

- **US #5** – Enquanto visitante das ruínas de Conímbriga, posso escolher um mosaico da listagem, de forma a observar uma imagem sua e informações sobre o mesmo.
- **US #6** – Enquanto visitante das ruínas de Conímbriga, posso aceder a um grupo a partir de mosaico, de forma a observar uma imagem sua e informações sobre o mesmo.
- **US #7** – Enquanto visitante das ruínas de Conímbriga, posso aceder a uma listagem dos grupos de mosaicos romanos de forma a selecionar um deles.
- **US #8** – Enquanto visitante das ruínas de Conímbriga, posso escolher um grupo da listagem, de forma a observar uma imagem sua e informações sobre o mesmo.
- **US #9** – Enquanto visitante das ruínas de Conímbriga, posso aceder a um mosaico a partir de um grupo, de forma a observar uma imagem sua e informações sobre o mesmo.
- **US #10** – Enquanto visitante das ruínas de Conímbriga, posso aceder a uma listagem das camadas de informação de forma a selecionar uma delas.
- **US #11** – Enquanto visitante das ruínas de Conímbriga, posso escolher uma camada da listagem, de forma a observar uma imagem sua e informações sobre a mesma.
- **US #12** – Enquanto visitante das ruínas de Conímbriga, posso selecionar a opção de início da experiência de RA, de forma a que esta inicie.
- **US #13** - Enquanto visitante das ruínas de Conímbriga, posso observar os Pontos de Interesse, de forma a saber onde se encontram os mosaicos.
- **US #14** - Enquanto visitante das ruínas de Conímbriga, posso distinguir os Pontos de Interesse, de forma a saber quais os respetivos mosaicos que contêm experiências de RA.
- **US #15** - Enquanto visitante das ruínas de Conímbriga, posso selecionar um Ponto de Interesse, de forma a obter informações sobre o respetivo mosaico.
- **US #16** - Enquanto visitante das ruínas de Conímbriga, posso apontar a câmara do *smartphone* para um mosaico romano, de forma a iniciar uma experiência de RA.
- **US #17** - Enquanto visitante das ruínas de Conímbriga, posso efetuar movimentos subtis com o *smartphone*, de forma a observar os *overlays* de perspetivas diferentes.
- **US #18** - Enquanto visitante das ruínas de Conímbriga, posso alternar entre os *overlays* do mosaico atual, de forma a observar os seus diferentes *overlays* em tempo real.
- **US #19** - Enquanto visitante das ruínas de Conímbriga, posso alternar entre camadas de informação, de forma a modificar o conjunto de *overlays* atual.
- **US #20** – Enquanto gestor de conteúdos, posso aceder à interface web, de forma a proceder ao *login*.
- **US #21** – Enquanto gestor de conteúdos, posso inserir o email e password da minha conta, de forma a autenticar-me na interface web.
- **US #22** – Enquanto gestor de conteúdos, posso aceder à listagem dos mosaicos, de forma a observá-los todos e as suas informações.
- **US #23** – Enquanto gestor de conteúdos, posso aceder à listagem dos grupos, de forma a observá-los todos e as suas informações.
- **US #24** – Enquanto gestor de conteúdos, posso aceder à listagem de camadas, de forma a observá-las todas e as suas informações.
- **US #25** – Enquanto gestor de conteúdos, posso aceder à listagem de *overlays*, de forma a observá-los todos e as suas informações.

- **US #26** – Enquanto gestor de conteúdos, posso aceder à listagem de *targets*, de forma a observá-los todos e as suas informações.
- **US #27** – Enquanto gestor de conteúdos, posso editar um mosaico, de forma a alterar as suas informações ou eliminá-lo do sistema.
- **US #28** – Enquanto gestor de conteúdos, posso editar um grupo, de forma a alterar as suas informações ou eliminá-lo do sistema.
- **US #29** – Enquanto gestor de conteúdos, posso editar uma camada, de forma a alterar as suas informações ou eliminá-la do sistema.
- **US #30** – Enquanto gestor de conteúdos, posso editar um *overlay*, de forma a alterar as suas informações ou eliminá-lo do sistema.
- **US #31** – Enquanto gestor de conteúdos, posso editar um *target*, de forma a alterar as suas informações ou eliminá-lo do sistema.
- **US #32** – Enquanto gestor de conteúdos, posso criar um novo mosaico, de forma a que seja adicionado ao leque de mosaicos existente.
- **US #33** – Enquanto gestor de conteúdos, posso criar um novo grupo, de forma a que seja adicionado ao leque de grupos existente.
- **US #34** – Enquanto gestor de conteúdos, posso criar uma nova camada, de forma a que seja adicionado ao leque de camadas existente.
- **US #35** – Enquanto gestor de conteúdos, posso criar um novo *overlay*, de forma a que seja adicionado ao leque de *overlays* existente.
- **US #36** – Enquanto gestor de conteúdos, posso criar um novo *target*, de forma a que seja adicionado ao leque de *targets* existente.
- **US #37** – Enquanto gestor de conteúdos, posso criar um novo *target*, de forma a que seja adicionado ao leque de *targets* existente.
- **US #38** – Enquanto gestor de conteúdos, posso fazer proceder ao *logout*, de forma a terminar a sessão na interface web.

### 5.1.2. Requisitos Funcionais e Não Funcionais

O levantamento dos requisitos funcionais e não funcionais foi realizado no primeiro semestre e reavaliado no segundo semestre, altura em que o desenvolvimento do projeto iniciava. Os requisitos funcionais foram definidos e incorporados com os respetivos casos de uso, de modo a que fiquem explícitos e mais completos. Segue-se, primeiramente, o *template* criado para a definição dos **requisitos funcionais** (RF):

Componente	Notas
ID	Identificador do requisito, no seguinte formato: RF-<# do Requisito>: <Nome>
Atores	Atores intervenientes
Descrição	Descrição geral do requisito funcional
Prioridade	O nível de prioridade pode ter uma das seguintes classificações: <ul style="list-style-type: none"> <li>• <b>Alta</b></li> <li>• <b>Média</b></li> <li>• <b>Baixa</b></li> </ul>

Pré-Condições	Condições essenciais para o cumprimento do requisito funcional
Fluxo de trabalho	Listagem de ações realizadas pelo respetivo ator, de modo a cumprir o requisito funcional

**Tabela 19:** Template para os requisitos funcionais.

A listagem completa dos requisitos funcionais pode ser consultada no Anexo B1. No entanto, para fins de demonstração, seguem-se alguns exemplos dos requisitos funcionais mais relevantes, dessa listagem:

<b>ID</b>	RF-1.1.1: Acesso às informações de um mosaico
<b>Atores</b>	Visitante das ruínas de Conímbriga
<b>Descrição</b>	O visitante pode aceder às informações sobre um mosaico, incluindo uma imagem, os seus detalhes e os seus grupos
<b>Prioridade</b>	Alta
<b>Pré-Condições</b>	O visitante já se deve encontrar com a aplicação móvel aberta e sem ter a experiência de RA ativa.
<b>Fluxo de trabalho</b>	<ul style="list-style-type: none"> <li>• O visitante acede à <i>tab</i> da listagem dos dados</li> <li>• O visitante escolhe a opção referente à listagem de mosaicos.</li> <li>• O visitante seleciona um mosaico.</li> </ul>

<b>ID</b>	RF-3.2: Visualização dos Pontos de Interesse distintos
<b>Atores</b>	Visitante das ruínas de Conímbriga
<b>Descrição</b>	O visitante pode visualizar os Pontos de Interesse que sejam distintos dos restantes, de forma a perceber que os mosaicos correspondentes contêm experiências de RA.
<b>Prioridade</b>	Média
<b>Pré-Condições</b>	O visitante já se deve encontrar com a aplicação móvel aberta e ter a experiência de RA ativa.
<b>Fluxo de trabalho</b>	<ul style="list-style-type: none"> <li>• O visitante movimenta o seu <i>smartphone</i>, até que os Pontos de Interesse distintos apareçam no campo de visão da câmara.</li> </ul>

<b>ID</b>	RF-3.4: Início da experiência de RA de um mosaico romano
<b>Atores</b>	Visitante das ruínas de Conímbriga
<b>Descrição</b>	O visitante pode iniciar uma experiência de RA de um mosaico romano.
<b>Prioridade</b>	Alta
<b>Pré-Condições</b>	O visitante já se deve encontrar com a aplicação móvel aberta e ter a experiência de RA ativa.
<b>Fluxo de trabalho</b>	<ul style="list-style-type: none"> <li>• O visitante aponta a câmara do <i>smartphone</i> para um mosaico que contenha uma experiência de RA. Caso a experiência não seja iniciada de imediato, o visitante</li> </ul>

	terá que movimentar subtilmente o <i>smartphone</i> , sem que o mosaico saia da mira da câmara.
--	---

<b>ID</b>	RF-3.4.1: Alternar entre <i>overlays</i> de um mosaico
<b>Atores</b>	Visitante das ruínas de Conímbriga
<b>Descrição</b>	O visitante pode alternar entre os <i>overlays</i> existentes no mosaico atual, em tempo real.
<b>Prioridade</b>	Alta
<b>Pré-Condições</b>	O visitante já se deve encontrar com a aplicação móvel aberta e ter a experiência de RA ativa. Além disso, é necessário que a experiência de RA do mosaico atual esteja ativa e que a camada de informação ativa/selecionada contenha <i>overlays</i> em comum com o mosaico atual.
<b>Fluxo de trabalho</b>	<ul style="list-style-type: none"> <li>• O visitante, com a câmara apontada para o mosaico e com a sua experiência de RA ativa, clica nas setas (esquerda ou direita) encontradas a meio do ecrã, para alternar entre os <i>overlays</i> do mosaico.</li> </ul>

<b>ID</b>	RF-6.10: Editar um <i>target</i>
<b>Atores</b>	Gestor de conteúdos
<b>Descrição</b>	O gestor de conteúdos pode editar os dados de um <i>target</i> específico ou eliminá-lo.
<b>Prioridade</b>	Alta
<b>Pré-Condições</b>	Ligação à Internet e <i>login</i> efetuado.
<b>Fluxo de trabalho</b>	<ul style="list-style-type: none"> <li>• O gestor de conteúdos acede à interface web</li> <li>• O gestor de conteúdos acede ao menu principal</li> <li>• O gestor de conteúdos clica no botão referente às operações sobre <i>targets</i></li> <li>• O gestor de conteúdos seleciona a opção de editar <i>targets</i></li> <li>• O gestor de conteúdos insere o ID do <i>target</i> que pretende editar e submete o pedido</li> <li>• O gestor de conteúdos edita os dados que pretender do <i>target</i> selecionado ou elimina-o.</li> </ul>

<b>ID</b>	RF-6.12: Criar um <i>target</i>
<b>Atores</b>	Gestor de conteúdos
<b>Descrição</b>	O gestor de conteúdos pode criar um novo <i>target</i> .
<b>Prioridade</b>	Alta
<b>Pré-Condições</b>	Ligação à Internet e <i>login</i> efetuado.
<b>Fluxo de trabalho</b>	<ul style="list-style-type: none"> <li>• O gestor de conteúdos acede à interface web</li> <li>• O gestor de conteúdos acede ao menu principal</li> <li>• O gestor de conteúdos clica no botão referente às operações sobre <i>targets</i></li> <li>• O gestor de conteúdos seleciona a opção de criar <i>targets</i></li> <li>• O gestor de conteúdos preenche o formulário e submete o pedido de criação do <i>target</i></li> </ul>

De seguida, relativamente aos **requisitos não funcionais** (RNF), foi definido o seguinte *template*:

Componente	Notas
ID	Identificador do requisito, no seguinte formato: R<NF>-<# do Requisito>: <Nome>
Descrição	Descrição geral do requisito não funcional
Prioridade	O nível de prioridade pode ter uma das seguintes classificações: <ul style="list-style-type: none"> <li>• <b>Alta</b></li> <li>• <b>Média</b></li> <li>• <b>Baixa</b></li> </ul>

**Tabela 20:** Template para os requisitos não funcionais.

Segue-se então a listagem dos requisitos não funcionais que foram delineados:

<b>ID</b>	RNF-1: Compatibilidade da aplicação móvel
<b>Descrição</b>	A aplicação deverá ser compatível com os sistemas operativos Android e iOS.
<b>Prioridade</b>	Alta

<b>ID</b>	RNF-2: Funcionamento offline
<b>Descrição</b>	A aplicação deverá funcionar corretamente e na íntegra, a partir dos dados locais (transferidos previamente).
<b>Prioridade</b>	Alta

<b>ID</b>	RNF-3: Responsividade da interface da aplicação móvel
<b>Descrição</b>	A interface da aplicação móvel deve adaptar-se ao dispositivo onde está a correr.
<b>Prioridade</b>	Alta

<b>ID</b>	RNF-4: Segurança dos dados
<b>Descrição</b>	Os dados da aplicação móvel/interface web devem estar devidamente protegidos e apenas serão acessíveis pelo gestor de conteúdos.
<b>Prioridade</b>	Alta

<b>ID</b>	RNF-5: Facilidade de utilização
<b>Descrição</b>	A aplicação móvel deve apresentar uma interface intuitiva e fácil de utilizar, de forma a que os utilizadores executem as tarefas dentro de um tempo satisfatório.
<b>Prioridade</b>	Média

Estes requisitos não funcionais conferem algumas qualidades como a portabilidade (RNF-1), disponibilidade (RNF-2), compatibilidade (RNF-3), segurança (RNF-4) e usabilidade (RNF-5).

## 5.2. Arquitetura do Sistema

Nesta seção é apresentada e descrita a arquitetura do sistema (Fig. 37), incluindo os seus componentes e o fluxo de trabalho entre os mesmos. Para o sistema em questão, foram separadas as responsabilidades, de modo a que houvesse uma parte do sistema que tratasse de aceder, processar, armazenar e controlar os dados requisitados pelo cliente (*Backend*) e outra parte que fosse responsável pela camada de apresentação dos dados ao cliente (*Frontend*). Deste modo, haverá mais flexibilidade e facilidade na manutenção do projeto, permitindo o foco em componentes específicos do sistema.

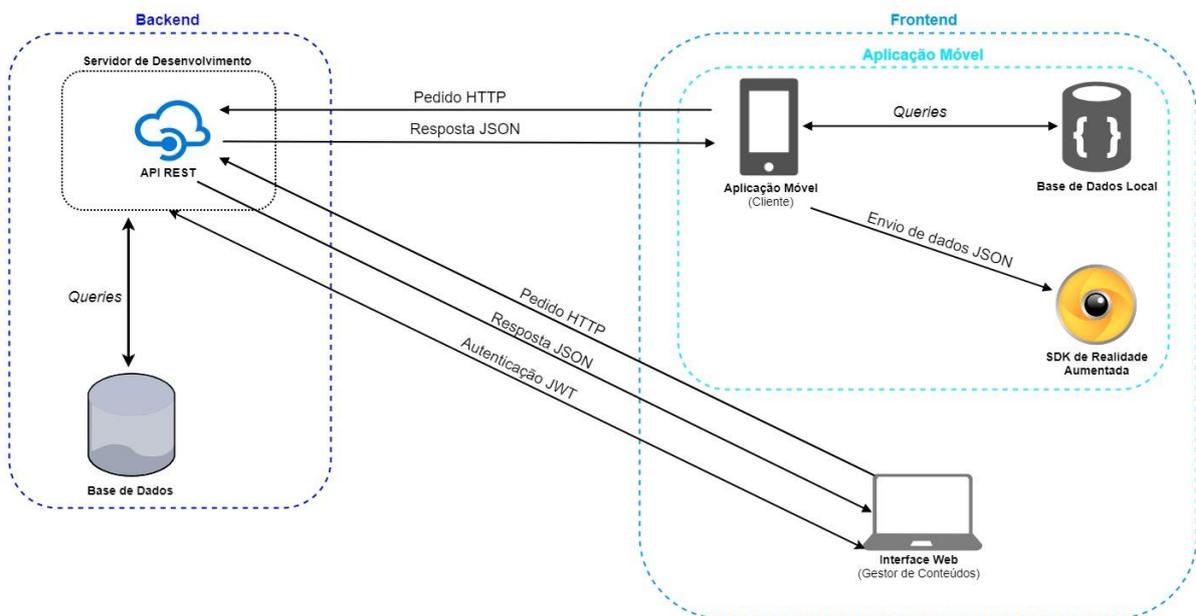


Figura 37: Arquitetura do sistema

No *Frontend*, serão encontrados os clientes da aplicação móvel, assim como o gestor de conteúdos da interface web. De modo a terem acesso aos dados, tanto a aplicação móvel como a interface terão que consumir o serviço API REST, presente no *Backend*, através de pedidos HTTP. Após a receção e interpretação dos pedidos, a API REST irá comunicar com a base dados e, posteriormente, enviar a resposta ao pedido HTTP, em formato JSON. Após a receção de dados provenientes da API, a aplicação móvel irá armazená-los numa base de dados local, de modo a que o cliente possa utilizá-la de modo offline. Estes dados locais serão ainda enviados para a componente de Realidade Aumentada, uma vez que esta é executada num contexto à parte. Por fim, para o gestor de conteúdos se autenticar na interface web, deverá ter as suas credenciais validadas no *Backend* e receber um JWT de volta, de forma a que possa aceder e utilizar a API sem restrições.

## 5.3. Modelo de Dados

O foco da aplicação móvel está na implementação da tecnologia de Realidade Aumentada, sobre os **mosaicos** romanos de Conímbriga. Para esse efeito, é realizado um registo destes mosaicos e, posteriormente, é feita a seleção de uma ou mais partes específicas dos mesmos,

denominadas por **targets**. Estas partes dos mosaicos são alvos de detecção por parte da tecnologia de RA e de sobreposição de informação virtual, representada por imagens 2D e denominada por **overlays**. Cada *target* terá o seu conjunto único de *overlays*. Os *overlays* podem ainda ser **transformados** de forma a que sejam colocados em lugares estratégicos do *target*, e também com diferentes escalas e orientações. Por sua vez, estes *overlays* estão distribuídos e organizados em conjuntos particulares, denominados por **camadas** de informação. Por fim, para efeitos de organização, os mosaicos podem ser distribuídos em diversos **grupos**. É possível então observar estas entidades, os seus campos e as suas ligações através do modelo de dado representado pelo diagrama de entidades e relacionamentos (Fig. 38).

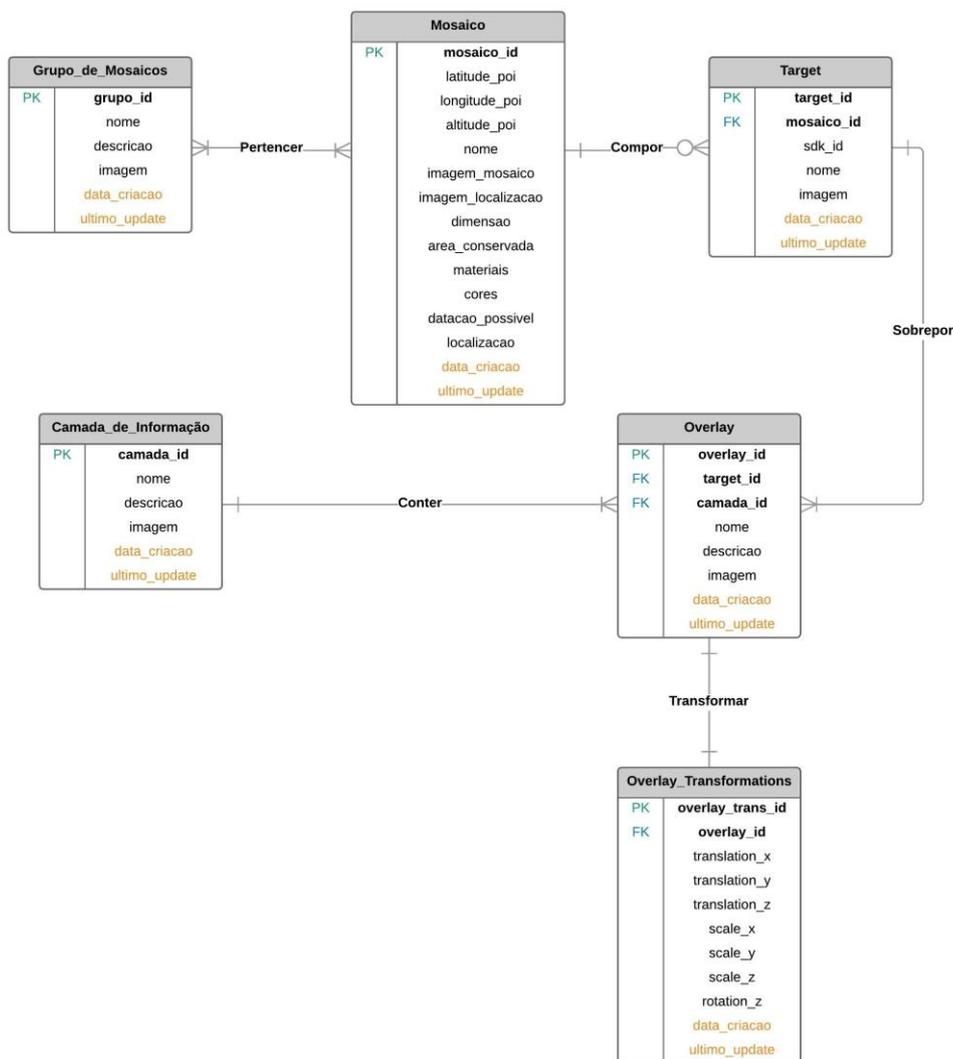


Figura 38: Diagrama de entidade e relacionamentos

A tabela *Mosaico* refere-se aos mosaicos romanos de Conímbriga e contém algumas informações sobre os mesmos, nomeadamente o nome, as dimensões, a área conservada, os materiais, as cores, a datação possível e o nome da sua localização. Além disso, esta tabela contém as coordenadas dos mosaicos, nomeadamente latitude, longitude e altitude, de forma a que seja possível saber a sua localização através dos Pontos de Interesse. Por último, esta tabela contém os caminhos das imagens armazenadas no servidor, que dizem respeito a si (*imagem\_mosaico*) e à zona onde estão localizados (*imagem\_localizacao*). Pode ser observado um exemplo de um objeto do tipo *Mosaico* na Figura 39. Este exemplo contém dois campos adicionais, *targets\_deste\_mosaico* e

*grupos*, de forma a ser possível saber quais os *targets* e os grupos pertencentes a um mosaico, através dos respetivos IDs. Estes campos são gerados pelas relações *Many-to-Many* (*Mosaico* e *Grupo\_de\_Mosaicos*) e *One-to-Many* (*Mosaico* e *Target*).

```
{
  "id": 1,
  "targets_deste_mosaico": [
    1,
    2,
    3,
    4
  ],
  "nome": "Mosaico geométrico com meandro em ressaltos e quadrados figurados",
  "dimensao": "3,41 m x 0,63 m (fragmento); 4,59 m x 2,88 m (fragmento)",
  "area_conservada": "Duas áreas distintas, estando um grande fragmento fora de contexto",
  "materiais": "Calcário",
  "cores": "Branco, preto, vermelho",
  "datacao_possivel": "Finais do Séc.II/Séc. III",
  "localizacao": "Casa de Cantaber",
  "imagem_mosaico": "http://localhost:8000/imagens_conimbrigar/mosaicos/mosaico1.png",
  "imagem_localizacao": "http://localhost:8000/imagens_conimbrigar/mosaicos-local/mosaico1-local.png",
  "latitude_poi": "40.09924400",
  "longitude_poi": "-8.49254500",
  "altitude_poi": 100,
  "data_criacao": "2018-06-13T19:49:24.004442+01:00",
  "ultimo_update": "2018-06-13T19:49:24.004442+01:00",
  "grupos": [
    1
  ]
},
```

**Figura 39:** Exemplo de um objeto do tipo *Mosaico*

A tabela *Grupos\_de\_Mosaicos* contém o nome dos grupos de mosaicos, assim como uma descrição e o caminho para a imagem que os representa (*imagem*). Pode ser observado um exemplo de um objeto do tipo *Grupo\_de\_Mosaicos* na Figura 40. Este exemplo contém um campo adicional, *mosaicos\_deste\_grupo*, de forma a ser possível saber quais os mosaicos pertencentes a um grupo, através dos respetivos IDs. Este campo é gerado pela relação *Many-to-Many* (*Mosaico* e *Grupo\_de\_Mosaicos*).

```
{
  "id": 1,
  "mosaicos_deste_grupo": [
    1
  ],
  "nome": "Casa de Cantaber",
  "descricao": "Grupo de mosaicos localizados na Casa de Cantaber",
  "imagem": "http://localhost:8000/imagens_conimbrigar/grupos/Casa_de_Cantaber_pDQX0bh.png",
  "data_criacao": "2018-06-13T19:47:59.769323+01:00",
  "ultimo_update": "2018-06-13T19:47:59.769323+01:00"
},
```

**Figura 40:** Exemplo de um objeto do tipo *Grupo\_de\_Mosaicos*

A tabela *Target* refere-se aos *targets* de cada mosaico romano, pelo que estes serão únicos para o seu respetivo mosaico. Por outro lado, um mosaico pode não conter qualquer *target* (não é considerado para a experiência de RA), conter apenas um ou conter vários *targets*. A tabela dos *targets* contém o nome dos mesmos, o caminho para a sua imagem (*imagem*) e um identificador que se encontra na interface do SDK de RA (*sdk\_id*). Pode ser observado um exemplo de um objeto do tipo *Target* na Figura 41. Este exemplo contém dois campos adicionais, *overlays\_deste\_target* e *mosaico*, de forma a ser possível saber quais os *overlays* e o mosaico pertencente a um *target*, através dos respetivos IDs. Estes campos são gerados pela relação *One-to-Many* (*Target* e *Overlay*; *Mosaico* e *Target*).

```

{
  "id": 1,
  "overlays_destes_target": [
    1
  ],
  "sdk_id": "5b21684093afd02f7e9b9a90",
  "nome": "target1_1",
  "imagem": "http://localhost:8000/imagens_conimbrigar/targets/target1_1_NrqEKh5.jpg",
  "data_criacao": "2018-06-13T20:00:21.211155+01:00",
  "ultimo_update": "2018-06-13T20:00:21.211155+01:00",
  "mosaico": 1
},

```

**Figura 41:** Exemplo de um objeto do tipo *Target*

A tabela *Overlays* contém o nome dos *overlays* existentes na aplicação, assim como uma descrição e o caminho para a sua imagem (*imagem*). Pode ser observado um exemplo de um objeto do tipo *Overlay* na Figura 42. Este exemplo contém dois campos adicionais, *target* e *camada*, de forma a ser possível saber qual o *target* e a camada pertencente a um *overlay*, através dos respetivos IDs. Este campo é gerado pela relação *One-to-Many* (*Target* e *Overlay*; *Target* e *Camada\_de\_Informacao*), uma vez que um *overlay* pertence apenas a um *target* e apenas a uma camada.

```

{
  "id": 6,
  "nome": "Logótipo dos 50 anos de Conimbriga",
  "descricao": "Texto \"50 Anos Museu Monográfico de Conimbriga\"",
  "imagem": "http://localhost:8000/imagens_conimbrigar/overlays/Logo-dos-50-anos-de-Conimbriga_yM19qJk.png",
  "data_criacao": "2018-06-14T00:55:05.853582+01:00",
  "ultimo_update": "2018-06-30T13:42:16.496987+01:00",
  "camada": 3,
  "target": 5
}

```

**Figura 42:** Exemplo de um objeto do tipo *Overlay*

A tabela *Camada\_de\_Informacao* contém o nome das camadas existentes na aplicação, uma descrição e o caminho da imagem alusiva (*imagem*). Uma camada de informação representa um conjunto de *overlays* com características particulares. Pode ser observado um exemplo de um objeto do tipo *Camada\_de\_Informacao* na Figura 43. Este exemplo contém um campo adicional, *overlays\_destes\_camada*, de forma a ser possível saber quais os *overlays* pertencentes a uma camada, através dos respetivos IDs. Este campo é gerado pela relação *One-to-Many* (*Camada* e *Overlay*).

```

{
  "id": 2,
  "overlays_destes_camada": [
    20,
    24,
    26,
    28,
    29,
    33
  ],
  "nome": "Restauro",
  "descricao": "Imagens colocadas em zonas danificadas do mosaico",
  "imagem": "http://localhost:8000/imagens_conimbrigar/camadas/camada1_Dn9X7NN.jpg",
  "data_criacao": "2018-06-14T00:26:46.618525+01:00",
  "ultimo_update": "2018-06-14T00:26:46.619526+01:00"
},

```

**Figura 43:** Exemplo de um objeto do tipo *Camada\_de\_Informacao*

A tabela *Overlay\_Transformations* refere-se às transformações que serão aplicadas sobre um determinado *overlay*, nomeadamente a nível da translação e da escala (nos eixos x, y e z), e de rotação, em z. Pode ser observado um exemplo de um objeto do tipo *Overlay\_Transformations* na Figura 44. Este exemplo contém um campo adicional, *overlay*, de forma a ser possível saber a que *overlay* as transformações são aplicadas, através do respetivo ID. Este campo é gerado pela relação *One-to-One* (*Overlay* e *Overlay\_Transformations*).

```
{
  "id": 1,
  "translation_x": "0.55",
  "translation_y": "0.20",
  "translation_z": "0.00",
  "scale_x": "1.00",
  "scale_y": "1.00",
  "scale_z": "1.00",
  "rotation_z": 45,
  "data_criacao": "2018-06-30T14:00:18.500532+01:00",
  "ultimo_update": "2018-06-30T14:00:18.500532+01:00",
  "overlay": 26
}
```

**Figura 44:** Exemplo de um objeto do tipo *Overlay\_Transformations*

Por fim, é importante mencionar que todas as tabelas contêm dois campos que dizem respeito à data em que foram criadas (*data\_criacao*) e à data da última alteração (*ultimo\_update*). Como será explicado mais à frente neste documento, estes campos são essenciais para a atualização dos dados da aplicação.

## 5.4. Descrição do Trabalho Desenvolvido

### 5.4.1. *Backend*

O primeiro passo no desenvolvimento do projeto foi dado precisamente no *Backend*. Seria necessário configurar o servidor de desenvolvimento do Django, começando pela integração da base de dados. Estando definidos os modelos de dados, foi criada uma base de dados local MySQL e conectada ao servidor. De seguida, os modelos de dados foram reproduzidos no Django, que contém esta característica, e migrados para a base de dados criando, assim, as respetivas tabelas. O próximo passo seria a implementação da API REST. Para tal, teria que ser bem estruturada e convergir com os requisitos do sistema. A estrutura da API, incluindo os seus *endpoints*, está descrita na Tabela 21.

Endpoint	Métodos HTTP	Descrição
mosaicos/	GET, POST	GET - Leitura de todos os mosaicos existentes POST - Criação de um ou mais mosaicos
mosaicos/<pk>/	GET, PUT, DELETE	GET – Leitura do mosaico especificado pela chave primária (pk) PUT – Atualização do mosaico especificado pela chave primária (pk) DELETE – Eliminação do mosaico especificado pela chave primária (pk)
mosaicos/<pk>/grupos	GET	GET – Leitura de todos os grupos existentes que pertençam ao mosaico especificado pela chave primária (pk)
grupos/	GET, POST	GET - Leitura de todos os grupos existentes POST - Criação de um ou mais grupos
grupos/<pk>/	GET, PUT, DELETE	GET – Leitura do grupo especificado pela chave primária (pk) PUT – Atualização do grupo especificado pela chave primária (pk) DELETE – Eliminação do grupo especificado pela chave primária (pk)
grupos/<pk>/mosaicos	GET	GET – Leitura de todos os mosaicos existentes que pertençam ao grupo especificado pela chave primária (pk)
camadas/	GET, POST	GET - Leitura de todas as camadas existentes POST - Criação de uma ou mais camadas
camadas/<pk>/	GET, PUT, DELETE	GET – Leitura da camada especificada pela chave primária (pk) PUT – Atualização da camada especificada pela chave primária (pk) DELETE – Eliminação da camada especificada pela chave primária (pk)
camadas/<pk>/overlays	GET	GET – Leitura de todos os overlays existentes que pertençam à camada especificada pela chave primária (pk)
overlays/	GET, POST	GET - Leitura de todos os overlays existentes POST - Criação de um ou mais overlays
overlays/<pk>/	GET, PUT, DELETE	GET – Leitura do overlay especificado pela chave primária (pk) PUT – Atualização do overlay especificado pela chave primária (pk) DELETE – Eliminação do overlay especificado pela chave primária (pk)
overlays/<pk>/target	GET	GET – Leitura do target que pertence ao overlay especificado pela chave primária (pk)
overlays/<pk>/camada	GET	GET – Leitura da camada que pertence ao overlay especificado pela chave primária (pk)
targets/	GET, POST	GET - Leitura de todos os targets existentes POST - Criação de um ou mais targets
targets/<pk>/	GET, PUT, DELETE	GET – Leitura do target especificado pela chave primária (pk) PUT – Atualização do target especificado pela chave primária (pk) DELETE – Eliminação do target especificado pela chave primária (pk)

targets/<sdk_id>/mosaico/	GET	GET – Leitura do mosaico que pertence ao target especificado pelo seu identificador no SDK (sdk_id)
transformations/	GET, POST	GET - Leitura de todas as transformações existentes POST - Criação de uma ou mais transformações
transformations/<pk>/	GET, PUT, DELETE	GET – Leitura da transformação especificada pela chave primária (pk) PUT – Atualização da transformação especificada pela chave primária (pk) DELETE – Eliminação da transformação especificada pela chave primária (pk)
transformations/<pk>/overlay	GET	GET – Leitura do overlay que pertence à transformação especificada pela chave primária (pk)

Tabela 21: Endpoints da API REST

Definida a estrutura da API, foram criados os respetivos serializadores e implementadas as funções (*views*) necessárias, de modo a que a API ficasse pronta a ser consumida pelo *frontend*. Os serializadores permitem a conversão de dados complexos como *querysets* e instâncias de modelos para tipos de dados nativos do Python que, por sua vez, serão traduzidos para JSON, XML ou outro tipo de conteúdo (Django REST Framework, n.d.).

Para garantir a segurança dos dados, a API teria que ser restringida a utilizadores autorizados. Teria que ser implementado um sistema que autorizasse, na mesma, a leitura de dados a qualquer utilizador, uma vez que a aplicação móvel não requer autenticação para a sua utilização e apenas utiliza a API para fins de leitura de dados, mas limitasse as restantes operações a utilizadores autenticados (que, neste caso, será o gestor de conteúdos). Para esse efeito, foi escolhida a autenticação por JWT (JSON Web Token) (Fig. 45). Estes *tokens* são representados por objetos JSON compostos por três elementos: *header*, *payload* e assinatura. O *header* contém informações sobre a forma como a assinatura deve ser gerada, incluindo o algoritmo de criptografia. O *payload* refere-se às informações (*claims*) sobre a entidade (normamente o utilizador autenticado), que podem ser reservadas, públicas ou privadas. Por fim, a assinatura resulta da concatenação dos *hashes* gerados a partir do *header* e do *payload*, utilizando codificação do tipo *base64Url*, a partir de uma chave secreta ou um certificado RSA. A integridade e autenticidade são garantidas também pela assinatura. No final, o JWT é representado por estes três componentes, separados por um ponto (“.”). Desta forma, o gestor de conteúdos terá que se autenticar na interface web, através do email e da password, e, após a sua validação, será criado um JWT com uma certa duração (que pode ser alterada manualmente, nas configurações do Django) e enviado para esse utilizador. A partir do momento que o gestor de conteúdos está autenticado, sempre que este quiser consumir a API REST, terá que enviar o JWT no *header* do pedido HTTP, de modo a que o servidor possa validar a assinatura desse JWT e confirmar a sua autenticidade (Stecky-Efantis, 2016). Todo este processo está devidamente automatizado e implementado de tal forma que o utilizador não se aperceba do mesmo.

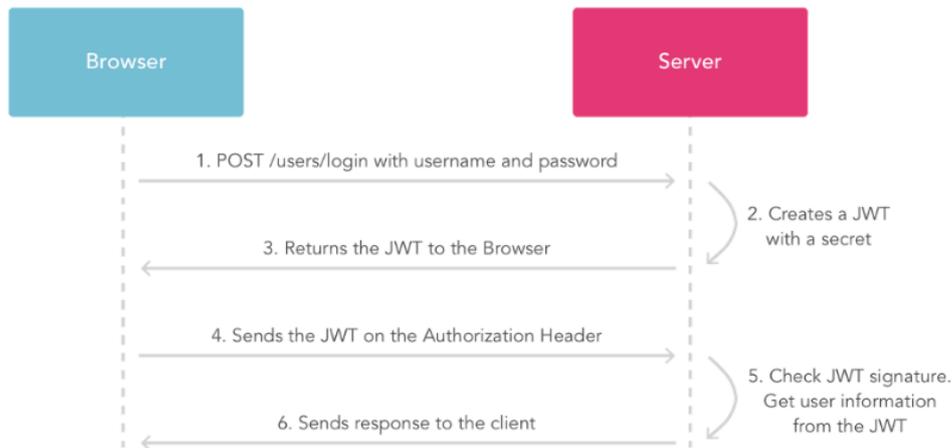


Figura 45: Diagrama de fluxo do processo de autenticação com JWT (Nascimento, 2018)

Uma vez que a aplicação móvel começou a ser desenvolvida primeiro que a interface web, enquanto esta última não estivesse funcional, os dados seriam todos observados, criados, editados e eliminados utilizando a interface fornecida pelo Django REST Framework (Fig. 46), que funcionava como uma API “navegável” (*browsable API*).

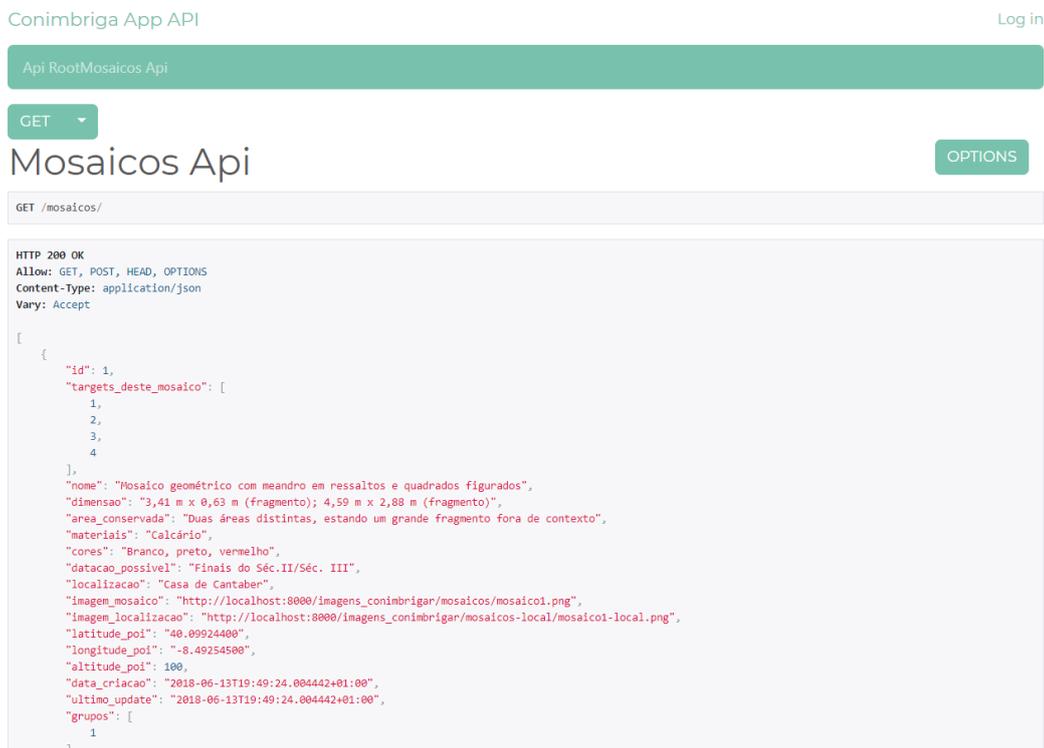


Figura 46: Interface da "Browsable API".

### 5.4.2. Interface Web

O início do desenvolvimento da interface web deu-se durante o desenvolvimento da aplicação móvel e, uma vez que ambas utilizam a *framework* Angular, o processo de desenvolvimento do *frontend* da interface tornou-se mais desembaraçado e relativamente rápido.

## Login

Uma vez que só o gestor de conteúdos pode aceder à interface web, foi implementado um sistema de *login* (Fig. 47), de modo a que este se pudesse autenticar. O processo de autenticação, baseado em JWT, foi descrito na seção anterior.

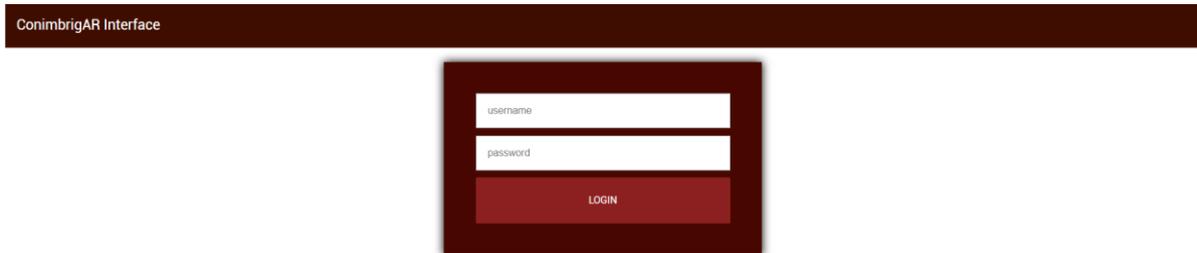


Figura 47: Janela de login

## Menu Principal

Caso o gestor de conteúdos insira as credenciais (email e password) corretas, será redireccionado para o menu principal da interface web (Fig. 48). Nesta janela, terá os cinco tipos de dados à disposição (*Mosaicos*, *Targets*, *Grupos de Mosaicos*, *Camadas de Informação* e *Overlays*) e a opção para fazer *logout* do sistema. Para cada um dos tipos de dados, o gestor terá as seguintes opções:

- **Ver todos**  
Opção para visualizar todos os objetos do tipo seleccionado
- **Gerir um <tipo>**  
Opção para gerir um objeto do tipo seleccionado (*e.g.*: *Gerir um mosaico*)
- **Adicionar um novo**  
Opção para criar um novo objeto do tipo seleccionado



Figura 48: Menu principal da interface web

## Ver Todos

A primeira opção, *Ver Todos*, permite ao gestor de conteúdos observar, numa nova página, uma listagem de todos os objetos do tipo escolhido, existentes na base dados, com as informações dos seus campos e o respetivo objeto JSON. Nas Figuras 49, 50 e 51 é possível observar um exemplo desta opção, selecionada para o tipo *Mosaicos*.

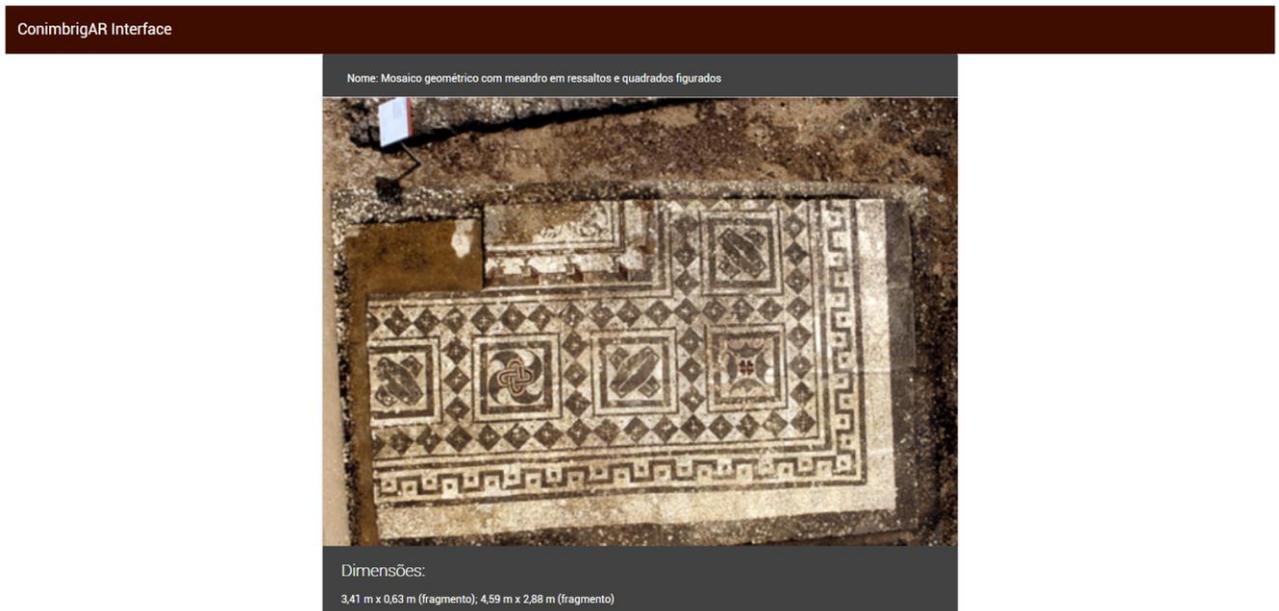


Figura 49: Opção *Ver Todos* - Listagem dos mosaicos (I)

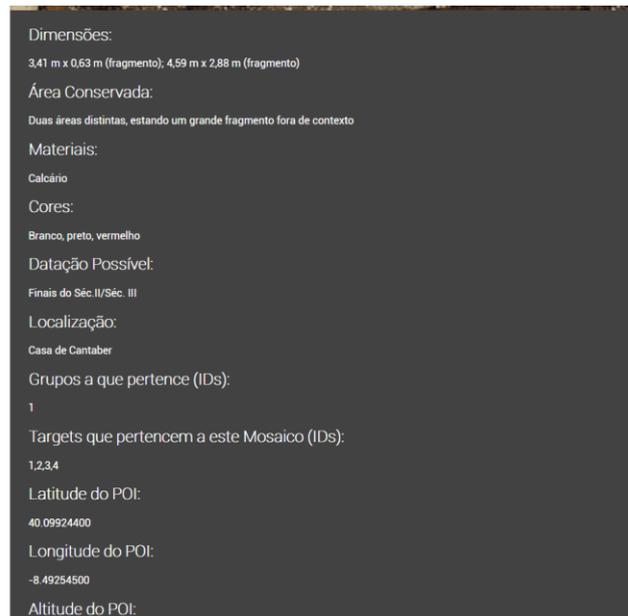


Figura 50: Opção *Ver Todos* - Listagem dos mosaicos (II)

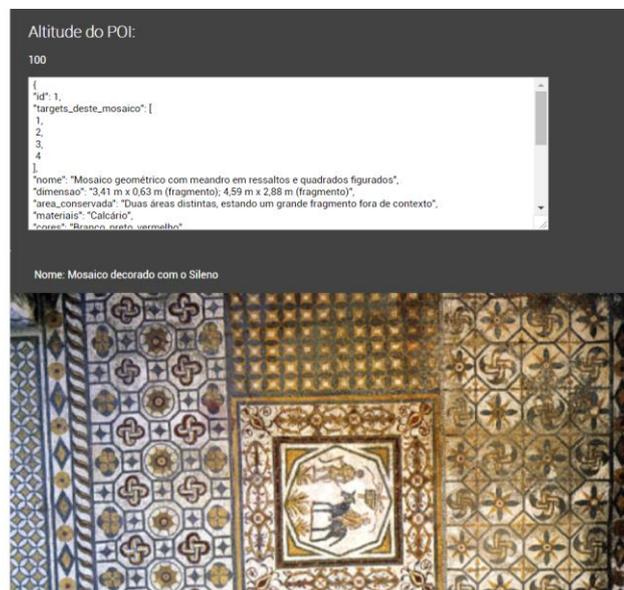


Figura 51: Opção *Ver Todos* - Listagem dos mosaicos (III)

## Gerir um <tipo>

A segunda opção, *Gerir um <tipo>*, permite ao gestor de conteúdos modificar informações sobre um objeto específico do respetivo tipo (<tipo>) ou, inclusive, eliminá-lo. Para efeitos de demonstração, pode ser observado um exemplo desta opção para o tipo *Target* (i.e., *Gerir um Target*) nas Figuras 52, 53 e 54. Ao selecionar esta opção, primeiramente, numa nova página, o gestor terá ao seu dispor uma lista de todos os objetos existentes do tipo escolhido (Fig. 52). O próximo passo será a seleção de um objeto dessa lista, de forma a ser possível geri-lo. Feita a seleção, o gestor será redireccionado para uma nova página (Fig. 53 e 54), onde poderá observar todas as informações acerca do objeto selecionado e um formulário que poderá ser preenchido, de forma a editar essas informações. Por fim, o gestor pode ainda eliminar o objeto, selecionado a respetiva opção.

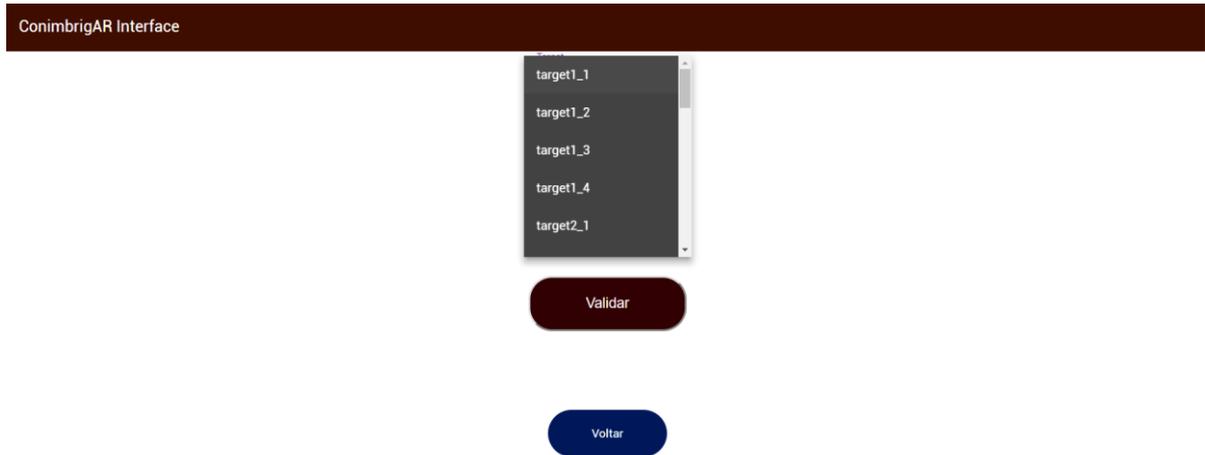


Figura 52: Opção *Gerir um Target* - Escolha do *target* a ser gerido

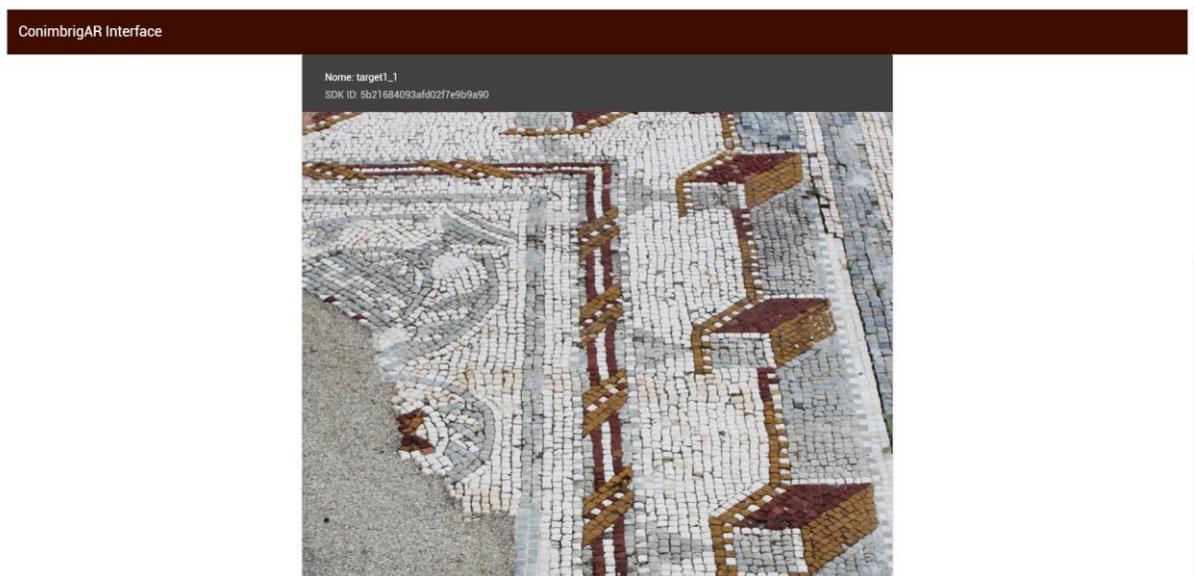


Figura 53: Opção *Gerir um Target* - Edição de um *target* (I)

Mosaico a que pertence (ID): 1  
Overlays que pertencem a este Target (IDs): 1

Nome

Descrição

Directório da Imagem

SDK ID

Editar

Eliminar

Voltar

Figura 54: Opção *Gerir um Target* - Edição de um *target* (II)

## Adicionar um novo

Por fim, o gestor de conteúdos pode seleccionar uma terceira opção, *Adicionar um novo*, para criar um novo objeto do tipo seleccionado. Para efeitos de demonstração, pode ser observado um exemplo desta opção para o tipo *Target* na Figura 55. Ao seleccionar esta opção, o gestor é então redireccionado para uma nova página (Fig. 55) com um formulário que deverá preencher com as informações do objeto a ser criado. Assim que o pedido for validado, o novo objeto será adicionado à base de dados.

ConimbrigAR Interface

Nome

Descrição

Imagem

Escolher ficheiro | Nenhum ficheiro seleccionado

SDK ID

Mosaico

Validar

Voltar

Figura 55: Opção *Adicionar um novo* – Criação de um objeto do tipo *Target*

### 5.4.3. Aplicação Móvel

#### Funcionamento offline

O *frontend* da aplicação móvel foi desenvolvido rigorosamente tendo em conta os diversos requisitos definidos no seu planeamento. Tendo isso em conta, um destes requisitos requiere o funcionamento offline de toda a aplicação móvel, incluindo as listagens e a experiência de RA. Na

primeira utilização da aplicação móvel, caso haja ligação à Internet, são transferidos todos os dados, com recurso à API, e estes são posteriormente armazenados localmente, numa base de dados SQLite. Para criar e ter acesso a estas bases de dados locais, é necessária a instalação do plugin nativo da Ionic<sup>15</sup> criado para o efeito. Desta forma, o utilizador poderá usufruir da aplicação móvel com recurso apenas aos dados locais. Tanto as listagens como a experiência de RA utilizam os dados locais, de maneira a que corram sem quaisquer problemas relacionados com a ligação à Internet. A metodologia para o funcionamento offline é a seguinte:

1. Ao entrar na aplicação móvel, verificar se existe alguma base de dados local
  - 1.1. Se não existir, transferir todos os dados e armazená-los na base de dados local
  - 1.2. Se existir, verificar se existem atualizações de dados. Esta verificação é feita recolhendo todos os dados da base de dados, observando e comparando o campo “ultimo\_update” dos mesmos e guardando o mais recente. Esta data é então comparada com a última data armazenada localmente, por forma a verificar se esta última é mais antiga.
    - 1.2.1. Se for mais antiga, os dados da base de dados locais são substituídos pelos dados transferidos.
    - 1.2.2. Se não for mais antiga, os dados transferidos não são armazenados localmente.

## Experiência de RA

A concretização da experiência de RA implica um processo metódico, que é iniciado pela seleção dos *targets*. Os *targets* escolhidos para a experiência terão que ser carregados na interface web da Wikitude (*Wikitude Studio*<sup>16</sup>) e, posteriormente, descarregados num único ficheiro “wtc”, denominado por coleção de *targets*. Esta coleção terá que ser integrada na aplicação móvel, de forma a que o SDK consiga interpretá-la e os seus *targets* possam ser utilizados.

Quando é iniciada a experiência de RA, é aberto um novo contexto (*i.e.*, *Webview*) e os dados locais referentes aos mosaicos, *targets*, camadas e *overlays* são enviados para esse contexto. Existem três conceitos fundamentais para entender a metodologia adoptada para a experiência de RA: camada atual, *target* atual e *overlay* atual. Inicialmente, a camada atual do sistema será a primeira camada existente nos dados locais. Esta pode ser mudada, sempre que estiver ativa uma experiência de RA, o que mudará também a camada atual. Uma experiência de RA está ativa a partir do momento em que um *target* é detetado. O *target* atual, todavia, não se trata do primeiro *target* detetado, mas sim do *target* que tiver *overlays* em comum com a camada atual. Este *target* terá então um *overlay* sobreposto, que será o *overlay* atual, e todos os restantes *targets* detetados simultaneamente terão os seus *overlays* ocultados. Além das camadas, também os *overlays* poderão ser mudados durante uma experiência, mudando consequentemente o *overlay* atual. A metodologia para a experiência RA é então a seguinte:

1. É detetado um *target*
2. Através do *target* detetado, são armazenados todos os *targets* do mosaico pertencente.
3. São comparados todos os *overlays* pertencentes a estes *targets* com os *overlays* pertencentes à camada atual
  - 3.1. Caso existam *overlays* em comum, estes serão armazenados e tidos em conta para a experiência de RA no mosaico atual. O primeiro *overlay*

<sup>15</sup> <https://ionicframework.com/docs/native/sqlite/>

<sup>16</sup> <https://studio.wikitude.com/>

dessa lista será o *overlay* atual e o *target* pertence a este *overlay* será o *target* atual.

- 3.2. Caso não existam *overlays* em comum, não irá surgir qualquer *overlay* sobre o mosaico atual e terá que ser escolhida outra camada. O passo 3 é repetido.

## Listagem de Dados

A implementação da listagem dos dados foi relativamente simples. A metodologia para a listagem de dados é a seguinte:

1. Aceder à *tab* das listagens
2. Escolher o tipo de dados desejado (*Mosaicos*, *Grupos* ou *Camadas*)
  - 2.1. Ao escolher o tipo *Mosaicos*, irá surgir uma listagem com os nomes de todos os mosaicos existentes na base de dados.
  - 2.2. Ao escolher o tipo *Grupos*, irá surgir uma listagem com os nomes de todos os grupos existentes na base de dados.
  - 2.3. Ao escolher o tipo *Camadas*, irá surgir uma listagem com os nomes de todos os grupos existentes na base de dados.
3. Escolher um objeto do tipo de dados desejado
  - 3.1. Caso tenha sido um objeto do tipo *Mosaicos*, serão observadas todas as informações acerca do respetivo objeto, juntamente com uma listagem dos grupos a que pertence.
    - 3.1.1. Ao selecionar um grupo da listagem de grupos a que o mosaico pertence, o utilizador será redirecionado para uma janela com todas as informações acerca do grupo escolhido.
  - 3.2. Caso tenha sido um objeto do tipo *Grupos*, serão observadas todas as informações acerca do respetivo objeto, juntamente com uma listagem dos mosaicos pertencentes.
    - 3.2.1. Ao selecionar um mosaico da listagem de mosaicos pertencentes ao grupo, o utilizador será redirecionado para uma janela com todas as informações acerca do mosaico escolhido.
  - 3.3. Caso tenha sido um objeto do tipo *Camadas*, serão observadas todas as informações acerca do respetivo objeto

Na próxima seção, será demonstrado o resultado final da aplicação móvel, após terem sido implementadas as metodologias descritas anteriormente.

## 5.5. Protótipo Final

O protótipo final da aplicação móvel foi terminado em junho e cumpriu com todos os requisitos de maior prioridade. Nesta seção, será feita uma demonstração do protótipo, incluindo as suas funcionalidades e a sua navegabilidade.

## Janela inicial



Figura 56: Janela inicial da aplicação móvel

A janela inicial é representada pela *tab* esquerda da aplicação, que diz respeito à experiência de RA (Fig. 56). Assim que o utilizador entre na aplicação, poderá então iniciar a experiência de imediato, clicando no respetivo botão *Iniciar Experiência!*. Caso contrário, existem mais duas *tabs* que o utilizador poderá consultar e que dizem respeito às listagens (*tab* central) e ao tutorial da aplicação (*tab* direita).

## Listagens



Figura 57: Janela da listagem de mosaicos (imagem à esquerda), de grupos (imagem central) e de camadas (imagem à direita)

Ao selecionar a *tab* das listagens, o utilizador observará primeiro a listagem dos mosaicos. No entanto, na mesma janela, o utilizador pode consultar também a listagem de grupos e a listagem de camadas (Fig. 57), se assim pretender. Estas listagens são compostas apenas pelos nomes dos respetivos objetos e por um pequeno ícone, contendo a sua imagem. Para observar as informações sobre um determinado mosaico, grupo ou camada, o utilizador terá que o selecionar na sua respetiva listagem.

## Informações sobre um Mosaico

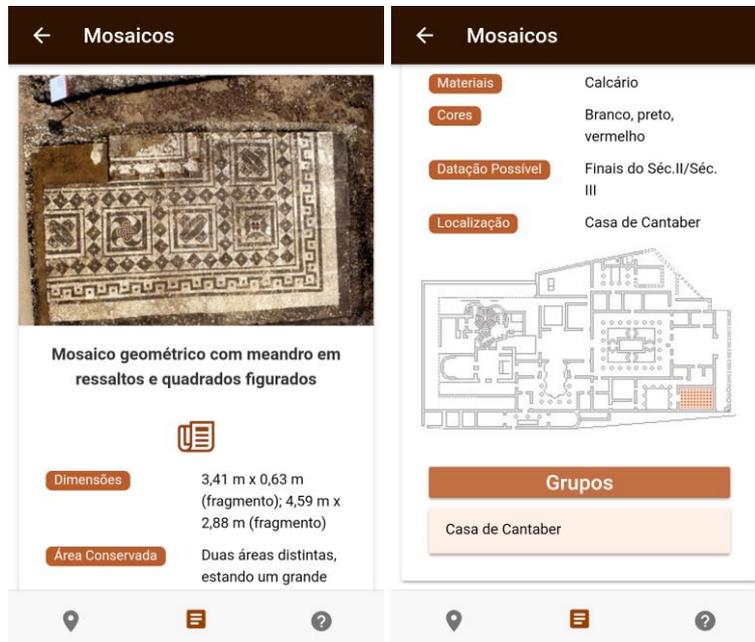


Figura 58: Janela das informações sobre um mosaico

Selecionando um mosaico, o utilizador irá observar uma imagem do mesmo, seguido de um conjunto de informações incluindo: nome, dimensões, área conservada, materiais, cores, datação possível e nome da localização (Fig. 58). Além disso, o utilizador poderá observar a marcação do mosaico na planta do local onde se encontra. Por fim, é disponibilizada uma lista com os nomes dos grupos a que o mosaico pertence. Ao clicar num destes grupos, o utilizador será redirecionado para a janela de informações sobre o respetivo grupo.

## Informações sobre um Grupo



Figura 59: Janela das informações sobre um grupo

Selecionando um grupo, o utilizador irá observar uma imagem do mesmo, seguido por uma descrição e pelo conjunto de mosaicos que lhe são pertencentes (Fig. 59). Ao clicar num destes mosaicos, o utilizador será redirecionado para a janela de informações sobre o respetivo mosaico.

## Informações sobre uma Camada



Figura 60: Janela das informações sobre uma camada

Selecionando uma camada, o utilizador irá observar uma imagem ilustrativa da mesma, seguido pela descrição daquilo que os seus *overlays* representam (Fig. 60).

## Experiência de RA



Figura 61: Janela da experiência de RA, sem qualquer target detetado

Ao iniciar uma experiência de RA, o utilizador será redirecionado para o contexto da Wikitude e será ligada a câmara do *smartphone*. Até ser detetado algum *target*, a janela consistirá apenas do ambiente real captado pela câmara, pela mensagem *ENCONTRA UM MOSAICO* e pelo logótipo da Wikitude (Fig. 61).

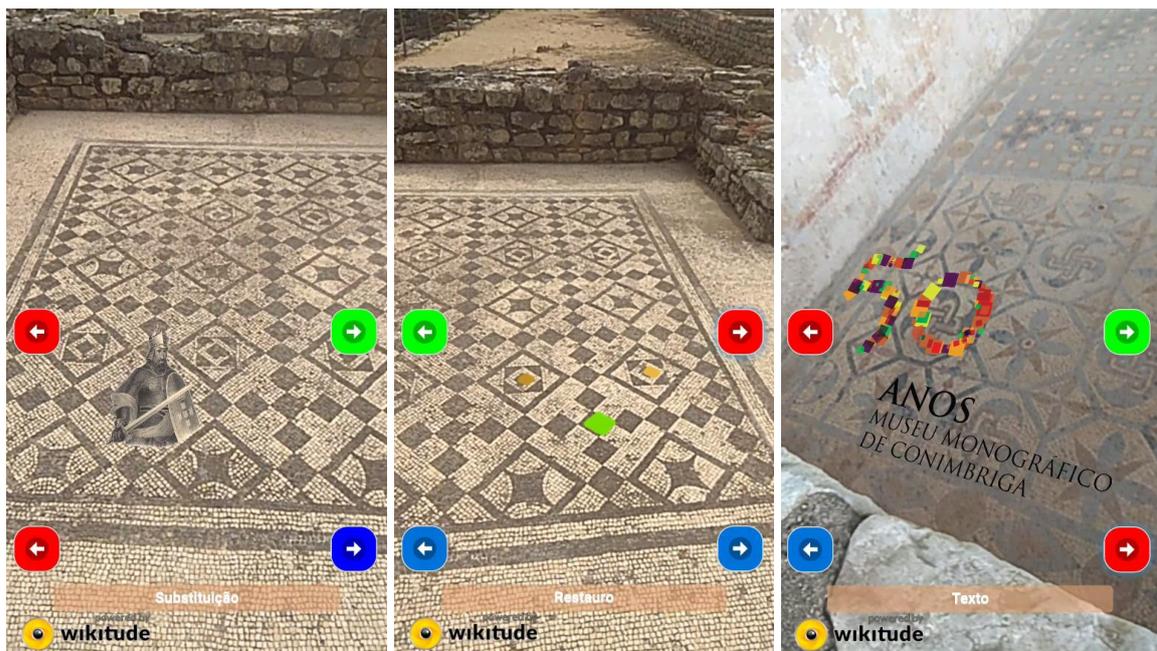


Figura 62: Janela da experiência de RA, com *overlays* de diversas camadas

Caso o utilizador aponte a câmara para um mosaico que contenha uma experiência de RA, assim que um dos seus *targets* for detetado, essa experiência será iniciada (Fig. 62). No fundo do ecrã, irá aparecer o nome da camada ativa/actual e, ligeiramente por cima, dois botões (representados por uma seta à esquerda e uma seta à direita, respetivamente) para alternar entre camadas. O utilizador pode também alternar entre os *overlays*, utilizando os botões (representados por uma seta à esquerda e uma seta à direita, respetivamente) localizados a meio do ecrã. Tanto

para as camadas como para os *overlays*, quando um botão apresentar a cor vermelha, será um indício de que não será possível navegar mais na direção correspondente. Caso contrário, ainda haverá mais conteúdo nessa direção. Assim que não houver qualquer *target* a ser detetado pelo Wikitude, a experiência de RA será interrompida e todos os botões irão desaparecer, ressurgindo quando for novamente detetado algum *target*.

No caso de não haver *overlays* em comum entre a camada atual/ativa e os *targets* do mosaico atual, irá surgir uma mensagem de aviso no ecrã e os botões de alternância de *overlays* ficaram ambos com cor vermelha. Além disso, no caso de o utilizador clicar num botão vermelho, irá aparecer a mensagem de aviso “Não há mais overlays!”, no caso de um botão de alternância de *overlays*, ou a mensagem “Não há mais camadas!”, no caso de um botão de alternância de camadas. (Fig. 63)

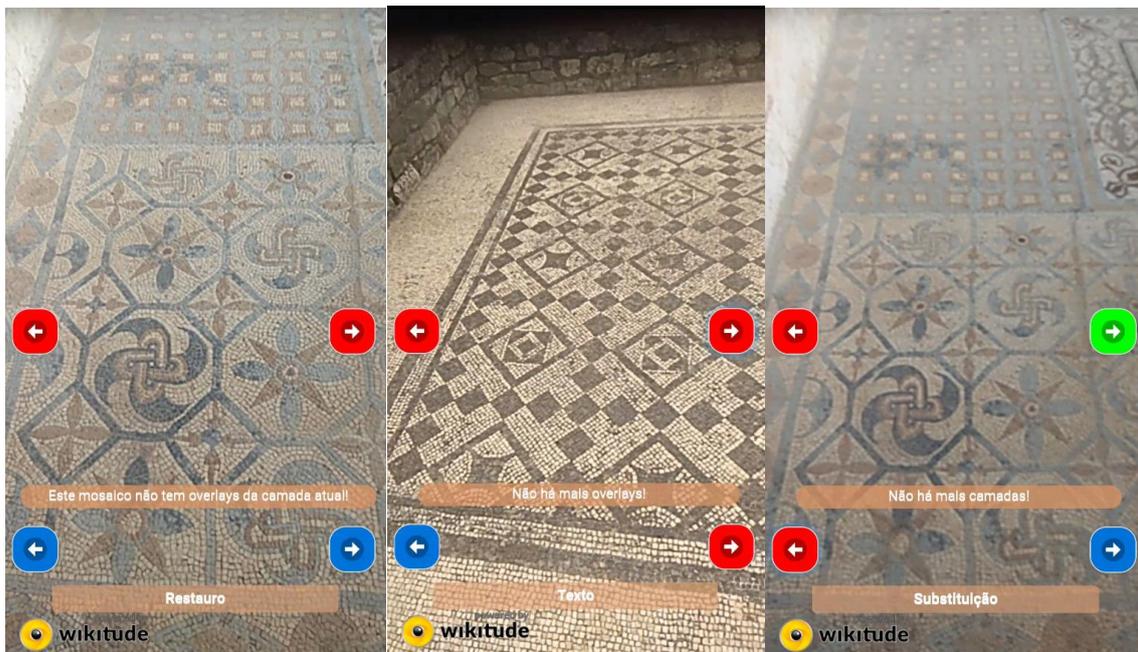


Figura 63: Janela da experiência de RA, com as mensagens de aviso do sistema.

Apesar de os Pontos de Interesse terem sido implementados, estes não foram testados até à finalização do protótipo final (incluindo os testes de usabilidade), devido a alguns ajustes que ainda necessitavam de ser feitos de modo a que estes funcionassem corretamente. No entanto, é esperado que o utilizador consiga observar estes Pontos de Interesse a “flutuar” sobre os respetivos mosaicos. Haverá depois tipos de Pontos de Interesse, distinguíveis ao olho do utilizador: normais e com experiência de RA. Desta forma, o utilizador pode descobrir quais os mosaicos que contêm experiências de RA. Por fim, ao clicar num destes Pontos de Interesse, surgirá uma pequena janela do lado direito do ecrã, com as informações do respetivo mosaico (as mesmas que podem ser consultadas através da listagem de mosaicos).

## 5.6. Testes de Usabilidade

Terminado o protótipo final da aplicação móvel, seria novamente altura de realizar testes em Conímbriga, mas, desta vez, com a participação de visitantes. Para tal, foram realizados testes de usabilidade. De modo a garantir alguma heterogeneidade nos resultados, o foco da seleção dos participantes recaiu sobre a diversidade, em vez de quantidade. Tendo isso em conta, teriam que ser escolhidos participantes de diferentes faixas etárias e com conhecimentos diferentes acerca da tecnologia de Realidade Aumentada, acerca do Património Cultural e acerca das ruínas de

Conímbriga. Resumindo os inquéritos pré-testes encontrados no Anexo C1, os perfis dos participantes nos testes de usabilidade podem ser observados na Tabela 22.

	Participante #1	Participante #2	Participante #3	Participante #4
<b>Idade</b>	68	24	27	58
<b>Visitou Conímbriga (Sim/Não)</b>	Sim	Sim	Sim	Sim
<b>Tem noção considerável da localização dos mosaicos em Conímbriga (Sim/Não)</b>	Não	Não	Não	Sim
<b>Importância do Património Cultural (Pouco/Suficiente/Bastante)</b>	Bastante	Bastante	Bastante	Bastante
<b>Já alguma vez utilizou a tecnologia de RA (Sim/Não)</b>	Não	Não	Sim	Não
<b>Facilidade em adaptar-se a novas aplicações de <i>smartphone</i> (Pouco/Suficiente/Bastante)</b>	Suficiente	Bastante	Bastante	Suficiente

**Tabela 22:** Inquérito pré-testes

Foram reunidos 4 participantes minimamente distintos, como é possível observar pela Tabela 22. Os nomes foram suprimidos, por questões de privacidade. Embora todos eles já tenham visitado as ruínas de Conímbriga, apenas um participante tem uma noção considerável da localização dos mosaicos em Conímbriga. Este fator pode ser um indicativo da necessidade dos Pontos de Interesse e da mais valia que estes podem representar numa visita a Conímbriga. Como seria de esperar, todos os participantes valorizam o Património Cultural, pelo que a dinamização dos mosaicos romanos de Conímbriga seria, à partida, do agrado destes participantes. No entanto, apenas um destes participantes já experimentou a tecnologia de RA, pelo que seria incerto saber *a priori* o que os restantes achavam desta abordagem para dinamizar as visitas a Conímbriga.

Os testes de usabilidade foram realizados em Conímbriga, numa manhã com condições de luminosidade minimizante aceitáveis (*i.e.*, céu coberto, mas sem a presença de sombras nos mosaicos). Estes testes seriam realizados a partir do mesmo *smartphone*, pelo que os participantes teriam que realizar as tarefas à vez. As tarefas foram as seguintes e pela ordem que seguem:

1. Consultar os detalhes de um mosaico que contenha um centauro.
2. Consultar as cores do mosaico anterior
3. Consultar os grupos do mosaico anterior e aceder ao grupo “Casa dos Repuxos”
4. Consultar novamente o mosaico que contém o centauro
5. Consultar os detalhes sobre a camada de “Substituição”
6. Iniciar uma experiência de RA
7. Encontrar a imagem do Presidente Marcelo Rebelo de Sousa sobreposta no mosaico

8. Contar o número de imagens sobrepostas que foram detetadas
9. Ler o texto escrito na imagem sobreposta “50 Anos Museu Monográfico De Conímbriga”

Este conjunto de tarefas foi executado sequencialmente. Enquanto os participantes realizavam as tarefas, estes teriam que explicar, sempre que possível, o seu raciocínio para as concretizar. Não seria dada qualquer ajuda ou dica, a não ser que o participante estivesse mais tempo do que o esperado, na respetiva tarefa. Após terem realizado todas as tarefas, os participantes teriam que preencher um inquérito pós-testes, de forma a fornecerem *feedback*.

Os resultados dos testes de usabilidade foram interessantes, na medida em que os participantes sentiram algumas dificuldades semelhantes e outras bastante distintas. Com base na prestação dos participantes (gravada utilizando o *ScreenRecorder*) e nos inquéritos pós-testes, encontrados no Anexo C2, foi possível retirar as seguintes conclusões:

- Todos os participantes sentiram dificuldades no acesso às informações sobre um mosaico. Estes, inclusive, optaram por descobrir as cores de um mosaico observando a sua imagem, ao invés de consultar as suas informações.
- O termo “camada” mostrou ser confuso para os participantes. Quando lhes foi pedido para consultarem as informações sobre uma camada, estes não sabiam como agir. Além disso, durante a experiência de RA, a presença do nome da camada no ecrã, assim como as respetivas setas de alternância, deixou os participantes algo emaranhados. No final, o termo acabou por fazer sentido aos participantes.
- Aquando questionados sobre a camada de informação que mais gostaram, os participantes deram respostas diferentes, pelo que é possível afirmar que cada visitante terá as suas preferências e gostos pessoais no que toca um conjunto de *overlays* que pretende observar sobre os mosaicos.

Apesar das dificuldades sentidas pelos participantes, estes conseguiram-se adaptar, em pouco tempo, à interface e à navegabilidade da aplicação móvel. Além disso, a apreciação em relação à experiência de RA foi, no geral, positiva e a iniciativa em si foi bastante louvada, sendo seguro afirmar que esta aplicação móvel não deixará os visitantes de Conímbriga indiferentes.

## 6. Conclusão e Trabalho Futuro

Terminada esta Dissertação, é possível retirar algumas conclusões importantes nas áreas de Realidade Aumentada e de Património Cultural. Tendo sido proposta a análise dos SDKs de Realidade Aumentada existentes atualmente, foi possível obter um leque considerável de opções disponíveis no mercado. Tendo isso em conta, existem bastantes e diversas opções de ferramentas para integrar a tecnologia de Realidade Aumentada nas aplicações móveis e, conseqüentemente, desenvolver projetos interessantes. Para o desenvolvimento híbrido, no entanto, as ofertas ainda são poucas. Os SDKs que permitem a integração de Realidade Aumentada em aplicações móveis híbridas foram investigados e testados, de forma a que fossem selecionados os mais adequados tendo em conta os requisitos exigidos para o projeto em mão. No final, o SDK Wikitude acabaria por ser a única escolha fiável e, apesar de não ter apresentado resultados positivos no seu desempenho nos testes aos SDKs, revelou ser uma ferramenta bastante completa, com diversas funcionalidades e com um potencial que deve ser explorado em experiências de Realidade Aumentada no exterior.

Apesar de não terem sido cumpridos todos os requisitos delineados para este projeto, foi possível criar um protótipo funcional da aplicação móvel para produzir experiências de RA sobre os mosaicos romanos de Conímbriga. Foi desenhada uma arquitetura que permite a utilização da aplicação móvel em modo offline, que pode ser aproveitada e melhorada no futuro, assim como o modelo de dados criado. Os testes de usabilidade realizados no final mostraram evidências de possíveis problemas do sistema. Foram encontrados vários pontos que devem ser melhorados na interface e estes estão disponíveis neste documento, de maneira a que possam ser utilizados. Sempre que forem feitas novas alterações à aplicação móvel, devem ser conduzidos novos testes de usabilidade, de maneira a garantir a sua qualidade.

Esta Dissertação pode servir de inspiração para novos projetos relacionados com a Realidade Aumentada, sobretudo no que toca à sua aplicação em espaços de Património Cultural. Havendo uma grande abundância destes espaços no nosso país, o leque de ideias e possibilidades que a Realidade Aumentada pode produzir é inimaginável. Esta tecnologia poderá dinamizar estes espaços, tornar as suas visitas mais atrativas e, com a sua evolução, certamente será ter a sua presença assegurada, futuramente, em alguns locais de Património Cultural. No caso de Conímbriga, os visitantes poderão passar por uma experiência imersiva com os mosaicos romanos, observando-os de maneiras distintas e constatando detalhes agora inexistentes ou difíceis de observar a olho nu. Desta forma, será promovida a arte, a cultura e a educação, através de uma experiência.

Embora a quantidade de mosaicos romanos de Conímbriga que o Wikitude consegue detetar ainda seja pequena, estes podem ser aproveitados e trabalhados ao máximo, enquanto os restantes mosaicos são investigados, no sentido de possibilitar a tecnologia de Realidade Aumentada sobre eles. Esta investigação poderá debruçar-se em tópicos como a forma como os *targets* devem ser registados, as condições de luminosidade mais favoráveis à deteção dos *targets*, o tipo de padrões e texturas dos mosaicos romanos mais fácil ou dificilmente detetáveis, etc. Ao prosseguir com esta investigação, não só a aplicação móvel ficará mais rica como será feita uma contribuição importante à ferramenta da *Wikitude* e à área da Realidade Aumentada em si. Esta tecnologia trará bastantes benefícios à nossa cultura, pelo que terá que ser explorada e apostada num futuro próximo.

## 7. Referências

- Abed, R. (2016). Hybrid vs Native Mobile Apps - The Answer is Clear. Retrieved from <https://ymedialabs.com/hybrid-vs-native-mobile-apps-the-answer-is-clear/>
- Addison, A. C. (2000). Emerging trends in virtual heritage. *IEEE Multimedia*, 7(2), 22–25. <https://doi.org/10.1109/93.848421>
- Alexseyenk, A. (2017). Native vs Hybrid apps. What to choose in 2017? Retrieved from <https://blog.techmagic.co/native-vs-hybrid-apps/>
- Amin, D., & Govilkar, S. (2015). Comparative Study of Augmented Reality Sdk's. *International Journal on Computational Science & Applications*, 5(1), 11–26. <https://doi.org/10.5121/ijcsa.2015.5102>
- Archeomaticamagazine. (2013). Archeoguide ad Olympia. Retrieved November 2, 2017, from <https://www.youtube.com/watch?v=Nva0Zns3cRc>
- Azuma, R. (1997). A Survey of Augmented Reality. *Presence: Teleoperators and Virtual Environments* 6, 4. <https://doi.org/10.1561/1100000049>
- Azuma, R. (2017). Making Augmented Reality a Reality. *Boxoffice*, 148(7), 29–30, 32. Retrieved from [http://search.proquest.com/docview/1032852311?accountid=10344%5Cnhttp://sfx.unilinc.edu.au:9003/csu?url\\_ver=Z39.88-2004&rft\\_val\\_fmt=info:ofi/fmt:kev:mtx:journal&genre=unknown&sid=ProQ:ProQ%3Aaipft&atitle=Making+a+Reality+a+Reality&title=Boxoffice&issn=00](http://search.proquest.com/docview/1032852311?accountid=10344%5Cnhttp://sfx.unilinc.edu.au:9003/csu?url_ver=Z39.88-2004&rft_val_fmt=info:ofi/fmt:kev:mtx:journal&genre=unknown&sid=ProQ:ProQ%3Aaipft&atitle=Making+a+Reality+a+Reality&title=Boxoffice&issn=00)
- Azuma, R., Billinghurst, M., & Klinker, G. (2011). Special Section on Mobile Augmented Reality. *Computers & Graphics*, 35(4), vii–viii. <https://doi.org/10.1016/j.cag.2011.05.002>
- Billinghurst, M., Clark, A., & Lee, G. (2015). A Survey of Augmented Reality. *Foundations and Trends® in Human-Computer Interaction*, 8(2–3), 73–272. <https://doi.org/10.1561/1100000049>
- BloomIdea. (2016). Aplicações Nativas vs Híbridadas: qual escolher para o seu projeto. Retrieved from <https://bloomidea.com/blog/aplicacoes-nativas-vs-hibridadas-qual-escolher-para-o-seu-projeto>
- Bostanci, E., Kanwal, N., Ehsan, S., & Clark, A. F. (2013). User Tracking Methods for Augmented Reality. *International Journal of Computer Theory and Engineering*, 93–98. <https://doi.org/10.7763/IJCTE.2013.V5.654>
- Bristowe, J. (2015). What is a Hybrid Mobile App? Retrieved from <https://developer.telerik.com/featured/what-is-a-hybrid-mobile-app/>
- Brogni, B. A., Avizzano, C. A., Evangelista, C., & Bergamasco, M. (1999). Technological approach for cultural heritage: augmented reality. *Robot and Human Interaction, 1999. RO-MAN '99. 8th IEEE International Workshop On*, 206–212. <https://doi.org/10.1109/ROMAN.1999.900341>
- Comport, A. I., Marchand, É., & Chaumette, F. (2003). A real-time tracker for markerless augmented reality. In *Proceedings - 2nd IEEE and ACM International Symposium on Mixed and Augmented Reality, ISMAR 2003* (pp. 36–45). <https://doi.org/10.1109/ISMAR.2003.1240686>

- Comport, A. I., Marchand, E., Pressigout, M., & Chaumette, F. (2006). Real-time markerless tracking for augmented reality: The virtual visual servoing framework. In *IEEE Transactions on Visualization and Computer Graphics* (Vol. 12, pp. 615–628). <https://doi.org/10.1109/TVCG.2006.78>
- Dalí, S. (1981). *The Unspeakable Confessions of Salvador Dalí*. Quill.
- Damiani, J. (2017). How Augmented Reality Audio Will Let You Soundtrack Your Life. Retrieved from <https://www.billboard.com/articles/business/7998347/how-augmented-reality-audio-soundtrack-your-life>
- Django REST Framework. (n.d.). Serializers. Retrieved from <http://www.django-rest-framework.org/api-guide/serializers/>
- Donga, H. (2017). Top 10 Mobile App Development Frameworks in 2017. Retrieved from <https://www.addonsolutions.com/blog/top-10-mobile-app-development-frameworks.html>
- Drum, T. (2016). The Economist RecoVR: Mosul. Retrieved from <http://www.thedrum.com/profile/project/246701/economist-recovr-mosul>
- El-Hakim, S. F., Beraldin, J. A., Picard, M., & Godin, G. (2004). Detailed 3D reconstruction of large-scale heritage sites with integrated techniques. *IEEE Computer Graphics and Applications*, 24(3), 21–29. <https://doi.org/10.1109/MCG.2004.1318815>
- Fischler, M. a, & Bolles, R. C. (1981). Random Sample Consensus: A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography. *Communications of the ACM*, 24(6), 381–395. <https://doi.org/10.1145/358669.358692>
- Fritz, F., Susperregui, A., & Linaza, M. T. (2005). Enhancing cultural tourism experiences with augmented reality technologies. *The 6th International Symposium on Virtual Reality Archaeology and Cultural Heritage VAST*, 20–21. Retrieved from <http://public-repository.epoch-net.org/publications/VAST2005/shortpapers/short2005.pdf>
- Gajotres. (2017). Top 7 JavaScript Hybrid Mobile Application Frameworks - Pros/Cons. Retrieved from <https://www.gajotres.net/top-7-mobile-application-html5-frameworks/>
- Gîrbacia, F., Butnariu, S., Orman, A. P., & Postelnicu, C. C. (2013). Virtual restoration of deteriorated religious heritage objects using augmented reality technologies. *European Journal of Science and Theology*, 9(2), 223–231.
- Goodby Silverstein & Partners. (2016). Dreams of Dalí. Retrieved from <https://goodbysilverstein.com/work/dreams-of-dali-2>
- Hansen, S. (2017). Advantages and Disadvantages of Django. *Hackernoon*. Retrieved from <https://hackernoon.com/advantages-and-disadvantages-of-django-499b1e20a2c5>
- Hoff, W. a, Nguyen, K., & Lyon, T. (1996). Computer vision-based registration techniques for augmented reality. *Proceedings of Intelligent Robots and Computer Vision XV, SPIE, 2904*, 1–10. <https://doi.org/10.1117/12.256311>
- Ibrahim, N., Mohamad Ali, N., & Mohd Yatim, N. F. (2011). Cultural learning in virtual heritage: An overview. In *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)* (Vol. 7067 LNCS, pp. 273–283). [https://doi.org/10.1007/978-3-642-25200-6\\_26](https://doi.org/10.1007/978-3-642-25200-6_26)
- INKONIQ. (2017). Top 7 Reasons to Pick IONIC Framework for Mobile Apps. *Medium*.

- Retrieved from <https://medium.com/inkoniq-blog/top-7-reasons-to-pick-ionic-framework-for-mobile-apps-30169275fb58>
- Jscrambler. (2017). 10 Frameworks for Mobile Hybrid Apps. Retrieved from <https://blog.jscrambler.com/10-frameworks-for-mobile-hybrid-apps/>
- Kanzig, T. (2015). Aplicações Móveis Híbridas. Retrieved from <https://pt.slideshare.net/thomaskanzig/aplicaes-mveis-hbridas>
- Krevelen, D. W. F. van, & Poelman, R. (2010). A Survey of Augmented Reality Technologies, Applications and Limitations. *The International Journal of Virtual Reality*, 9(2), 1–20. <https://doi.org/10.1155/2011/721827>
- Lima, J. P., Roberto, R., Teichrieb, V., & Marques, G. (2015). Study about Natural Feature Tracking for Augmented Reality Applications on Mobile Devices | Estudo sobre Rastreamento Baseado em Características Naturais para Aplicações de Realidade Aumentada em Dispositivos Móveis. *17th Symposium on Virtual and Augmented Reality, SVR*, 7–14. <https://doi.org/10.1109/SVR.2015.9>
- Looper, J. (2015). What is a WebView? Retrieved from <https://developer.telerik.com/featured/what-is-a-webview/>
- Martins, L. B. (2016). VisitAR - Realidade Aumentada | IT People Innovation. Retrieved from <https://pt.slideshare.net/LuisBravoMartins/visitar-realidade-aumentada-it-people-innovation>
- Milgram, P., Takemura, H., Utsumi, A., & Kishino, F. (1995). Augmented reality: a class of displays on the reality-virtuality continuum, 2351, 282–292. <https://doi.org/10.1117/12.197321>
- Monico, N. (2016). The Ruins of Byblos, An Ancient Playground for the Affluent. Retrieved from <http://www.justluxe.com/travel/luxury-vacations/feature-1963516.php>
- Mullen, T. (2011). *Prototyping Augmented Reality*. (JOHN WILEY & SONS INC, Ed.). Sybex.
- Naimark, M. (2016). VR / AR Fundamentals — 3) Other Senses (Touch, Smell, Taste, Mind)Michael Naimark. *Medium*. Retrieved from <https://medium.com/@michaelnaimark/vr-ar-fundamentals-3-other-senses-haptic-smell-taste-mind-e6d101d752da>
- Nascimento, W. (2018). Entendendo tokens JWT (Json Web Token). *Medium*. Retrieved from <https://medium.com/tableless/entendendo-tokens-jwt-json-web-token-413c6d1397f6>
- Neumann, U., & Cho, Y. (1996). A self-tracking augmented reality system. *Proceedings of the ACM Symposium on Virtual Reality Software and Technology*, 1–7. Retrieved from <http://graphics.usc.edu/cgit/publications/papers/STAR-ACM-VRST96.pdf>
- Neumann, U., & Park, J. (1998). Extendible object-centric tracking for augmented reality. *Virtual Reality Annual International Symposium, 1998. Proceedings., IEEE*, 148–155.
- News, E. (2015). Augmented Reality. *Ercim News*, October 20(103), 28–29.
- Noh, Z., & Sunar, M. S. (2009). A Review on 3D Reconstruction for Augmented Reality in Virtual Heritage System. *Learning by Playing*, 50–61. [https://doi.org/10.1007/978-3-642-03364-3\\_7](https://doi.org/10.1007/978-3-642-03364-3_7)
- Pardes, A. (2017). IKEA's new app flaunts what you'll love most about AR. Retrieved from <https://www.wired.com/story/ikea-place-ar-kit-augmented-reality/>

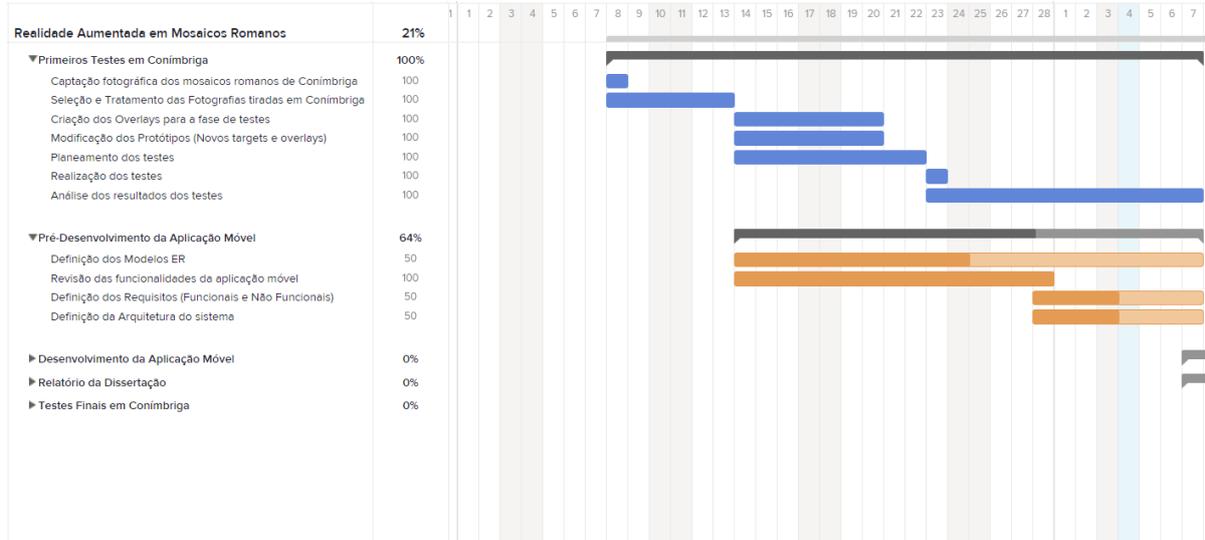
- Patil, R. (2016). Pros and Cons of Cross-Platform Mobile App Development. Retrieved from <https://www.infoq.com/articles/mobile-cross-platform-app-development>
- Pavlidis, G., Koutsoudis, A., Arnaoutoglou, F., Tsioukas, V., & Chamzas, C. (2007). Methods for 3D digitization of Cultural Heritage. *Journal of Cultural Heritage*, 8(1), 93–98. <https://doi.org/10.1016/j.culher.2006.10.007>
- Petit, A. (2013). Robust model-based tracking for markerless complex objects. Retrieved from <https://www.youtube.com/watch?v=PtdxNAvso08>
- Rami A. (2017). Mobile app development: Native vs cross-platform vs hybrid. Retrieved from <https://www.linkedin.com/pulse/mobile-app-development-native-vs-cross-platform-hybrid-rami-assi>
- Reitmayr, G., & Drummond, T. W. (2007). Going out: Robust model-based tracking for outdoor augmented reality. In *Proceedings - ISMAR 2006: Fifth IEEE and ACM International Symposium on Mixed and Augmented Reality* (pp. 109–118). <https://doi.org/10.1109/ISMAR.2006.297801>
- Rodrigues, M. V. (2016). Realidade aumentada para promover a História e a Cultura. Retrieved October 14, 2017, from <http://observador.pt/2016/12/23/it-people-innovation-app-visitatar/>
- Roussou, M. (2000). Virtual heritage: from the research lab to the broad public. *Virtual Reality*, (July), 93–101.
- Sacomani, P. (2017). Native, Web or Hybrid Apps? What's The Difference? Retrieved from <https://www.mobiloud.com/blog/native-web-or-hybrid-apps/>
- Sanders, D. H. (2007). Why Do Virtual Heritage? *Proceedings from the 34th Computer Applications and Quantitative Methods in Archaeology Conference Fargo ND USA*, 563–572. Retrieved from <http://www.archaeology.org/online/features/virtualheritage/>
- Schmalstieg, D., Schmalstieg, D., Wagner, D., & Wagner, D. (2008). Mobile Phones as a Platform for Augmented Reality. *Proceedings of the IEEE VR 2008 Workshop on Software Engineering and Architectures for Realtime Interactive Systems (Reno, NV, USA)*, Pp. 43-44, 43–44. <https://doi.org/10.1.1.144.6950>
- Scopigno, R., Gabellone, F., Malomo, L., Banterle, F., Pingi, P., Amato, G., & Cardillo, F. A. (2015). LecceAR: An Augmented Reality App. *Digital Presentation and Preservation of Cultural and Scientific Heritage*, 5, 99–108.
- SemaIT. (2015). Aplicações Móveis - Escolhas e Soluções. Retrieved from <https://semait.com/aplicacoes-moveis-escolhas-e-solucoes/>
- Silverstein, G. (2016). Dreams of Dalí: a virtual reality experience. Retrieved from <https://www.youtube.com/watch?v=I-Fb11PL5WA>
- Stecky-Efantis, M. (2016). 5 Easy Steps to Understanding JSON Web Tokens (JWT). Retrieved from <https://medium.com/vandium-software/5-easy-steps-to-understanding-json-web-tokens-jwt-1164c0adfcec>
- Stone, R., & Ojika, T. (2000). Virtual heritage: What next? *IEEE Multimedia*, 7(2), 73–74. <https://doi.org/10.1109/93.848434>
- Sutherland, I. E. (1968). A head-mounted three dimensional display. *Proceedings of the December 9-11, 1968, Fall Joint Computer Conference, Part I on - AFIPS '68 (Fall, Part I)*, 757. <https://doi.org/10.1145/1476589.1476686>

- Vacchetti, L., Lepetit, V., & Fua, P. (2003). Fusing online and offline information for stable 3D tracking in real-time. *Computer Vision and Pattern Recognition, 2003. Proceedings. 2003 IEEE Computer Society Conference On*, 2, II-241-8 vol.2. <https://doi.org/10.1109/CVPR.2003.1211476>
- Vallino, J. R. (1998). Interactive Augmented Reality. *Technical Report Interactive Augmented Reality*, 1–21. <https://doi.org/10.1111/j.1540-5834.2011.00603.x>
- Vlahakis, V., Ioannidis, N., Karigiannis, J., Tsotros, M., Gounaris, M., Almeida, L., ... Ioannidis, N. (2001). ARCHEOGUIDE: First results of an Augmented Reality, Mobile Computing System in Cultural Heritage Sites. *Proceedings of the 2001 Conference on Virtual Reality, Archeology, and Cultural Heritage - VAST '01*, (January), 131. <https://doi.org/10.1145/584993.585015>
- Walsh, P. (2015). What is a WebView? Retrieved from <https://developer.metacert.com/blog/what-is-a-webview/>
- Watson, B., & Luebke, D. (2005). The ultimate display: Where will all the pixels come from? In *Computer* (Vol. 38, pp. 54–61). <https://doi.org/10.1109/MC.2005.274>
- Wuest, H., Vial, F., & Stricker, D. (2005). Adaptive line tracking with multiple hypotheses for augmented reality. In *Proceedings - Fourth IEEE and ACM International Symposium on Symposium on Mixed and Augmented Reality, ISMAR 2005* (Vol. 2005, pp. 62–69). <https://doi.org/10.1109/ISMAR.2005.8>
- Younes, G., Kahil, R., Jallad, M., Asmar, D., Elhadj, I., Turkiyyah, G., & Al-Harithy, H. (2017). Virtual and augmented reality for rich interaction with cultural heritage sites: A case study from the Roman Theater at Byblos. *Digital Applications in Archaeology and Cultural Heritage*, 5, 1–9. <https://doi.org/10.1016/j.daach.2017.03.002>
- Ziamou, L. (2016). “Dreams of Dali”: The Surreal in the Virtual. Retrieved from [https://www.huffingtonpost.com/lilia-ziamou/dreams-of-dali-the-surrea\\_b\\_9168946.html](https://www.huffingtonpost.com/lilia-ziamou/dreams-of-dali-the-surrea_b_9168946.html)

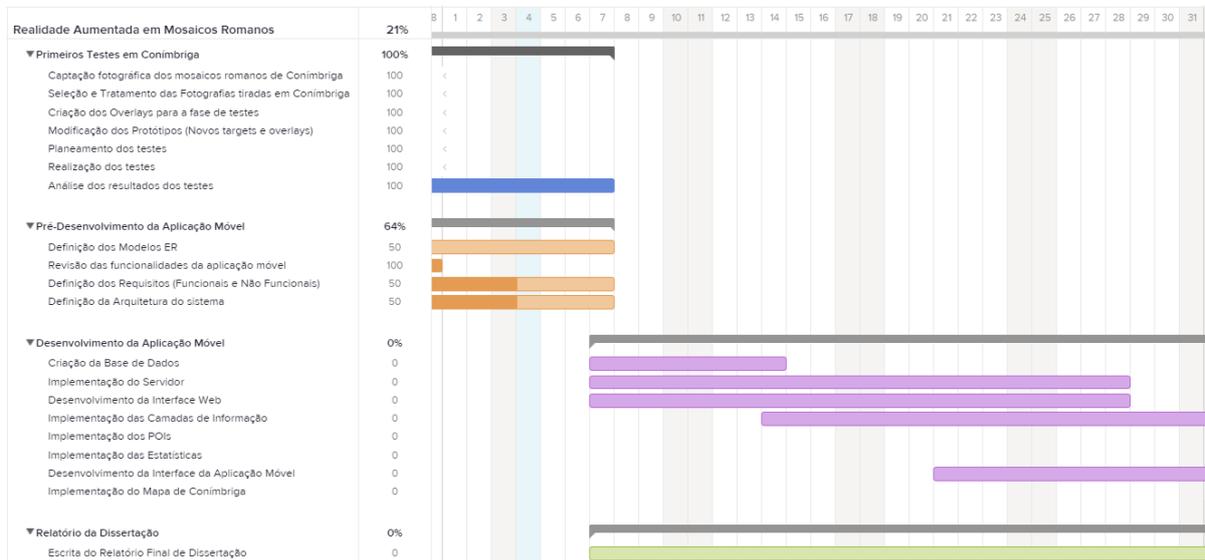
# Anexos

## A: Diagramas de Gantt do segundo semestre

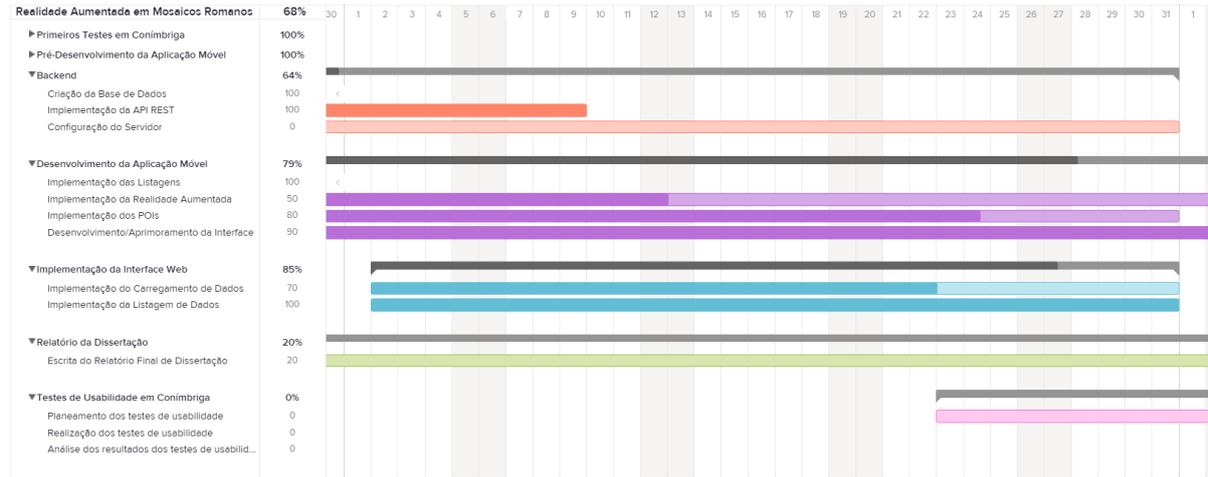
### Fevereiro



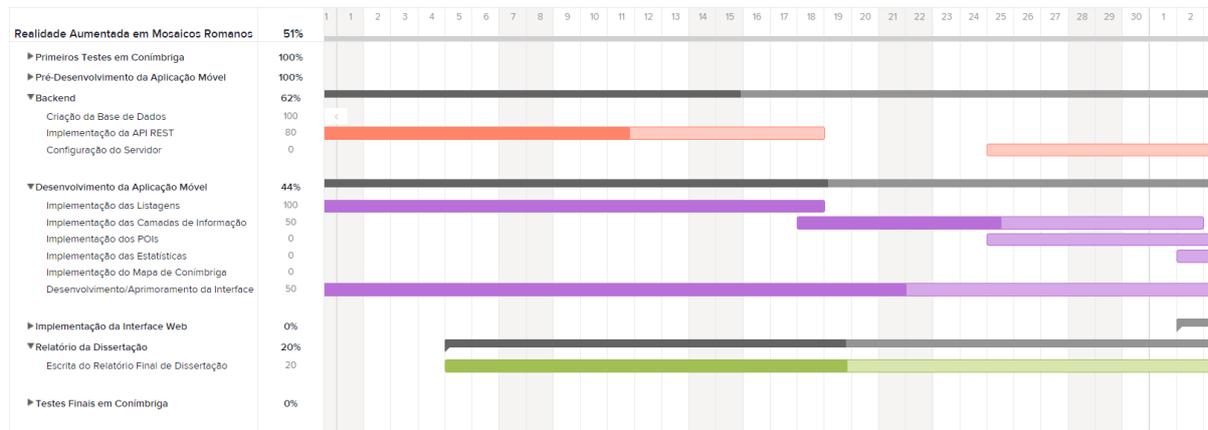
### Março



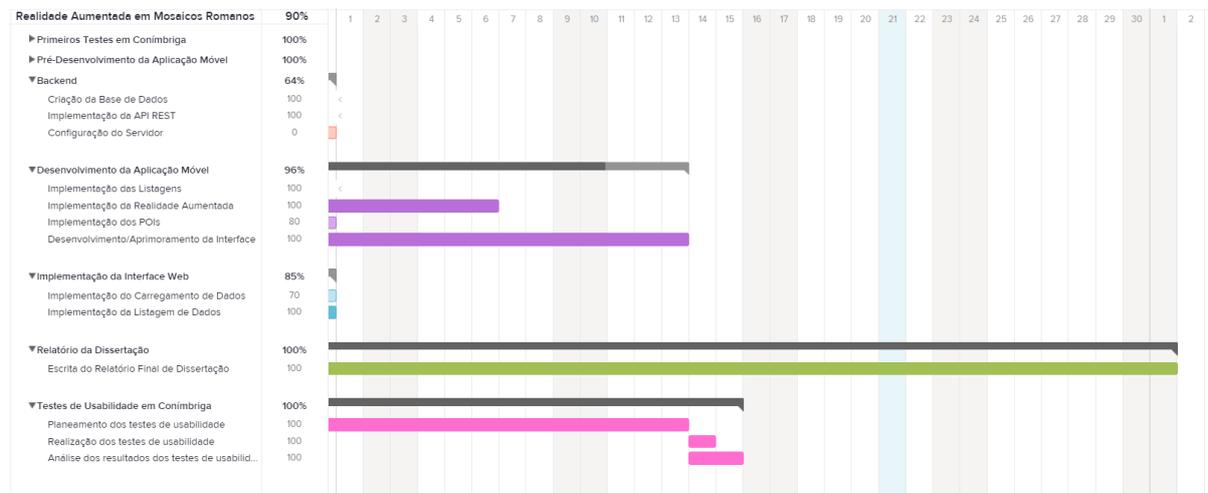
### Abril



### Maio



### Junho



## B: Listagem dos Requisitos Funcionais

A listagem dos requisitos funcionais respeita o seguinte *template*:

Componente	Notas
ID	Identificador do requisito, no seguinte formato: RF-<# do Requisito>: <Nome>
Atores	Atores intervenientes
Descrição	Descrição geral do requisito funcional
Prioridade	O nível de prioridade pode ter uma das seguintes classificações: <ul style="list-style-type: none"> <li>• <b>Alta</b></li> <li>• <b>Média</b></li> <li>• <b>Baixa</b></li> </ul>
Pré-Condições	Condições essenciais para o cumprimento do requisito funcional
Fluxo de trabalho	Listagem de ações realizadas pelo respetivo ator, de modo a cumprir o requisito funcional

Segue-se então a listagem:

<b>ID</b>	RF-1: Acesso à seção de listagem dos dados
<b>Atores</b>	Visitante das ruínas de Conímbriga
<b>Descrição</b>	O visitante pode aceder a uma seção com a listagem de mosaicos, grupos e camadas, na aplicação móvel.
<b>Prioridade</b>	Alta
<b>Pré-Condições</b>	O visitante já se deve encontrar com a aplicação móvel aberta e sem ter a experiência de RA ativa.
<b>Fluxo de trabalho</b>	<ul style="list-style-type: none"> <li>• O visitante acede à <i>tab</i> da listagem dos dados</li> </ul>

<b>ID</b>	RF-1.1: Acesso à listagem de mosaicos
<b>Atores</b>	Visitante das ruínas de Conímbriga
<b>Descrição</b>	O visitante pode aceder à listagem de mosaicos, observando apenas ícones com as respetivas imagens e os seus nomes.
<b>Prioridade</b>	Alta
<b>Pré-Condições</b>	O visitante já se deve encontrar com a aplicação móvel aberta e sem ter a experiência de RA ativa.
<b>Fluxo de trabalho</b>	<ul style="list-style-type: none"> <li>• O visitante acede à <i>tab</i> da listagem dos dados</li> </ul>

	<ul style="list-style-type: none"> <li>• O visitante escolhe a opção referente à listagem de mosaicos.</li> </ul>
--	---

<b>ID</b>	RF-1.2: Acesso à listagem de grupos
<b>Atores</b>	Visitante das ruínas de Conímbriga
<b>Descrição</b>	O visitante pode aceder à listagem de grupos, observando apenas ícones com as respetivas imagens e os seus nomes.
<b>Prioridade</b>	Alta
<b>Pré-Condições</b>	O visitante já se deve encontrar com a aplicação móvel aberta e sem ter a experiência de RA ativa.
<b>Fluxo de trabalho</b>	<ul style="list-style-type: none"> <li>• O visitante acede à <i>tab</i> da listagem dos dados</li> <li>• O visitante escolhe a opção referente à listagem de grupos.</li> </ul>

<b>ID</b>	RF-1.3: Acesso à listagem de camadas
<b>Atores</b>	Visitante das ruínas de Conímbriga
<b>Descrição</b>	O visitante pode aceder à listagem de camadas, observando apenas ícones com as respetivas imagens e os seus nomes.
<b>Prioridade</b>	Alta
<b>Pré-Condições</b>	O visitante já se deve encontrar com a aplicação móvel aberta e sem ter a experiência de RA ativa.
<b>Fluxo de trabalho</b>	<ul style="list-style-type: none"> <li>• O visitante acede à <i>tab</i> da listagem dos dados</li> <li>• O visitante escolhe a opção referente à listagem de camadas.</li> </ul>

<b>ID</b>	RF-1.1.1: Acesso às informações de um mosaico
<b>Atores</b>	Visitante das ruínas de Conímbriga
<b>Descrição</b>	O visitante pode aceder às informações sobre um mosaico, incluindo uma imagem, os seus detalhes e os seus grupos
<b>Prioridade</b>	Alta
<b>Pré-Condições</b>	O visitante já se deve encontrar com a aplicação móvel aberta e sem ter a experiência de RA ativa.
<b>Fluxo de trabalho</b>	<ul style="list-style-type: none"> <li>• O visitante acede à <i>tab</i> da listagem dos dados</li> <li>• O visitante escolhe a opção referente à listagem de mosaicos.</li> <li>• O visitante seleciona um mosaico.</li> </ul>

<b>ID</b>	RF-1.1.1.1: Acesso às informações do grupo de um mosaico
<b>Atores</b>	Visitante das ruínas de Conímbriga
<b>Descrição</b>	O visitante pode aceder às informações de um grupo diretamente do mosaico onde se encontra.
<b>Prioridade</b>	Alta

<b>Pré-Condições</b>	O visitante já se deve encontrar com a aplicação móvel aberta e sem ter a experiência de RA ativa.
<b>Fluxo de trabalho</b>	<ul style="list-style-type: none"> <li>• O visitante acede à <i>tab</i> da listagem dos dados</li> <li>• O visitante escolhe a opção referente à listagem de mosaicos.</li> <li>• O visitante seleciona um mosaico.</li> <li>• O visitante seleciona um grupo.</li> </ul>

<b>ID</b>	RF-1.2.1: Acesso às informações de um grupo
<b>Atores</b>	Visitante das ruínas de Conímbriga
<b>Descrição</b>	O visitante pode aceder às informações sobre um grupo, incluindo uma imagem, uma descrição e os seus mosaicos.
<b>Prioridade</b>	Alta
<b>Pré-Condições</b>	O visitante já se deve encontrar com a aplicação móvel aberta e sem ter a experiência de RA ativa.
<b>Fluxo de trabalho</b>	<ul style="list-style-type: none"> <li>• O visitante acede à <i>tab</i> da listagem dos dados</li> <li>• O visitante escolhe a opção referente à listagem de grupos.</li> <li>• O visitante seleciona um grupo.</li> </ul>

<b>ID</b>	RF-1.2.1.1: Acesso às informações do mosaico de um grupo
<b>Atores</b>	Visitante das ruínas de Conímbriga
<b>Descrição</b>	O visitante pode aceder às informações de um mosaico diretamente do grupo onde se encontra.
<b>Prioridade</b>	Alta
<b>Pré-Condições</b>	O visitante já se deve encontrar com a aplicação móvel aberta e sem ter a experiência de RA ativa.
<b>Fluxo de trabalho</b>	<ul style="list-style-type: none"> <li>• O visitante acede à <i>tab</i> da listagem dos dados</li> <li>• O visitante escolhe a opção referente à listagem de grupos.</li> <li>• O visitante seleciona um grupo.</li> <li>• O visitante seleciona um mosaico.</li> </ul>

<b>ID</b>	RF-1.3.1: Acesso às informações de uma camada
<b>Atores</b>	Visitante das ruínas de Conímbriga
<b>Descrição</b>	O visitante pode aceder às informações sobre uma camada, incluindo uma imagem e uma descrição.
<b>Prioridade</b>	Alta
<b>Pré-Condições</b>	O visitante já se deve encontrar com a aplicação móvel aberta e sem ter a experiência de RA ativa.
<b>Fluxo de trabalho</b>	<ul style="list-style-type: none"> <li>• O visitante acede à <i>tab</i> da listagem dos dados</li> <li>• O visitante escolhe a opção referente à listagem de camadas.</li> <li>• O visitante seleciona uma camada.</li> </ul>

<b>ID</b>	RF-2: Acesso ao tutorial
<b>Atores</b>	Visitante das ruínas de Conímbriga
<b>Descrição</b>	O visitante pode aceder a um tutorial acerca das funcionalidades da aplicação móvel.
<b>Prioridade</b>	Baixa
<b>Pré-Condições</b>	O visitante já se deve encontrar com a aplicação móvel aberta e sem ter a experiência de RA ativa.
<b>Fluxo de trabalho</b>	<ul style="list-style-type: none"> <li>• O visitante acede à <i>tab</i> do tutorial</li> </ul>

<b>ID</b>	RF-3: Início da experiência de RA
<b>Atores</b>	Visitante das ruínas de Conímbriga
<b>Descrição</b>	O visitante pode iniciar uma experiência de RA.
<b>Prioridade</b>	Alta
<b>Pré-Condições</b>	O visitante já se deve encontrar com a aplicação móvel aberta e sem ter a experiência de RA ativa.
<b>Fluxo de trabalho</b>	<ul style="list-style-type: none"> <li>• O visitante acede à <i>tab</i> do tutorial</li> <li>• O visitante clica no botão encontrado nessa <i>tab</i>, para dar início à experiência de RA.</li> </ul>

<b>ID</b>	RF-3.1: Visualização dos Pontos de Interesse
<b>Atores</b>	Visitante das ruínas de Conímbriga
<b>Descrição</b>	O visitante pode visualizar os Pontos de Interesse distribuídos pelos mosaicos romanos.
<b>Prioridade</b>	Média
<b>Pré-Condições</b>	O visitante já se deve encontrar com a aplicação móvel aberta e ter a experiência de RA ativa.
<b>Fluxo de trabalho</b>	<ul style="list-style-type: none"> <li>• O visitante movimenta o seu <i>smartphone</i>, até que os Pontos de Interesse apareçam no campo de visão da câmara.</li> </ul>

<b>ID</b>	RF-3.2: Visualização dos Pontos de Interesse distintos
<b>Atores</b>	Visitante das ruínas de Conímbriga
<b>Descrição</b>	O visitante pode visualizar os Pontos de Interesse que sejam distintos dos restantes, de forma a perceber que os mosaicos correspondentes contêm experiências de RA.
<b>Prioridade</b>	Média
<b>Pré-Condições</b>	O visitante já se deve encontrar com a aplicação móvel aberta e ter a experiência de RA ativa.
<b>Fluxo de trabalho</b>	<ul style="list-style-type: none"> <li>• O visitante movimenta o seu <i>smartphone</i>, até que os Pontos de Interesse distintos apareçam no campo de visão da câmara.</li> </ul>

<b>ID</b>	RF-3.3: Visualização de informações sobre o mosaico através do Ponto de Interesse
<b>Atores</b>	Visitante das ruínas de Conímbriga
<b>Descrição</b>	O visitante pode visualizar as informações acerca de um mosaico, ao clicar no respetivo Ponto de Interesse.
<b>Prioridade</b>	Média
<b>Pré-Condições</b>	O visitante já se deve encontrar com a aplicação móvel aberta e ter a experiência de RA ativa.
<b>Fluxo de trabalho</b>	<ul style="list-style-type: none"> <li>• O visitante movimenta o seu <i>smartphone</i>, até que os Pontos de Interesse apareçam no campo de visão da câmara.</li> <li>• O visitante seleciona o Ponto de Interesse.</li> </ul>

<b>ID</b>	RF-3.4: Início da experiência de RA de um mosaico romano
<b>Atores</b>	Visitante das ruínas de Conímbriga
<b>Descrição</b>	O visitante pode iniciar uma experiência de RA de um mosaico romano.
<b>Prioridade</b>	Alta
<b>Pré-Condições</b>	O visitante já se deve encontrar com a aplicação móvel aberta e ter a experiência de RA ativa.
<b>Fluxo de trabalho</b>	<ul style="list-style-type: none"> <li>• O visitante aponta a câmara do <i>smartphone</i> para um mosaico que contenha uma experiência de RA. Caso a experiência não seja iniciada de imediato, o visitante terá que movimentar subtilmente o <i>smartphone</i>, sem que o mosaico saia da mira da câmara.</li> </ul>

<b>ID</b>	RF-3.4.1: Alternar entre <i>overlays</i> de um mosaico
<b>Atores</b>	Visitante das ruínas de Conímbriga
<b>Descrição</b>	O visitante pode alternar entre os <i>overlays</i> existentes no mosaico atual, em tempo real.
<b>Prioridade</b>	Alta
<b>Pré-Condições</b>	O visitante já se deve encontrar com a aplicação móvel aberta e ter a experiência de RA ativa. Além disso, é necessário que a experiência de RA do mosaico atual esteja ativa e que a camada de informação ativa/selecionada contenha <i>overlays</i> em comum com o mosaico atual.
<b>Fluxo de trabalho</b>	<ul style="list-style-type: none"> <li>• O visitante, com a câmara apontada para o mosaico e com a sua experiência de RA ativa, clica nas setas (esquerda ou direita) encontradas a meio do ecrã, para alternar entre os <i>overlays</i> do mosaico.</li> </ul>

<b>ID</b>	RF-3.4.2: Alternar entre camadas de informação
<b>Atores</b>	Visitante das ruínas de Conímbriga
<b>Descrição</b>	O visitante pode alternar entre as camadas de informação existentes, durante uma experiência de RA de um mosaico.
<b>Prioridade</b>	Alta

<b>Pré-Condições</b>	O visitante já se deve encontrar com a aplicação móvel aberta e ter a experiência de RA ativa. Além disso, é necessário que também esteja ativa a experiência de RA de um mosaico.
<b>Fluxo de trabalho</b>	<ul style="list-style-type: none"> <li>O visitante, com a câmara apontada para um mosaico e com a sua experiência de RA ativa, clica nas setas (esquerda ou direita) encontradas no fundo do ecrã, acima do nome da camada atual, para alternar entre as camadas.</li> </ul>

<b>ID</b>	RF-3.4.3: Visualizar o nome da camada atual
<b>Atores</b>	Visitante das ruínas de Conímbriga
<b>Descrição</b>	O visitante pode observar qual a camada ativa no momento.
<b>Prioridade</b>	Alta
<b>Pré-Condições</b>	O visitante já se deve encontrar com a aplicação móvel aberta e ter a experiência de RA ativa. Além disso, é necessário que também esteja ativa a experiência de RA de um mosaico.
<b>Fluxo de trabalho</b>	<ul style="list-style-type: none"> <li>O visitante, com a câmara apontada para um mosaico e com a sua experiência de RA ativa, pode observar o nome da camada ativa no fundo do ecrã.</li> </ul>

<b>ID</b>	RF-3.4.4: Mensagens de alerta
<b>Atores</b>	Visitante das ruínas de Conímbriga
<b>Descrição</b>	O visitante observa mensagens de alerta sempre que acontecem os seguintes casos: <ol style="list-style-type: none"> <li>Não existem <i>overlays</i> em comum entre a camada ativa e o mosaico atual</li> <li>Não existem mais <i>overlays</i> (durante a alternância de <i>overlays</i>)</li> <li>Não existem mais camadas (durante a alternância de camadas)</li> </ol>
<b>Prioridade</b>	Alta
<b>Pré-Condições</b>	O visitante já se deve encontrar com a aplicação móvel aberta e ter a experiência de RA ativa. Além disso, é necessário que também esteja ativa a experiência de RA de um mosaico.
<b>Fluxo de trabalho</b>	<ul style="list-style-type: none"> <li>O visitante, com a câmara apontada para um mosaico e com a sua experiência de RA ativa: <ol style="list-style-type: none"> <li>Encontra-se com uma camada ativa que não contém <i>overlays</i> em comum com o mosaico; Alterna a camada ativa e passa para uma que não contém <i>overlays</i> em comum com o mosaico</li> <li>Carrega nas setas de alternância de <i>overlays</i> (esquerda ou direita) até não haver mais <i>overlays</i></li> <li>Carrega nas setas de alternância de camadas (esquerda ou direita) até não haver mais camadas</li> </ol> </li> </ul>

<b>ID</b>	RF-4: Atualização dos dados da aplicação
<b>Atores</b>	Visitante das ruínas de Conímbriga
<b>Descrição</b>	O visitante pode atualizar os dados da aplicação móvel

<b>Prioridade</b>	Baixa
<b>Pré-Condições</b>	O visitante já se deve encontrar com a aplicação móvel aberta e não ter a experiência de RA ativa. Além disso, o visitante deve estar ligado à Internet
<b>Fluxo de trabalho</b>	<ul style="list-style-type: none"> <li>• O visitante acede à <i>tab</i> inicial</li> <li>• O visitante deve seleccionar a opção de atualização de dados</li> </ul>

<b>ID</b>	RF-5: Login na interface web
<b>Atores</b>	Gestor de conteúdos
<b>Descrição</b>	O gestor de conteúdos pode aceder à interface web, efetuando o <i>login</i> .
<b>Prioridade</b>	Alta
<b>Pré-Condições</b>	Ligação à Internet.
<b>Fluxo de trabalho</b>	<ul style="list-style-type: none"> <li>• O gestor de conteúdos acede à interface web</li> <li>• O gestor de conteúdos introduz o seu email e password e submete o pedido de <i>login</i></li> </ul>

<b>ID</b>	RF-6: Aceder ao menu principal
<b>Atores</b>	Gestor de conteúdos
<b>Descrição</b>	O gestor de conteúdos pode aceder ao menu principal da interface web.
<b>Prioridade</b>	Alta
<b>Pré-Condições</b>	Ligação à Internet e <i>login</i> efetuado.
<b>Fluxo de trabalho</b>	<ul style="list-style-type: none"> <li>• O gestor de conteúdos acede à interface web</li> </ul>

<b>ID</b>	RF-6.1: Aceder à listagem de mosaicos
<b>Atores</b>	Gestor de conteúdos
<b>Descrição</b>	O gestor de conteúdos pode aceder à listagem de mosaicos.
<b>Prioridade</b>	Alta
<b>Pré-Condições</b>	Ligação à Internet e <i>login</i> efetuado.
<b>Fluxo de trabalho</b>	<ul style="list-style-type: none"> <li>• O gestor de conteúdos acede à interface web</li> <li>• O gestor de conteúdos acede ao menu principal</li> <li>• O gestor de conteúdos clica no botão referente às operações sobre mosaicos</li> <li>• O gestor de conteúdos seleciona a opção de listagem de mosaicos</li> </ul>

<b>ID</b>	RF-6.2: Aceder à listagem de grupos
<b>Atores</b>	Gestor de conteúdos
<b>Descrição</b>	O gestor de conteúdos pode aceder à listagem de grupos.
<b>Prioridade</b>	Alta

<b>Pré-Condições</b>	Ligação à Internet e <i>login</i> efetuado.
<b>Fluxo de trabalho</b>	<ul style="list-style-type: none"> <li>• O gestor de conteúdos acede à interface web</li> <li>• O gestor de conteúdos acede ao menu principal</li> <li>• O gestor de conteúdos clica no botão referente às operações sobre grupos</li> <li>• O gestor de conteúdos seleciona a opção de listagem de grupos</li> </ul>

<b>ID</b>	RF-6.3: Aceder à listagem de camadas
<b>Atores</b>	Gestor de conteúdos
<b>Descrição</b>	O gestor de conteúdos pode aceder à listagem de camadas.
<b>Prioridade</b>	Alta
<b>Pré-Condições</b>	Ligação à Internet e <i>login</i> efetuado.
<b>Fluxo de trabalho</b>	<ul style="list-style-type: none"> <li>• O gestor de conteúdos acede à interface web</li> <li>• O gestor de conteúdos acede ao menu principal</li> <li>• O gestor de conteúdos clica no botão referente às operações sobre camadas</li> <li>• O gestor de conteúdos seleciona a opção de listagem de camadas</li> </ul>

<b>ID</b>	RF-6.4: Aceder à listagem de <i>targets</i>
<b>Atores</b>	Gestor de conteúdos
<b>Descrição</b>	O gestor de conteúdos pode aceder à listagem de <i>targets</i> .
<b>Prioridade</b>	Alta
<b>Pré-Condições</b>	Ligação à Internet e <i>login</i> efetuado.
<b>Fluxo de trabalho</b>	<ul style="list-style-type: none"> <li>• O gestor de conteúdos acede à interface web</li> <li>• O gestor de conteúdos acede ao menu principal</li> <li>• O gestor de conteúdos clica no botão referente às operações sobre <i>targets</i></li> <li>• O gestor de conteúdos seleciona a opção de listagem de <i>targets</i></li> </ul>

<b>ID</b>	RF-6.5: Aceder à listagem de <i>overlays</i>
<b>Atores</b>	Gestor de conteúdos
<b>Descrição</b>	O gestor de conteúdos pode aceder à listagem de <i>overlays</i> .
<b>Prioridade</b>	Alta
<b>Pré-Condições</b>	Ligação à Internet e <i>login</i> efetuado.
<b>Fluxo de trabalho</b>	<ul style="list-style-type: none"> <li>• O gestor de conteúdos acede à interface web</li> <li>• O gestor de conteúdos acede ao menu principal</li> <li>• O gestor de conteúdos clica no botão referente às operações sobre <i>overlays</i></li> <li>• O gestor de conteúdos seleciona a opção de listagem de <i>overlays</i></li> </ul>

<b>ID</b>	RF-6.6: Editar um mosaico
<b>Atores</b>	Gestor de conteúdos

<b>Descrição</b>	O gestor de conteúdos pode editar os dados de um mosaico específico ou eliminá-lo.
<b>Prioridade</b>	Alta
<b>Pré-Condições</b>	Ligação à Internet e <i>login</i> efetuado.
<b>Fluxo de trabalho</b>	<ul style="list-style-type: none"> <li>• O gestor de conteúdos acede à interface web</li> <li>• O gestor de conteúdos acede ao menu principal</li> <li>• O gestor de conteúdos clica no botão referente às operações sobre mosaicos</li> <li>• O gestor de conteúdos seleciona a opção de editar mosaicos</li> <li>• O gestor de conteúdos insere o ID do mosaico que pretende editar e submete o pedido</li> <li>• O gestor de conteúdos edita os dados que pretender do mosaico selecionado ou elimina-o.</li> </ul>

<b>ID</b>	RF-6.7: Editar um grupo
<b>Atores</b>	Gestor de conteúdos
<b>Descrição</b>	O gestor de conteúdos pode editar os dados de um grupo específico ou eliminá-lo.
<b>Prioridade</b>	Alta
<b>Pré-Condições</b>	Ligação à Internet e <i>login</i> efetuado.
<b>Fluxo de trabalho</b>	<ul style="list-style-type: none"> <li>• O gestor de conteúdos acede à interface web</li> <li>• O gestor de conteúdos acede ao menu principal</li> <li>• O gestor de conteúdos clica no botão referente às operações sobre grupos</li> <li>• O gestor de conteúdos seleciona a opção de editar grupos</li> <li>• O gestor de conteúdos insere o ID do grupo que pretende editar e submete o pedido</li> <li>• O gestor de conteúdos edita os dados que pretender do grupo selecionado ou elimina-o.</li> </ul>

<b>ID</b>	RF-6.8: Editar uma camada
<b>Atores</b>	Gestor de conteúdos
<b>Descrição</b>	O gestor de conteúdos pode editar os dados de uma camada específica ou eliminá-la.
<b>Prioridade</b>	Alta
<b>Pré-Condições</b>	Ligação à Internet e <i>login</i> efetuado.
<b>Fluxo de trabalho</b>	<ul style="list-style-type: none"> <li>• O gestor de conteúdos acede à interface web</li> <li>• O gestor de conteúdos acede ao menu principal</li> <li>• O gestor de conteúdos clica no botão referente às operações sobre camadas</li> <li>• O gestor de conteúdos seleciona a opção de editar camadas</li> <li>• O gestor de conteúdos insere o ID da camada que pretende editar e submete o pedido</li> <li>• O gestor de conteúdos edita os dados que pretender da camada selecionada ou elimina-a.</li> </ul>

<b>ID</b>	RF-6.9: Editar um <i>overlay</i>
<b>Atores</b>	Gestor de conteúdos

<b>Descrição</b>	O gestor de conteúdos pode editar os dados de um <i>overlay</i> específico ou eliminá-lo.
<b>Prioridade</b>	Alta
<b>Pré-Condições</b>	Ligação à Internet e <i>login</i> efetuado.
<b>Fluxo de trabalho</b>	<ul style="list-style-type: none"> <li>• O gestor de conteúdos acede à interface web</li> <li>• O gestor de conteúdos acede ao menu principal</li> <li>• O gestor de conteúdos clica no botão referente às operações sobre <i>overlays</i></li> <li>• O gestor de conteúdos seleciona a opção de editar <i>overlays</i></li> <li>• O gestor de conteúdos insere o ID do <i>overlay</i> que pretende editar e submete o pedido</li> <li>• O gestor de conteúdos edita os dados que pretender do <i>overlay</i> selecionado ou elimina-o.</li> </ul>

<b>ID</b>	RF-6.10: Editar um <i>target</i>
<b>Atores</b>	Gestor de conteúdos
<b>Descrição</b>	O gestor de conteúdos pode editar os dados de um <i>target</i> específico ou eliminá-lo.
<b>Prioridade</b>	Alta
<b>Pré-Condições</b>	Ligação à Internet e <i>login</i> efetuado.
<b>Fluxo de trabalho</b>	<ul style="list-style-type: none"> <li>• O gestor de conteúdos acede à interface web</li> <li>• O gestor de conteúdos acede ao menu principal</li> <li>• O gestor de conteúdos clica no botão referente às operações sobre <i>targets</i></li> <li>• O gestor de conteúdos seleciona a opção de editar <i>targets</i></li> <li>• O gestor de conteúdos insere o ID do <i>target</i> que pretende editar e submete o pedido</li> <li>• O gestor de conteúdos edita os dados que pretender do <i>target</i> selecionado ou elimina-o.</li> </ul>

<b>ID</b>	RF-6.11: Criar um mosaico
<b>Atores</b>	Gestor de conteúdos
<b>Descrição</b>	O gestor de conteúdos pode criar um novo mosaico.
<b>Prioridade</b>	Alta
<b>Pré-Condições</b>	Ligação à Internet e <i>login</i> efetuado.
<b>Fluxo de trabalho</b>	<ul style="list-style-type: none"> <li>• O gestor de conteúdos acede à interface web</li> <li>• O gestor de conteúdos acede ao menu principal</li> <li>• O gestor de conteúdos clica no botão referente às operações sobre mosaicos</li> <li>• O gestor de conteúdos seleciona a opção de criar mosaicos</li> <li>• O gestor de conteúdos preenche o formulário e submete o pedido de criação do mosaico</li> </ul>

<b>ID</b>	RF-6.12: Criar um grupo
<b>Atores</b>	Gestor de conteúdos

<b>Descrição</b>	O gestor de conteúdos pode criar um novo grupo.
<b>Prioridade</b>	Alta
<b>Pré-Condições</b>	Ligação à Internet e <i>login</i> efetuado.
<b>Fluxo de trabalho</b>	<ul style="list-style-type: none"> <li>• O gestor de conteúdos acede à interface web</li> <li>• O gestor de conteúdos acede ao menu principal</li> <li>• O gestor de conteúdos clica no botão referente às operações sobre grupos</li> <li>• O gestor de conteúdos seleciona a opção de criar grupos</li> <li>• O gestor de conteúdos preenche o formulário e submete o pedido de criação do grupo</li> </ul>

<b>ID</b>	RF-6.13: Criar uma camada
<b>Atores</b>	Gestor de conteúdos
<b>Descrição</b>	O gestor de conteúdos pode criar uma nova camada.
<b>Prioridade</b>	Alta
<b>Pré-Condições</b>	Ligação à Internet e <i>login</i> efetuado.
<b>Fluxo de trabalho</b>	<ul style="list-style-type: none"> <li>• O gestor de conteúdos acede à interface web</li> <li>• O gestor de conteúdos acede ao menu principal</li> <li>• O gestor de conteúdos clica no botão referente às operações sobre camadas</li> <li>• O gestor de conteúdos seleciona a opção de criar camadas</li> <li>• O gestor de conteúdos preenche o formulário e submete o pedido de criação da camada</li> </ul>

<b>ID</b>	RF-6.11: Criar um <i>overlay</i>
<b>Atores</b>	Gestor de conteúdos
<b>Descrição</b>	O gestor de conteúdos pode criar um novo <i>overlay</i> .
<b>Prioridade</b>	Alta
<b>Pré-Condições</b>	Ligação à Internet e <i>login</i> efetuado.
<b>Fluxo de trabalho</b>	<ul style="list-style-type: none"> <li>• O gestor de conteúdos acede à interface web</li> <li>• O gestor de conteúdos acede ao menu principal</li> <li>• O gestor de conteúdos clica no botão referente às operações sobre <i>overlays</i></li> <li>• O gestor de conteúdos seleciona a opção de criar <i>overlays</i></li> <li>• O gestor de conteúdos preenche o formulário e submete o pedido de criação do <i>overlay</i></li> </ul>

<b>ID</b>	RF-6.12: Criar um <i>target</i>
<b>Atores</b>	Gestor de conteúdos
<b>Descrição</b>	O gestor de conteúdos pode criar um novo <i>target</i> .
<b>Prioridade</b>	Alta

<b>Pré-Condições</b>	Ligação à Internet e <i>login</i> efetuado.
<b>Fluxo de trabalho</b>	<ul style="list-style-type: none"> <li>• O gestor de conteúdos acede à interface web</li> <li>• O gestor de conteúdos acede ao menu principal</li> <li>• O gestor de conteúdos clica no botão referente às operações sobre <i>targets</i></li> <li>• O gestor de conteúdos seleciona a opção de criar <i>targets</i></li> <li>• O gestor de conteúdos preenche o formulário e submete o pedido de criação do <i>target</i></li> </ul>

<b>ID</b>	RF-7: Efetuar <i>logout</i>
<b>Atores</b>	Gestor de conteúdos
<b>Descrição</b>	O gestor de conteúdos pode terminar a sessão atual na interface web, fazendo o <i>logout</i> .
<b>Prioridade</b>	Alta
<b>Pré-Condições</b>	Ligação à Internet e <i>login</i> efetuado.
<b>Fluxo de trabalho</b>	<ul style="list-style-type: none"> <li>• O gestor de conteúdos acede à interface web</li> <li>• O gestor de conteúdos clica no botão de <i>logout</i> para terminar a sessão.</li> </ul>

## C: Formulários dos testes de usabilidade

### C1: Questões pré-testes

Participante #1

Sexo (M/F): **M**

Idade: **68**

1 - Já ouviu falar das ruínas de Conímbriga? (**Sim/Não**)

**Sim**

1.1. - Se **sim**, já alguma vez visitou o espaço? (**Sim/Não**)

**Sim**

1.1.1 - Se **sim**, tem uma noção considerável da localização dos edifícios/mosaicos? (**Sim / Não**)

**Não**

1.1.2 - Se **não**, estaria disposto a visitar este espaço? (**Sim/Não**)

2 - Conhece o conceito de Património Cultural?

**Sim**

2.1. - Se **sim**, o quão importante é para si? (**Pouco/Suficiente/Bastante**)

**Bastante**

3 - Sabe/Tem noção do que é a Realidade Aumentada? (**Sim / Não**)

**Não**

3.1 - Se **sim**, já alguma vez utilizou esta tecnologia? (**Sim / Não**)

**Não**

3.1.1 - Se **sim**, descreva brevemente a sua experiência.

3.1.2 - Se **sim**: (**Responder apenas se respondeu Sim à pergunta 2**)

3.1.2.1 - Acha que é uma forma adequada de preservação do Património Cultural? (**Sim/Não**.)

**Dê um motivo))**

3.1.2.2 - Acha que é uma forma atrativa de visitar locais de Património Cultural? (**Sim/Não**.) **Dê**

**um motivo)**

3.2 - Se **não**, está curioso em experimentá-la? (**Sim/Não**)

**Sim**

4 - O quão facilmente acha que se adapta a aplicações móveis novas? (**Pouco/Suficiente/Bastante**)

**Suficiente**

Participante #2

Sexo (M/F): F

Idade: 24

1 - Já ouviu falar das ruínas de Conímbriga? **(Sim/Não)**

**Sim**

1.1. - Se **sim**, já alguma vez visitou o espaço? **(Sim/Não)**

**Sim**

1.1.1 - Se **sim**, tem uma noção considerável da localização dos edifícios/mosaicos? **(Sim / Não)**

**Não**

1.1.2 - Se **não**, estaria disposto a visitar este espaço? **(Sim/Não)**

2 - Conhece o conceito de Património Cultural?

**Sim**

2.1. - Se **sim**, o quão importante é para si? **(Pouco/Suficiente/Bastante)**

**Bastante**

3 - Sabe/Tem noção do que é a Realidade Aumentada? **(Sim / Não)**

**Sim**

3.1 - Se **sim**, já alguma vez utilizou esta tecnologia? **(Sim / Não)**

**Não**

3.1.1 - Se **sim**, descreva brevemente a sua experiência.

3.1.2 - Se **sim**: **(Responder apenas se respondeu Sim à pergunta 2)**

3.1.2.1 - Acha que é uma forma adequada de preservação do Património Cultural? **(Sim/Não. Dê um motivo))**

**Dê um motivo))**

3.1.2.2 - Acha que é uma forma atrativa de visitar locais de Património Cultural? **(Sim/Não. Dê um motivo)**

**um motivo)**

3.2 - Se **não**, está curioso em experimentá-la? **(Sim/Não)**

**Sim**

4 - O quão facilmente acha que se adapta a aplicações móveis novas? **(Pouco/Suficiente/Bastante)**

**Bastante**

Participante #3

Sexo (M/F): **M**

Idade: **27**

1 - Já ouviu falar das ruínas de Conímbriga? (**Sim/Não**)

**Sim.**

1.1. - Se **sim**, já alguma vez visitou o espaço? (**Sim/Não**)

**Sim.**

1.1.1 - Se **sim**, tem uma noção considerável da localização dos edifícios/mosaicos? (**Sim / Não**)

**Sim, já visitei o espaço, mas o mesmo já sofreu alterações depois da minha última visita, pelo que o meu mapa mental pode já não corresponder ao estado actual.**

1.1.2 - Se **não**, estaria disposto a visitar este espaço? (**Sim/Não**)

2 - Conhece o conceito de Património Cultural?

**Sim.**

2.1. - Se **sim**, o quão importante é para si? (**Pouco/Suficiente/Bastante**)

**Bastante.**

3 - Sabe/Tem noção do que é a Realidade Aumentada? (**Sim / Não**)

**Sim.**

3.1 - Se **sim**, já alguma vez utilizou esta tecnologia? (**Sim / Não**)

**Sim.**

3.1.1 - Se **sim**, descreva brevemente a sua experiência.

**Utilizei a tecnologia como meio de jogar videojogos, como forma criativa para fotografia e gravação de vídeo.**

3.1.2 - Se **sim**: (**Responder apenas se respondeu Sim à pergunta 2**)

3.1.2.1 - Acha que é uma forma adequada de preservação do Património Cultural? (**Sim/Não.**

**Dê um motivo)).**

**Sim. O facto dos artefactos com a designação de Património Cultural implicar que quer pela antiguidade, unicidade, fragilidade entre outros factores, um tipo de interacção mais limitado por parte do visitante torna este tipo de tecnologia atractiva para explorar não só os objectos na sua forma actual, como dar uso criativo na forma de ser apresentado e meios de exploração de maior dinâmica.**

3.1.2.2 - Acha que é uma forma atrativa de visitar locais de Património Cultural? (**Sim/Não. Dê**

**um motivo).**

**Sim. O potencial criativo e interactivo que os elementos virtuais proporcionam é propício a que se expanda a forma como o visitante percepção e conhece o conteúdo museológico. Numa fase mais literal podem ser exploradas réplicas e reconstruções, num âmbito mais expandido o uso criativo como forma de exploração artística do tipo de interacção que se tem.**

3.2 - Se **não**, está curioso em experimentá-la? (**Sim/Não**)

4 - O quão rápido achas que te adaptas a utilizar aplicações móveis novas? (**Pouco/Suficiente/Bastante**).

**Bastante.**

Participante #4

Sexo (M/F): **M**

Idade: **58**

1 - Já ouviu falar das ruínas de Conímbriga? **(Sim/Não)**

**Sim**

1.1. - Se **sim**, já alguma vez visitou o espaço? **(Sim/Não)**

**Sim**

1.1.1 - Se **sim**, tem uma noção considerável da localização dos edifícios/mosaicos? **(Sim / Não)**

**Sim**

1.1.2 - Se **não**, estaria disposto a visitar este espaço? **(Sim/Não)**

2 - Conhece o conceito de Património Cultural?

**Sim**

2.1. - Se **sim**, o quanto importante é para si? **(Pouco/Suficiente/Bastante)**

**Bastante**

3 - Sabe/Tem noção do que é a Realidade Aumentada? **(Sim / Não)**

**Sim**

3.1 - Se **sim**, já alguma vez utilizou esta tecnologia? **(Sim / Não)**

**Não**

3.1.1 - Se **sim**, descreva brevemente a sua experiência.

3.1.2 - Se **sim**: **(Responder apenas se respondeu Sim à pergunta 2)**

3.1.2.1 - Acha que é uma forma adequada de preservação do Património Cultural? **(Sim/Não)**.

**Dê um motivo))**

3.1.2.2 - Acha que é uma forma atrativa de visitar locais de Património Cultural? **(Sim/Não. Dê**

**um motivo)**

3.2 - Se **não**, está curioso em experimentá-la? **(Sim/Não)**

**Sim**

4 - O quanto facilmente acha que se adapta a aplicações móveis novas? **(Pouco/Suficiente/Bastante)**

**Suficiente**

## C2: Questões pós-testes

Nome: Participante #1

Idade: 65

Sexo (M/F): M

1 - Quais foram as maiores dificuldades sentidas?

NÃO NOTEI DIFICULDADES

2 - Quais foram as tarefas mais difíceis?

PRATICAMENTE NENHUMAS

3 - Quais foram as tarefas mais fáceis?

TODAS

4 - Que tipo de camadas gostou mais?

TODAS SÃO INTERESSANTES

5 - Que tipo de camadas (não existentes ainda) sugere?

NENHUMAS

6 - De entre os conceitos utilizados na aplicação (Camadas, Overlays, Targets, Pontos de Interesse) quais é que consegue associar àquilo que presenciou com a aplicação?

TODAS ELAS ESTÃO COMTEMPORÂNEAS

7 - Há alguma informação (não existente ainda) sobre os mosaicos que gostasse de ver?

NÃO

8 - Que problemas identificou na interface?

NENHUM

9 - Mais alguma sugestão?

NÃO

Nome: Participante #2

Idade: 24

Sexo (M/F): F

1 - Quais foram as maiores dificuldades sentidas?

Saber onde clicar, no entanto, com a prática torna-se intuitivo.

2 - Quais foram as tarefas mais difíceis?

Conseguir ler o texto no final.

3 - Quais foram as tarefas mais fáceis?

Emenbrar as imagens.

4 - Que tipo de camadas gostou mais?

Substituição.

5 - Que tipo de camadas (não existentes ainda) sugere?

6 - De entre os conceitos utilizados na aplicação (Camadas, Overlays, Targets, Pontos de Interesse) quais é que consegue associar àquilo que presenciou com a aplicação?

Camadas e Pontos de Interesse.

7 - Há alguma informação (não existente ainda) sobre os mosaicos que gostasse de ver?

8 - Que problemas identificou na interface?

Nenhum.

9 - Mais alguma sugestão?

Nome: Participante #3

Idade: 27

Sexo (M/F): M

1 - Quais foram as maiores dificuldades sentidas?

Conseguir encontrar a listagem de mosaicos e navegação nessa parte do teste. A navegação por camadas não é a mais intuitiva no momento de usar o reconhecimento de câmara.

2 - Quais foram as tarefas mais difíceis?

Encontrar a listagem de mosaicos e navegar entre camadas e nas diferentes imagens de cada camada.

3 - Quais foram as tarefas mais fáceis?

Efectuar a procura de elementos com recurso à câmara e navegação nos detalhes dos mosaicos.

4 - Que tipo de camadas gostou mais?

Camada de restauro.

5 - Que tipo de camadas (não existentes ainda) sugere?

Camadas animadas ou que estimularem algum tipo de criatividade por parte do visitante (ex: ~~animar~~ elementos com animação ou áudio).

6 - De entre os conceitos utilizados na aplicação (Camadas, Overlays, Targets, Pontos de Interesse) quais é que consegue associar àquilo que presenciou com a aplicação?

Camadas, pontos de interesse.

7 - Há alguma informação (não existente ainda) sobre os mosaicos que gostasse de ver?

Significados ou contextos das formas para cada mosaico em específico.

8 - Que problemas identificou na interface?

A navegação com câmara estar ~~mais~~ separada dos menus de mosaico e ajuda. A navegação de camadas não é a mais intuitiva (o sistema cromático é insuficiente para saber quantos elementos de cada camada ainda não foram descobertos).

9 - Mais alguma sugestão?

O acesso aos menus de listagem de mosaico e ajuda serem integrados na interface de navegação da câmara, para que não seja necessário abandonar ~~isso~~ a procura com câmara e sair por acidente da aplicação.

Nome: Participante #4

Idade: 58

Sexo (M/F): MASCULINO

1 - Quais foram as maiores dificuldades sentidas?

ENTENDER ALGUNS DOS PASSOS DOS ACESSOS AOS CONTEÚDOS

2 - Quais foram as tarefas mais difíceis?

A ADAPTAÇÃO AOS COMANDOS EM ALGUNS MOMENTOS

3 - Quais foram as tarefas mais fáceis?

INICIAR A APLICAÇÃO E ACEDER AOS PRIMEIROS CONTEÚDOS.

4 - Que tipo de camadas gostou mais?

AS QUE ~~REVELEM~~ REVELAM AS FORMAS INTERNAS NOS MOSAICOS

5 - Que tipo de camadas (não existentes ainda) sugere?

OBSERVAR LINHAS E OUTRAS FORMAS QUE AJUDEM A ENTENDER CADA MOSAICO.

6 - De entre os conceitos utilizados na aplicação (Camadas, Overlays, Targets, Pontos de Interesse) quais é que consegue associar àquilo que presenciou com a aplicação?

CAMADAS, PONTOS DE INTERESSE.

7 - Há alguma informação (não existente ainda) sobre os mosaicos que gostasse de ver?

VOCABULÁRIO DE FORMAS, CONTEÚDOS ESPECÍFICOS DE CADA MOSAICO.

8 - Que problemas identificou na interface?

É INTERESSANTE MAS GOSTARIA DE PODER EVITAR UTILIZAR OS COMANDOS DO PRÓPRIO TELEMÓVEL, SE POSSÍVEL OU DESEJÁVEL FOR.

9 - Mais alguma sugestão?

FOI UMA EXPERIÊNCIA MUITO INTERESSANTE. PESSOALMENTE, GOSTARIA QUE A APLICAÇÃO NOS AJUDASSE A DESCOBRIR CADA MOSAICO PELO SEU CONTEÚDO PRÓPRIO.

## **D: Vídeos dos testes realizados**

Neste anexo encontram-se alguns exemplos de vídeos representativos das três sessões de testes que ocorreram ao longo desta dissertação: Testes aos SDKs na Calçada Portuguesa (Anexo D1), Testes aos SDKs em Conímbriga (Anexo D2) e Testes de Usabilidade (Anexo D3).

### **D1: Testes aos SDKs na Calçada Portuguesa**

- <https://vimeo.com/277819004>
- <https://vimeo.com/277818982>
- <https://streamable.com/ln48r>
- <https://streamable.com/t0ziw>
- <https://streamable.com/0nqqg>
- <https://streamable.com/phzp8>

### **D2: Testes aos SDKs em Conímbriga**

- <https://streamable.com/ux7tf>
- <https://streamable.com/vmegt>
- <https://streamable.com/mv295>
- <https://streamable.com/uj57l>
- <https://streamable.com/jey6m>
- <https://streamable.com/xll2r>
- <https://streamable.com/p385c>
- <https://streamable.com/77ehm>
- <https://streamable.com/s1m8u>

### **D3: Testes de Usabilidade**

- <https://streamable.com/yqnhq>