



DESENVOLVIMENTO DE UM AMBIENTE MOBILE GAMIFICADO COM RECURSOS DE GEOLOCALIZAÇÃO E REALIDADE AUMENTADA PARA O FORTALECIMENTO DO TURISMO

Anny C. Walker Silva, Luciano Ferreira Silva, Marcelo H. Oliveira Henklain

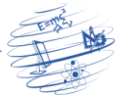
Departamento de Ciência da Computação (UFRR)
Av. Cap. Ene Garcês, 2413 - Aeroporto, Boa Vista - RR, 69310-000
annyufr@gmail.com, lucianoferreira@ufrr.br, marcelo.henklain@ufrr.br

Abstract. *The expansion of studies on Augmented Reality (AR) associated with geolocation makes it possible to analyze the efficiency of its applicability in several areas, such as tourism, a strategic area for the economic development of a region. It is in this context that this article presents the development of a mobile application guided by gamification principles, implemented from AR and geolocation technologies, whose purpose is to facilitate and promote engagement in relation to the behavior of “exploring tourist attractions in a city”, having been chosen for this work the city of Boa Vista, Roraima. To this end, steps for the construction of this application are proposed and described which, at the end of the study, resulted in an example of a combination of tools and computational techniques for the implementation of geolocation and AR, guided by gamification, with a potential to contribute to the strengthening of tourism.*

Resumo. *A expansão dos estudos sobre Realidade Aumentada (RA) associada à geolocalização possibilita a análise da eficiência de sua aplicabilidade em diversas áreas, como o turismo, área estratégica para o desenvolvimento econômico de uma região. É nesse contexto que este artigo apresenta o desenvolvimento de uma aplicação mobile orientada por princípios de gamificação, implementados a partir das tecnologias de RA e de geolocalização, cuja finalidade é facilitar e promover engajamento em relação ao comportamento de “explorar pontos turísticos de uma cidade”, tendo sido escolhido para este trabalho a cidade de Boa Vista, Roraima. Para tanto, são propostos e descritos passos da construção dessa aplicação que, no final do estudo, culminaram em um exemplo de combinação de ferramentas e técnicas computacionais de implementação da geolocalização e da RA, orientadas pela gamificação, com potencial de contribuição para o fortalecimento do turismo.*

1. Introdução

O turismo, de acordo com Gonçalves, Faria e Horta (2020), pode impactar de modo significativo na economia de uma localidade, assumindo papel proeminente na geração de emprego e renda. Em 2015, por exemplo, 3,17% dos empregos no Brasil estavam vinculados a esse setor (Gonçalves et al., 2020). Tal impacto não possui um teto pré-



definido e não está restrito a regiões específicas. Por essa razão, o turismo figura, indistintamente, como uma oportunidade para o crescimento econômico, sendo importante o desenvolvimento de estratégias para fortalecê-lo. Nesse contexto, avaliamos que a criação de tecnologias para incentivar e facilitar o comportamento do turista de explorar prédios históricos, paisagens e atrações similares em uma cidade, consiste em um modo promissor de apoio ao turismo e, em última análise, à economia.

Ao examinarmos as necessidades e características de diversos estados brasileiros, consideramos neste estudo que o fortalecimento de atividades turísticas pode ser, especialmente, útil para o Estado de Roraima. De acordo com dados do IBGE (2017), esse Estado encontra-se na última posição do *ranking* nacional em termos de receitas e economia. Investir em turismo pode ser, então, uma estratégia que culmine em crescimento econômico, o que é importante, inclusive, para a melhora da qualidade de vida da população roraimense.

Ademais, dados sugerem que Roraima possui inegável potencial turístico, tendo, entre 2010 e 2015, duplicado o seu faturamento nesse ramo da economia (Gonçalves et al., 2020). Além disso, segundo jornal local, esse setor representa, pelo menos, 10% do volume de serviços de Roraima (Folha de Boa Vista, 2016). Considerando esse cenário, surgiu a iniciativa de potencializar o turismo roraimense por meio do desenvolvimento de uma aplicação para dispositivo móvel, denominada “*Roraima Tour*”. A sua finalidade é (1) remover obstáculos que um turista possa ter para identificar quais são e onde estão os pontos turísticos de uma região, bem como (2) aumentar a probabilidade de que esse turista se engaje no comportamento de explorar esses pontos turísticos, incluindo tanto aqueles mais visitados quanto os menos famosos, contribuindo, assim, para a ampliação da efetividade econômica de todos os produtos turísticos existentes em uma cidade.

Para facilitar a identificação de pontos turísticos e, principalmente, gerar engajamento em relação à exploração desses locais, foram adotadas, no desenvolvimento do *app* “*Roraima Tour*”, técnicas de gamificação apoiadas nas tecnologias de geolocalização e de Realidade Aumentada (RA). A gamificação serve ao propósito de estimular a exploração de pontos turísticos por meio do *app*. Gamificar significa incorporar em um contexto que não é um jogo, tal como um *app* turístico, características de um jogo (e.g., desafios, pontuação). A finalidade disso é gerar engajamento (Zichermann e Cunningham, 2011), o que envolve, por exemplo, o aumento de duração do comportamento de explorar pontos turísticos. No caso do *app* “*Roraima Tour*”, ele foi gamificado no sentido de que o turista tem que cumprir desafios que consistem nas visitas aos pontos turísticos e, nesse local, precisam coletar um totem como indicação de que o desafio foi superado.

Para a visualização dos totens em cada ponto turístico, é que foram necessárias as tecnologias de geolocalização e de RA. A tecnologia de geolocalização permitiu que o *app* tivesse a capacidade de reconhecer onde o turista se encontra e, ainda, se esse local coincide com um ponto turístico específico, enquanto a RA foi o meio a partir do qual conseguimos viabilizar que o turista visualizasse na tela do seu *smartphone* totens posicionados em locais específicos, dentro de um ponto turístico.

O *app* “*Roraima Tour*” é compatível com o Sistema Operacional Android, que é reconhecidamente o mais utilizado em *smartphones* de diferentes marcas (Haris et al.,



2018), e abarca os pontos turísticos da capital roraimense, Boa Vista. Não obstante, em versões futuras do *app*, podemos adaptá-lo para outras plataformas, como o iOS, e incluir outros municípios do Estado de Roraima ou de outros Estados brasileiros. Com base em Negrusa et al. (2015), ressaltamos que, com as características atuais, o *app* “*Roraima Tour*” já pode ser considerado um caso de tecnologia sustentável para estimular a exploração turística de uma cidade, pois consegue atender às necessidades da sociedade presente, em termos de exploração econômica do turismo, sem comprometer recursos para as gerações futuras.

Vale ressaltar que neste trabalho o nosso objetivo foi descrever as tecnologias adotadas (e combinadas) para a viabilização do *app*, bem como as suas funcionalidades. Destacamos que a versão do *app* que apresentaremos é um protótipo, o qual ainda carece de um povoamento mais completo e de avaliações de usabilidade e de aceitação mais aprofundadas. Contudo, esse protótipo nos permite observar e inferir sobre o potencial da combinação das técnicas e estratégias computacionais aqui propostas com a finalidade de auxílio ao turismo.

Neste estudo, começaremos apresentando o conceito de gamificação, bem como conceitos relevantes para a compreensão do funcionamento das tecnologias adotadas, relativas à geolocalização e RA. Nesse processo, descreveremos em detalhes como essas técnicas e tecnologias se aplicam ao *app* “*Roraima Tour*”. Passaremos, então, ao exame dos trabalhos relacionados, a partir dos quais identificamos lacunas na literatura científica e características promissoras a serem incorporadas em nosso *app*. Finalmente, discutiremos a metodologia de desenvolvimento do *app* e os detalhes de sua implementação.

2. Fundamentação Teórica

Nesta seção, encontram-se os conceitos que nortearam esta pesquisa, a saber: técnicas de geolocalização, RA baseada em geolocalização, plataforma Android e gamificação. Começaremos pelo conceito de gamificação.

2.1. Gamificação

Busarello (2016) explica que a gamificação consiste no uso de técnicas, mecanismos e sistemáticas de jogos, fora do contexto de jogos, com a finalidade de manter o usuário engajado ao sistema gamificado. O autor destaca que, para gamificar um sistema, é preciso compreender e aplicar quatro características de mecanismos de jogos, a saber: (1) metas: compõem o propósito que o usuário persegue constantemente, procurando sempre cumprir; (2) regras: usadas para definir como o usuário deve se comportar e agir dentro da aplicação para atingir as metas; (3) sistema de *feedback*: meio pelo qual o usuário identifica como está a sua evolução dentro do sistema, no sentido de quanto está se aproximando ou não do alcance das metas; (4) participação voluntária: ocorre quando o usuário quer atingir as metas, concorda com as regras e se guia pelo sistema de *feedback* para avaliar seu progresso em relação ao alcance das metas.



No *app* “Roraima Tour”, as metas consistiam na exploração de pontos turísticos da cidade de Boa Vista, cadastrados previamente: Monumento aos Pioneiros; Monumento ao Garimpeiro; Portal do Milênio, na praça das Águas; e Pedra Pintada. A regra, por sua vez, determinava que para o cumprimento de cada meta era preciso não só chegar ao local do ponto turístico como, também, encontrar e capturar um totem. Por sua vez, a captura do totem, apenas possível de ser visualizado por meio do *app*, e a indicação na tela do *app* de meta cumprida, consistiam nos principais mecanismos de *feedback* em relação ao comportamento de explorar pontos turísticos. Uma vez que o uso do nosso *app* é integralmente voluntário, podemos considerar que estamos aplicando as quatro características de um jogo, em um contexto de turismo de uma cidade, que não é um jogo. Por essa razão, podemos falar que estamos gamificando o comportamento de “explorar pontos turísticos”, o que fizemos justamente para aumentar a probabilidade de ocorrência e duração desse comportamento. Em última instância, ao conseguirmos sucesso nesse objetivo, estamos contribuindo com o fortalecimento do turismo local.

Para viabilizar que o nosso *app* pudesse identificar a localização do dispositivo do usuário e exibir os totens nos locais apropriados, precisamos de duas tecnologias que apresentaremos a seguir. A primeira é a geolocalização e a segunda é a RA.

2.2. Técnicas de Geolocalização

O GPS (*Global Position System*) é um sensor capaz de localizar a posição de um dispositivo no mundo por meio de um sistema de três dimensões, conhecidas como: latitude, longitude e altitude. Seus dados são atualizados e mantidos por satélites ao redor do planeta (Alves e Negrão, 2017). Para determinar uma coordenada no GPS é utilizado um cálculo de triangulação da posição do dispositivo entre três satélites para coordenadas 2D (latitude e longitude), ou quatro satélites para coordenadas 3D (latitude, longitude e altitude). A Figura 1A ilustra o cálculo de uma coordenada 2D, por meio de triangulação de três satélites.

De posse do sistema de coordenadas do GPS é possível utilizar conceitos matemáticos para tratar os dados obtidos. Um exemplo dessa possibilidade é a tecnologia *Geofence*, que consiste em uma representação virtual de uma área geográfica circular, criada a partir do GPS, que permite detectar automaticamente quando um dispositivo entra ou sai do seu perímetro (Namiot e Sneps-Sneppe, 2013). A Figura 1B ilustra os conceitos de área circular definida pela tecnologia *Geofence* e de detecção da entrada e saída dessa área, por um determinado dispositivo.

Assim, com a tecnologia *Geofence*, é possível desenvolver uma aplicação que consiga manter registros de pontos turísticos de uma cidade, que são as áreas geográficas, e que, adicionalmente, consiga identificar se o dispositivo do usuário entrou ou saiu dessa área. Com esse recurso, é possível programar a aplicação para realizar ações, a depender da entrada ou saída de uma região, viabilizando, portanto, maior interatividade para o turista, que poderá, por exemplo, ver mensagens no *app* com informações específicas sobre determinada região da cidade, na qual se encontra em um momento específico.

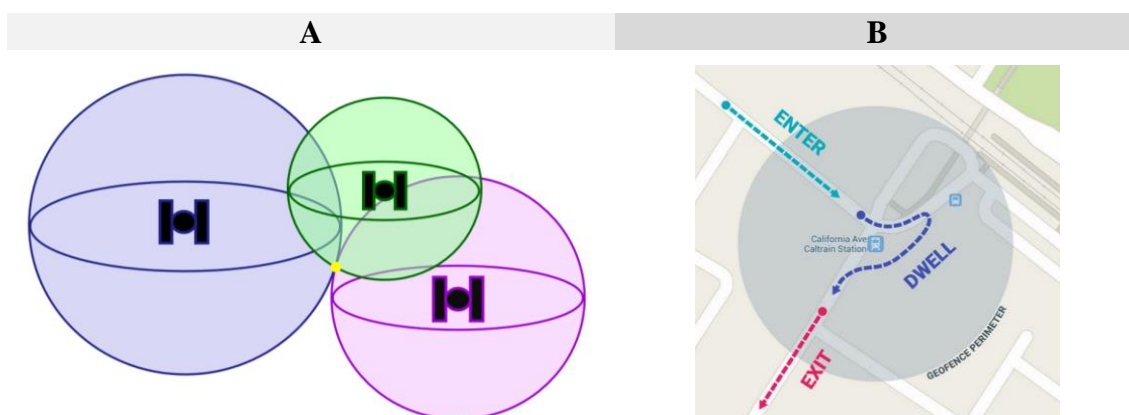


Figura 1. A - Representação de triangulação entre três satélites, em que o ponto amarelo é a posição do dispositivo (Alves e Negrão, 2017); B - Representação da aplicação da tecnologia *Geofence*.

No contexto da gamificação proposta para o *app* “*Roraima Tour*”, a tecnologia *Geofence* é crucial, para que se possa garantir que o turista chegou de fato ao ponto turístico correto, sendo possível, portanto, registrar que cumpriu uma missão e, assim, fornecer os *feedbacks* apropriados, de modo a incentivar o seu comportamento de exploração. Não obstante, avaliamos que, isoladamente, a tecnologia *Geofence* poderia ser insuficiente para implementar as técnicas de gamificação em todo o seu potencial, pois o usuário poderia ficar pouco motivado apenas com mensagens de parabéns exibidas na tela do *smartphone*. Para contornar esse risco é que adotamos a RA, como forma de inserir no ambiente do usuário um item que indicasse, de modo lúdico e típico de um contexto de jogo, a superação do desafio. Adotamos, então, a ideia de inserir totens em cada ponto de turístico, que deveriam ser capturados. Passamos, então, à exposição sobre como funciona a RA baseada na tecnologia de geolocalização.

2.3. Realidade aumentada (RA) baseada em geolocalização

Para Azuma et al. (2001), a RA consiste em incrementar a percepção do usuário e como ele interage com o mundo através de meios virtuais, combinados com ambientes reais. Os autores acrescentam, ainda, que na RA os objetos computadorizados devem aparentar-se com a realidade de maneira a coexistirem com os objetos reais. Dessa forma, sistemas caracterizados como RA devem possuir, minimamente, as seguintes características: combinação do real com o virtual; interação em tempo real; e ambientação tridimensional.

Existe uma série de tecnologias que viabilizam a criação de RA, podendo ser, por exemplo, PCs (*Personal Computer*) que captam o mundo real via uma câmera e misturam os objetos virtuais em um único ambiente, apresentando a informação ao usuário via um monitor (Krevelen e Poelman, 2010). Uma alternativa de *hardware* que vem fomentando o desenvolvimento de aplicações em RA são os *smartphones*, que compõem um sistema de custo relativamente acessível e completo, reunindo em um único dispositivo a capacidade de captura (câmera), uma unidade de processamento e o meio de exibição (*display*).



Além dos *smartphones* possuírem a câmera, a capacidade de processamento em tempo real, para misturar objetos virtuais e reais, eles possuem também um GPS embutido, o que viabiliza a implementação da RA baseada em geolocalização, que consiste em usar o sistema de coordenadas geográficas para definir onde ficarão localizados, no mundo real, os objetos virtuais do ambiente em RA. Assim, tornou-se possível a inserção de totens em locais específicos da cidade de Boa Vista, viabilizando, portanto, o funcionamento do *app* “*Roraima Tour*” tal como planejado. Finalmente, precisamos esclarecer aspectos técnicos da plataforma Android, para a qual o nosso *app* foi desenvolvido.

2.3. Plataforma Android

A plataforma Android possui 86,8% do mercado de sistemas operacionais para *smartphones*, sendo, portanto, amplamente adotada (Haris et al., 2018). Nesse sentido, ao criar *apps* para Android, o desenvolvedor amplia as chances de que sua aplicação seja conhecida e utilizada.

Em grande medida, esse sucesso do Android está associado ao fato de que o Google adquiriu a empresa Android Inc., investiu no projeto e o colocou como *Open Source*, assim como pela criação de uma aliança de grandes empresas multinacionais de tecnologia (e.g., Samsung, Motorola), liderada pelo Google, que colaboram entre si para o aperfeiçoamento dessa plataforma, o que é conhecido como *Open Handset Alliance* (OHA¹). Além disso, os smartphones com Android, tipicamente, oferecem ao usuário poder de processamento, sensores embutidos (como o GPS), código *Open Source* e bom custo-benefício (Burnette, 2009).

Tecnicamente, o kernel do Sistema Operacional (SO) Android é o Linux e, sobre essa primeira camada, existe uma camada posterior com bibliotecas que assistem o funcionamento da plataforma e que contém uma máquina virtual responsável por mediar a interação entre aplicações, desenvolvidas na linguagem Java, e o Linux. Na sequência, o Android contém uma camada de *framework* para as aplicações, responsável por permitir configurações e o adequado funcionamento das aplicações do SO, as quais compõem a última camada da arquitetura do Android e incluem funcionalidades como contatos, SMS (*Short Message Service*), telefone, *browser* para navegação na Internet, câmera, entre outras.

Assim, por viabilizar todas as tecnologias necessárias para a realização desse projeto, a plataforma Android foi escolhida para receber o *app* “*Roraima Tour*”. Agora, apresentaremos uma revisão da literatura envolvendo trabalhos que mesclam geolocalização e RA no contexto do turismo.

3. Trabalhos relacionados

Reitmayr e Schmalstieg (2004), a fim de suprir as necessidades e requisições de turistas de Viena, desenvolveram uma aplicação com o recurso de RA, baseada em

¹ <https://www.openhandsetalliance.com/>

geolocalização, que torna possível navegar e ver informações adicionais em um ambiente. Eles usaram um mini notebook com *Windows XP*, GPS e sensor de orientação, acoplados a uma câmera em um capacete, com um óculos *optical-see-through* da Sony (*Glasstron*).

Para a implementação dessa aplicação, os autores usaram uma plataforma de desenvolvimento conhecida como *Studierstube*, que suporta comunicação entre diferentes dispositivos, para gerar um ambiente RA. A interface do usuário é mostrada no *display*, podendo o conteúdo ser de origem textual ou objeto gráfico, renderizado de acordo com o espaço físico. Os objetos gráficos dessa aplicação foram desenhados de forma a complementar o ambiente com naturalidade, enquanto os objetos textuais foram dispostos para informar e guiar o usuário pela navegação do ambiente. A Figura 2A ilustra a interface da aplicação.

Outra abordagem, no campo da RA aplicada no turismo, foi o trabalho de Fritz, Susperregui e Linaza (2005), realizado na cidade de San Sebastián, na Espanha. Eles desenvolveram o projeto PRISMA, que adiciona RA aos binóculos de turismo, nos quais o turista pode inserir uma moeda e olhar pontos turísticos distantes. Para implementação, os autores usaram o *ARToolKit*, com suporte às bibliotecas gráficas *OpenGL* e *VRML*. O ambiente RA criado foi acoplado ao binóculo, com a câmera e um sensor rastreador, capaz de detectar o ponto de vista e a rotação. O PRISMA foi modelado de forma que os sensores possibilitassem ao usuário selecionar do banco de dados as informações referentes às entradas, contextualizando os ambientes vistos através da câmera. A Figura 2B ilustra esquematicamente esse processo.

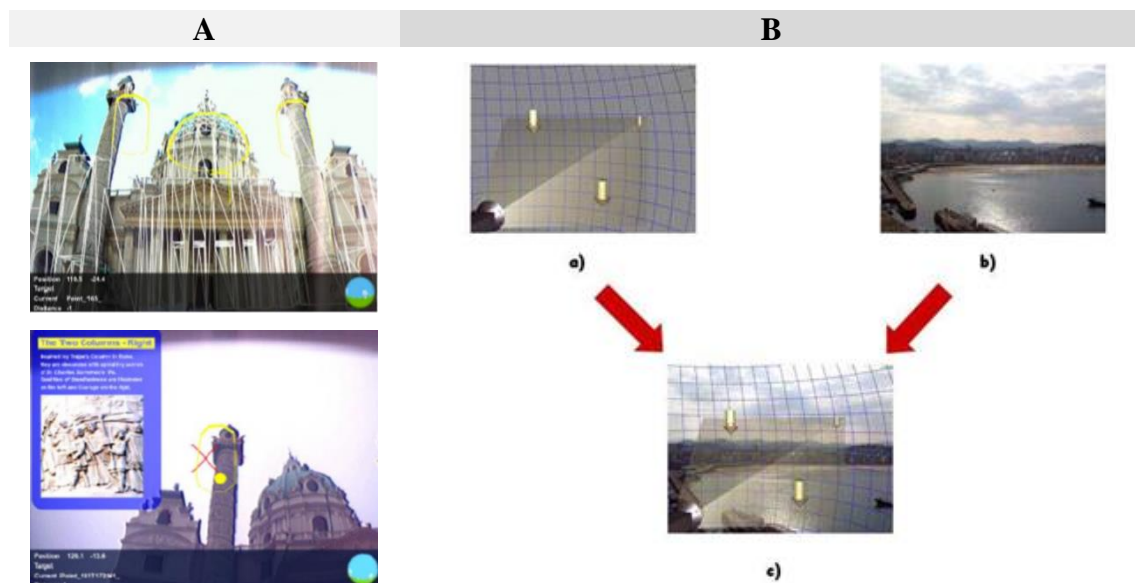


Figura 2. A - Interface da aplicação desenvolvida por Reitmayr e Schmalstieg (2004); B - Resultado do trabalho de Fritz, Susperregui e Linaza (2005).

Identificamos, ainda, o trabalho de Malca et al. (2019), no qual foi desenvolvido o *app* ARTOUR, cuja principal funcionalidade consiste no uso da RA para a exibição em 3D dos atrativos turísticos da cidade de Riobamba, no Equador. Essa exibição é acionada



quando o usuário aponta a câmera do seu *smartphone* para um dos pontos turísticos cadastrados, o que o permite ver em sua tela uma réplica em 3D dessa atração e uma explicação sobre ela. Os pesquisadores desenvolveram esse *app* com a finalidade de aumentar a facilidade e o interesse dos turistas por conhecer a cidade. Para a construção do *app* foi utilizado Flutter v1.7.8, sendo necessário para a criação da RA o uso do Unity v18.3.7 e do Vuforia SDK (*Software Development Kit*) v8.3.8, além de tecnologias de geolocalização.

Nesses três trabalhos que examinamos, foi possível notar a viabilidade técnica de se combinar as tecnologias de RA e de geolocalização e como essa implementação pode ser realizada. Além disso, estudos como o de Cranmer, Dieck e Fountoulaki (2020) sugerem que a RA associada à geolocalização provê aos turistas uma forma inovadora de explorar atrações de uma cidade, permitindo que obtenham informações relevantes, interativas e consistentes com o local no qual se encontram e que isso tem sido efetivo para engajar os turistas nesse comportamento de explorar pontos turísticos. Cranmer et al. (2020) salientam, ainda, que no contexto turístico, a aplicação da RA tem sido realizada, principalmente, por meio de *smartphones* em função da facilidade de implementação, baixos custos, confiabilidade e a capacidade de executar aplicações robustas de RA.

Ressalta-se que, nos estudos examinados, combinados com os achados da revisão sistemática sobre pesquisas aplicando RA no contexto turístico, conduzida por Liang e Elliot (2021), encontramos exemplos de uso dessas tecnologias na Europa (e.g., Reino Unido, Irlanda, Espanha, Grécia, Áustria, Noruega, Dinamarca, Finlândia, Eslováquia, Romênia, República Tcheca), na Ásia (e.g., Coreia do Norte e Malásia), na América do Norte (e.g., Estados Unidos) e na América do Sul (e.g., Equador). Todavia, não encontramos relatos de uso da RA e da geolocalização no Brasil de modo geral e, tampouco, em estados brasileiros da região Norte. Isso indica a existência de uma lacuna na literatura científica porque, embora a tecnologia já esteja bem consolidada, ainda não dispomos de dados empíricos, para garantir a efetividade de sua aplicação em relação ao comportamento de turistas em um contexto cultural diverso, em relação aos países já investigados.

Nesse sentido, a condução de um estudo no Brasil e, especificamente, em um estado que precisa encontrar alternativas para acelerar o seu desenvolvimento econômico, consiste em uma contribuição relevante do ponto de vista científico e social. Ademais, notamos também que, muitos dos estudos existentes, embora se preocupem com a questão do engajamento dos turistas, não adotam, de modo deliberado, técnicas de gamificação para potencializar o engajamento em relação ao uso da tecnologia de apoio ao turismo. Desse modo, inovamos neste estudo em relação ao desenvolvimento de um *app* para uma nova localidade, bem como pelo fato de que essa aplicação foi construída segundo princípios de gamificação, de tal modo que a RA e a geolocalização são usadas como recursos promotores de engajamento.



4. Metodologia de Desenvolvimento

Como primeiro passo no desenvolvimento da aplicação, buscou-se modelar o sistema, determinando, inicialmente, o papel de cada tecnologia, assim como a relação entre elas e como seriam utilizadas para cumprir o objetivo de gerar um ambiente RA gamificado, que auxilie o turismo de Roraima. Apresentamos a seguir os elementos para a compreensão da arquitetura da aplicação “*Roraima Tour*”.

A gamificação foi modelada, definindo como metas para o usuário, a atividade de capturar totens em cada *Geofence* de ponto turístico visitado. As regras estabelecem que o usuário só pode coletar o totem da meta, quando estiver no perímetro correspondente ao ponto turístico. O sistema de *feedback* permite que o usuário veja, marcado no PIN (*Personal Identification Number*) correspondente ao *Geofence*, o status do seu objetivo: incompleto ou completo.

Para a implementação da aplicação, foram usadas as seguintes ferramentas: o Android Studio com as linguagens Java e XML (eXtensible Markup Language); o Firebase como banco de dados, que foi escolhido devido a sua possibilidade de acesso global, flexibilidade e gratuidade; e o Unity com o kit de RA KudanAR, usando a linguagem C#, para a construção do ambiente em RA. A Figura 3 apresenta a arquitetura construída para a aplicação.

Essa arquitetura explica a relação entre as tecnologias e o evento gerado por cada interação. O usuário que possuir um dispositivo Android com o produto da pesquisa instalado, ao se movimentar, gerará atualização nas coordenadas do aparelho. A aplicação precisa ser iniciada, mas não necessita estar em primeiro plano.

Uma classe Java funcionando como um serviço em *background*, monitora as atualizações de coordenadas e verifica se a coordenada atual está dentro do perímetro de alguma *Geofence* pré-determinada. Se a coordenada não estiver em uma das *Geofences* listadas, o serviço continua normalmente. Do contrário, é enviada uma notificação ao usuário, informando que ele adentrou em uma *Geofence* conhecida.

O usuário pode optar por desconsiderar notificações do *app*, exceto dentro de *Geofences*, no qual a notificação aparece e é mantida na tela até que uma ação seja apresentada pelo usuário. Ao ser clicada, ela inicia uma *Activity*², que busca dados referentes ao *Geofence* atual no banco de dados Firebase e os organiza na tela. O usuário só tem acesso aos dados do local enquanto estiver dentro da *Geofence* correspondente a esse local. Se ele sair do perímetro, perderá esse acesso.

Com a *Activity* referente a *Geofence* atual, o usuário possui três opções: ver a história do local em forma de texto; ver álbum de fotos do local e tirar suas próprias fotos; e iniciar o modo caça ao tesouro em RA, no qual ele procurará com a câmera por um totem relacionado ao ponto turístico. O totem possui suas próprias latitude e longitude dentro do perímetro da *Geofence* do local, que ficam armazenadas no banco de dados.

² Refere-se a um evento específico em relação ao qual o usuário pode agir. Uma vez que as *activities*, geralmente, interagem com o usuário, a própria classe *Activity* cria uma janela que o desenvolvedor pode configurar do modo que desejar.

No Unity, a biblioteca que trabalha com coordenadas possui limitação na precisão devido à quantidade de casas decimais disponíveis, sendo do tipo *int*, enquanto as coordenadas enviadas por *broadcast* da aplicação principal são do tipo *double*, tendo, portanto, maior precisão. Para lidar melhor com esses dados, foi criado um tipo de vetor de duas dimensões que trata latitude e longitude como *double*. Para calcular a distância entre usuário e totem, foi usada a equação de distância entre dois pontos. Foi desenvolvida, ainda, uma biblioteca Java capaz de ler os dados (sobre o usuário, latitude e longitude do totem e do usuário), enviados por *broadcast* em formato JSON, a partir da aplicação principal.

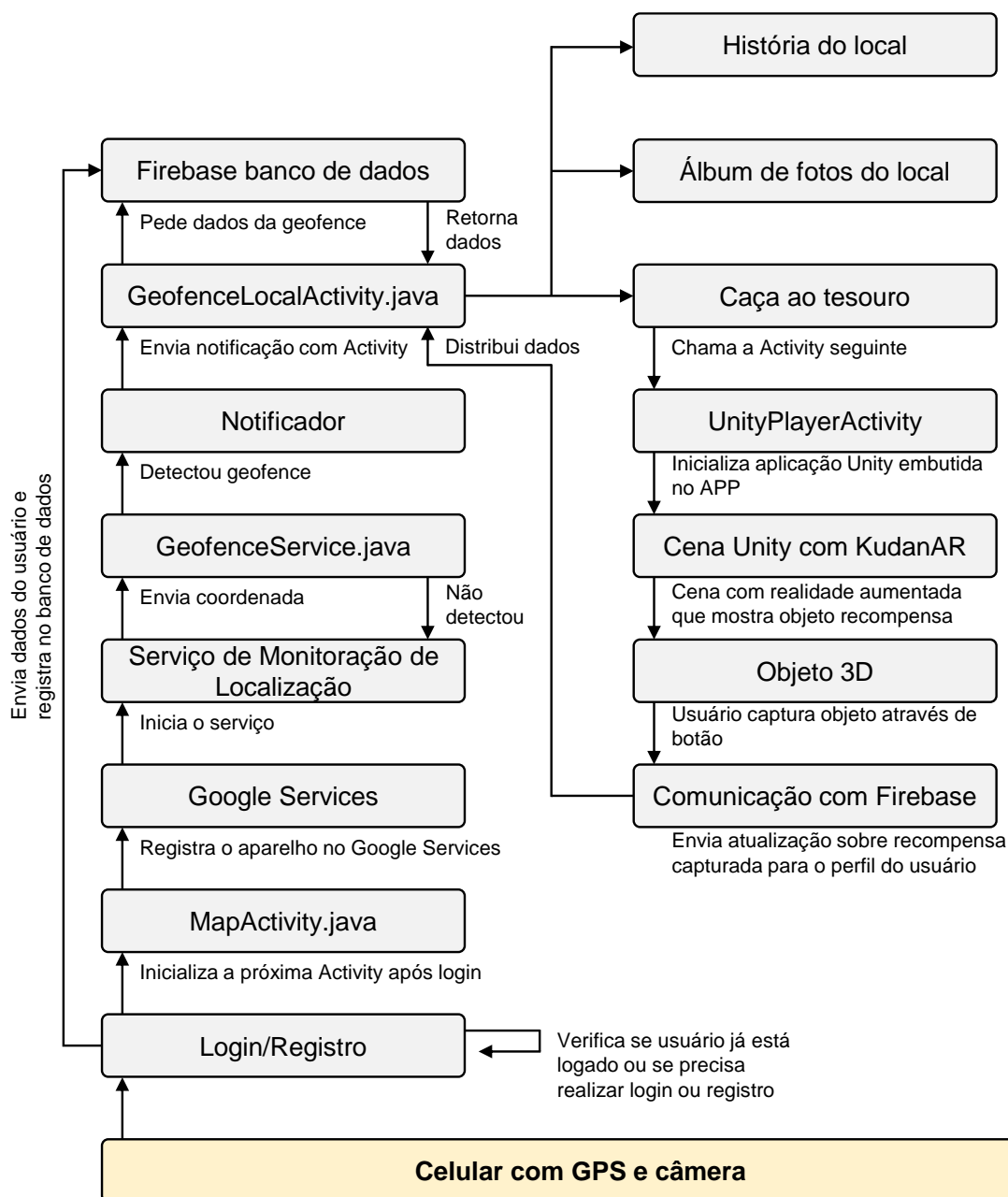


Figura 3. Arquitetura da aplicação “Roraima Tour”.



Quando o usuário com o dispositivo se encontra próximo à latitude e longitude do totem, ele aparece na tela do usuário, através do KudanAR, juntamente com um botão que permite a captura desse totem. O KudanAR, então, rastreia a posição em três dimensões do chão e a orientação em três dimensões do dispositivo em relação ao chão, e permite a visualização do objeto 3D na câmera. Após o clique no botão de captura, o Unity se comunica com o Firebase e envia a atualização de dados referentes ao ponto turístico visitado pelo usuário.

5. Detalhes e Resultados da implementação

A implementação foi dividida em módulos de forma a auxiliar na visualização dos resultados e das relações entre elementos essenciais do sistema, além de facilitar a manutenção e organização do projeto conforme orientações de Parnas (1972). A modelagem de módulos foi organizada da seguinte forma:

- Módulo de acesso: é a aplicação base com sistema de *Login / Logout* e *Sign Up* associado ao Facebook, com telas de perfil de usuário e dinâmica de botões e transições de tela.
- Módulo principal: ele é responsável pela tela do menu principal com um mapa de Roraima e os pontos turísticos marcados por um PIN e, ainda, provê o serviço em *background* de reconhecimento de *Geofence* utilizando a geolocalização do aparelho.
- Módulo de gamificação: aplicação que contém as funcionalidades disponíveis em cada ponto turístico, incluindo (1) caça ao tesouro, que permite ao usuário procurar através da câmera de seu aparelho o totem escondido, (2) álbum de fotos, que só é possível ver no local e (3) a história relacionada a cada lugar.
- Módulo administrativo: é responsável por lidar com as atualizações das missões e manejar a comunicação entre servidor e cliente.

Nas próximas subseções são apresentados os resultados de implementação de cada módulo e a explicação de como eles funcionam e interagem entre si. A título de informação, as imagens do *app* “Roraima Tour”, apresentadas nas próximas seções, foram capturadas a partir de um aparelho Android Moto G4 Plus (Resolução: 720 x 1280 pixels).

5.1. Resultados do Módulo 1

No Módulo 1, temos as telas de Login, Figuras 4A e 4B, de Cadastro, Figura 4C, e de Perfil do usuário, Figura 4D. As classes e arquivos importantes para constituir essas telas são os seguintes:

- *Activity_login.xml*: possui todos os elementos visuais que compõem a tela de Login;

- `CommonActivity.java`: esta classe estende de `AppCompatActivity.java` (uma classe padrão Android que trata diversos elementos de interação com o usuário como mensagens do tipo *toast*, barra de progresso *dialog* etc.). A classe `CommonActivity.java` possui métodos em comum utilizados por todas as `Activities` do presente projeto.
- `LoginActivity.java`: esta classe estende de outra classe (`CommonActivity.java`), sendo a responsável por fazer a interação entre os elementos `activity_login.xml` e o banco de dados Firebase. Essa classe possui os métodos necessários para testar o login do usuário no banco de dados, utilizando a Internet, e permitir que ele faça *login* com sua conta do Facebook, como mostrado na Figura 4B. Essa classe também realiza a checagem de arquivo *cache* para saber se o usuário já fez *login* no aplicativo anteriormente. Caso ele já tenha feito o *login*, essa tela não aparecerá e o usuário será diretamente redirecionado para a tela principal do *app*. Essa tela leva para a tela principal caso o *login* seja realizado ou para tela de registro de novo cadastro caso o usuário ainda não possua um.

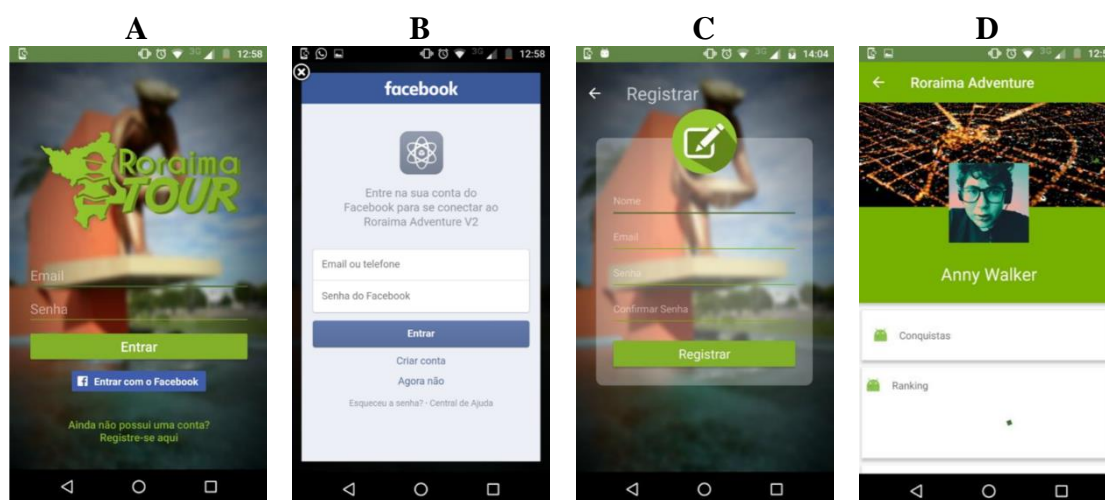


Figura 4. Prints das telas resultantes do Módulo 1 do *app* “Roraima Tour”.

Para funcionar e existir, a tela de Cadastro, Figura 4C, utiliza os seguintes arquivos:

- `Activity_sign_up.xml`: esse arquivo possui todos os elementos visuais que compõem a tela de Cadastro;
- `SignUpActivity.java`: essa classe estende de `CommonActivity.java`. A `SignUpActivity.java` é responsável por armazenar no banco de dados Firebase os dados do usuário e levá-lo para a tela de *login* para realizar o seu acesso à aplicação. Essa classe também faz os testes de campo para detectar erros ou inconsistência de dados.



A tela de Perfil de usuário, Figura 4D, para funcionar e existir, utiliza os seguintes arquivos:

- `Activity_profile.xml`: esse arquivo possui todos os elementos visuais que compõem a tela de Perfil;
- `ProfileActivity.java`: esta classe é responsável por fazer a comunicação com o banco de dados do Firebase ou do Facebook de modo a resgatar os dados do usuário logado e mostrar na tela para ele.

5.2. Resultados do Módulo 2

A tela Mapa / Menu principal, Figura 5, é composta pelos arquivos:

- `Activity_maps.xml`: esse arquivo possui todos os elementos visuais que compõem a tela Mapa / Menu principal. Utilizando a chave do projeto “*Roraima Tour*” no console do Google é gerado o mapa com os botões de interação, como ilustra a Figura 5A. Os principais botões são apresentados com detalhes na Figura 6, sendo eles:
 - Botão A: para entrar na missão que o usuário se encontra no Geofence atual, Figura 5C;
 - Botão B: para acessar o perfil;
 - Botão C: para abrir a lista de missões, como pode-se observar na Figura 5B;
 - Botão D: para abrir a atualização de novas missões;
 - Botão E: faz o trajeto do local atual até a missão desejada, como ilustra a Figura 5D.
- `MapActivity.java`: essa classe estende de `FragmentActivity.java` (não é `AppCompatActivity.java`, pois essa tela precisa dos métodos encontrados na classe `FragmentActivity.java` para lidar com o mapa do Google). Essa classe também implementa todas as classes necessárias para lidar com os serviços do Google. A `MapActivity.java` conecta o celular do usuário nos serviços Google (isso é necessário para obter a localização do usuário e o serviço de mapa). A `MapActivity.java` também inicia o processo de monitoração de localização e checagem de *Geofence* do usuário. Ela também cria as *Geofences*. No futuro, a criação de *Geofences* será feita a partir de uma classe do tipo serviço que fará isso em *background* utilizando *beacons* do Google.

A monitoração de localização e *Geofence* faz com que seja gerada uma notificação no celular do usuário, quando ele se encontra no raio de um dos *Geofences*. Essa notificação gera um pacote com informações do *Geofence* determinado, sendo, então, criada uma tela para o usuário visualizar essas informações.

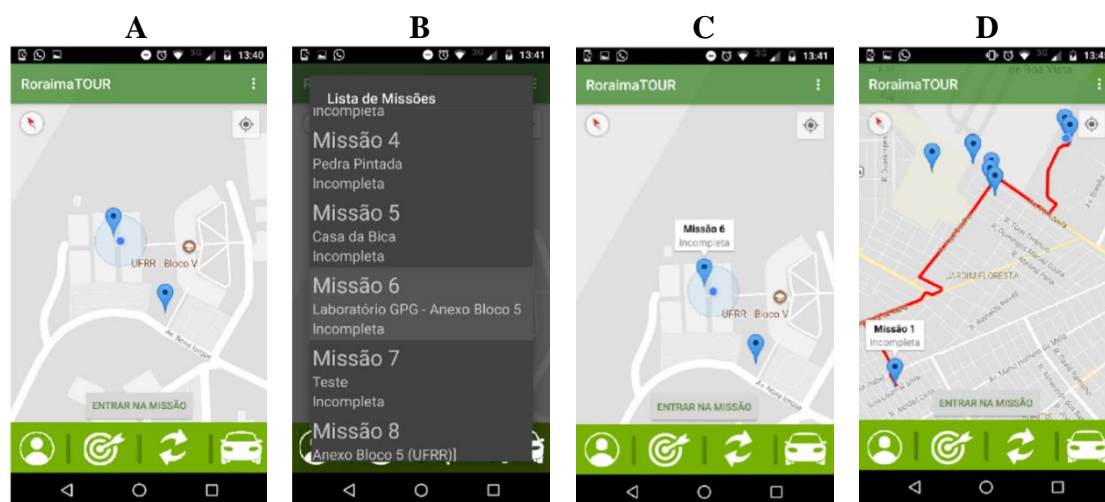


Figura 5. Prints das telas resultantes do Módulo 2 do app “Roraima Tour”.

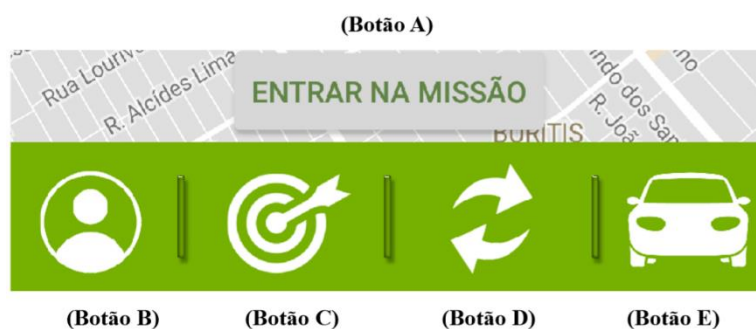


Figura 6. Prints dos botões do menu principal do app “Roraima Tour”.

A tela *Geofence* local é composta pelos arquivos:

- Activity_Geofence_local.xml: esse arquivo possui todos os elementos visuais que compõem a Figura 7A. As opções de botões são respectivamente: História/descrição associada ao local, exemplo na Figura 7B; tirar foto para salvar no álbum de fotos associado ao local, como nas Figuras 7C e 7D. O botão de caça ao tesouro, o terceiro na Figura 7A, é implementado no Módulo 3.
- Geofence LocalActivity.java: essa classe estende de *CommonActivity.java*. *Geofence LocalActivity.java* é responsável por exibir os dados do *Geofence* em que o usuário se encontra e permitir a interação entre o usuário e as possibilidades do local através dos botões de compartilhamento nas redes sociais, botão de história do local, botão de fotos do local e botão de caça ao tesouro utilizando RA (Módulo 3) do local.

Na lista a seguir, apresentamos outras classes em *background* ou auxiliares:

- LibraryClass.java: classe que contém os métodos para lidar com o SharedPreferences (arquivo local do Android).
- User.java: classe que contém os métodos e atributos de um dado do tipo usuário.
- CustomApplication.java: estende de Application.java, sendo responsável por inicializar serviços que serão utilizados em todas as outras classes do aplicativo (Firebase e FacebookSDK).
- Geofenceservice.java: estende de IntentService.java. A Geofenceservice.java lida com a criação da notificação contendo o pacote de informações da *Geofence* local e gera uma *intent* (“tela”) para quando o usuário clicar na notificação, isso abrir uma nova tela com as informações do *Geofence* local.
- SplashScreenActivity.java e activity_splash_screen.xml: tela de *splash screen* (ou boas-vindas) do aplicativo.

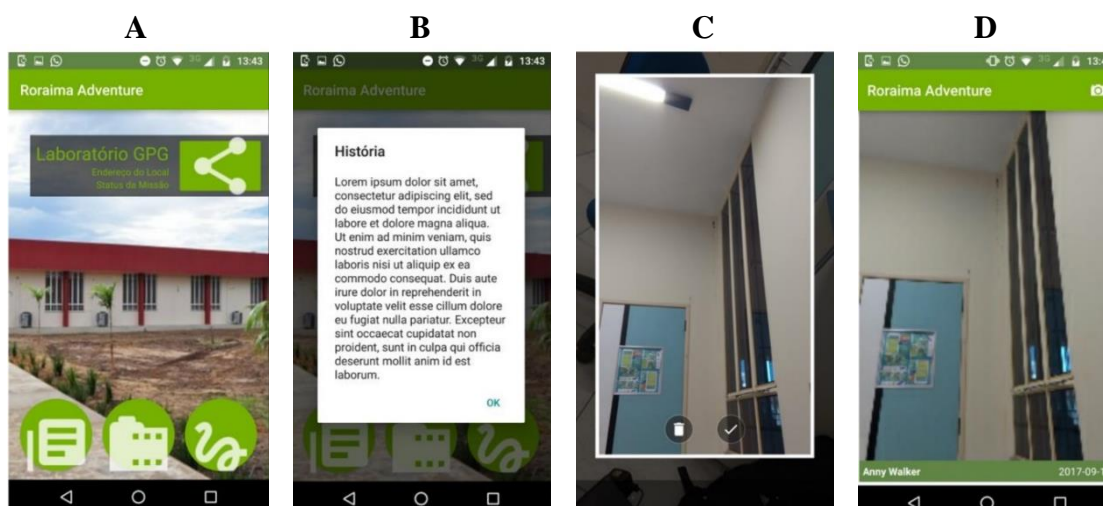


Figura 7. Prints da Activity correspondente a *Geofence* do app “Roraima Tour”.

5.3. Resultados do Módulo 3

O Módulo 3 realiza a comunicação entre Unity e Android. A aplicação Unity é necessária, pois é nela que é feita a parte de computação gráfica, dispondo o objeto 3D na câmera do dispositivo e os botões e textos de interface, como na Figura 8, recurso do *app* disponível no terceiro botão da Figura 7A. A localização do usuário é necessária para captar o local e momento exato para *spawnar* / inserir na tela o totem do ponto turístico. A API de geolocalização do Unity não foi suficiente para indicar com precisão a posição atual do dispositivo do usuário, gerando cerca de quatro segundos de atraso para a execução do processo, pois a atualização de localização da API Unity de Localização é de baixa frequência. Por isso, foi desenvolvido um *plugin* Android para Unity que ouça mensagens de *broadcast* enviadas por um serviço Android capaz de enviar a localização atual do usuário em tempo de atualização razoável.

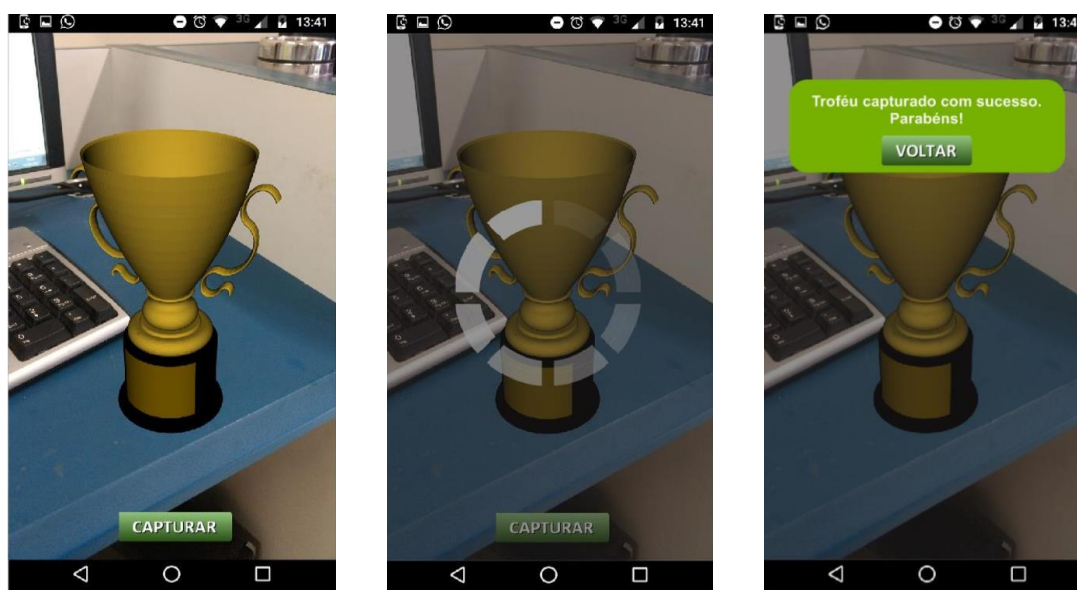


Figura 8. Prints do modo caça ao tesouro do app “Roraima Tour”.

O Módulo Android separado, que se comunica com a aplicação Unity, é composto pelos seguintes elementos:

- MainActivity.java e activity_main.xml: arquivos que juntos geram uma tela com um botão para iniciar o *broadcast* de uma mensagem / *string*.
- MyService.java: classe que estende de Service.java e possui um método do tipo Runnable que autoexecuta novamente em um segundo. O método Runnable faz *broadcast* com contador para ser recebido pelas aplicações que possuam o filtro adequado para conseguir detectar a mensagem.
- Módulo Unity Separado que se comunica com aplicação Android: Para fazer um *plugin* Unity com Android foi necessária a criação de projeto Android exportado do tipo .jar (biblioteca), que foi, então, adicionado no projeto Unity, o qual foi denominado de Plugin BroadcastReceiver.
- MyReceiver.java: essa classe estende do tipo BroadcastReceiver.java que implementa obrigatoriamente o método onReceive. Esse método trata um determinado pacote de dados recebido via *broadcast* e armazena o resultado em uma variável estática.
- Projeto Unity: o projeto Unity tem como biblioteca Android o Plugin BroadcastReceiver e altera o AndroidManifest.xml (normalmente gerado automaticamente pelo Unity, mas que pode ser editado) para adicionar o filtro adequado que faz com que o Módulo Unity enxergue o *broadcast* de mensagem do Módulo Android. A partir disso, a classe Receiver.csharp, que trata a classe MyReceiver.java da biblioteca do plugin, recebe a mensagem que chegou pela classe MyReceiver.java e armazena essa mensagem em uma variável que pode ser manipulada no projeto Unity. Além disso, o projeto usa a classe



SaveToFirebase.csharp para registrar a captura do totem no banco de dados Firebase associando a conquista ao usuário logado na aplicação.

5.4. Resultados do Módulo 4

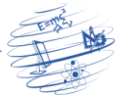
Esse módulo foi construído para tornar o “Roraima Tour” um aplicativo dinâmico, com atualizações e melhorias, como estratégia para manter o *app* sendo sempre interessante, novo e de uso contínuo pelo público-alvo. A Figura 9 exibe uma imagem dessa tela, que foi desenvolvida à parte, tendo sido construída em PHP (*Hypertext Preprocessor*), HTML (*Hypertext Markup Language*) e CSS (*Cascading Style Sheets*), com *framework* Bootstrap do Twitter, em formato de página, tendo a finalidade de permitir a inserção de novas missões no mapa do aplicativo. A interface da página possui formulário de preenchimento sobre os detalhes da missão e botão para inserção. Esse botão envia para o Firebase Cloud Database um JSON com todos os valores do formulário, usando CURL em PHP e usando o método `sendPushNotification()`, enviando, assim, para todos os dispositivos que têm a aplicação instalada para receber uma notificação com todos os valores do formulário.

Para que a aplicação já instalada no dispositivo do usuário tome conhecimento dessa atualização de missão, foi desenvolvido um submódulo no projeto geral em *background* à aplicação, também com FCM (*Firestore Cloud Messaging*), para criar uma classe `NotificationUtils.java` capaz de lidar com mensagens recebidas pelo FCM e transformar em notificação o conteúdo. Foi criada também uma classe chamada `MyFirebaseMessagingService.java`, capaz de identificar em *background* quando o serviço FCM envia uma mensagem para o dispositivo através da *web*. Caso o dispositivo esteja sem conexão à Internet, quando a nova missão for cadastrada, ele não receberá a notificação. Para reparar tal disfuncionalidade, foi adicionado um botão na `MissionsActivity.java`, em que o usuário pode solicitar a atualização manual de missões. Sempre que atualizadas, as missões são inseridas como PIN no mapa e adicionadas ao banco de dados interno de missões para que estejam disponíveis mesmo *offline*.

6. Discussão

Os testes com o *app* “Roraima Tour” foram realizados pela primeira autora em um Moto G4 Plus, versão Android 6.0.1, com conexão Oi Velox 2mbps via Wi-Fi. Identificamos que existe uma fragilidade da aplicação com relação a sua dependência de Internet, uma vez que os dados são constantemente consultados, cadastrados e atualizados em um banco de dados externo. Observamos que, o problema majoritário, decorrente de uma conexão com baixa velocidade, é a redução da taxa de atualização de quadros dos ambientes da aplicação. Esse problema pode impactar negativamente a experiência do usuário, principalmente quando ele está usando a tecnologia de RA, pois quebra a sua imersão no ambiente virtual.

Com relação aos pontos positivos deste trabalho, destacamos que a modelagem e comunicação com as aplicações Android e Unity, através de biblioteca desenvolvida em linguagem de programação Java e JSON (*JavaScript Object Notation*), abre portas para



diferentes aplicações e ideias. Não obstante, consideramos promissora a proposta de Malca et al. (2019), no sentido de usar Flutter como SDK (*Software Development Kit*) de desenvolvimento de modo a viabilizar, por meio de uma única linguagem, Dart, o desenvolvimento de aplicações que sejam compatíveis com múltiplos sistemas operacionais.

Ainda do ponto de vista técnico, merece destaque o módulo de cadastro de novas *Geofences* e atualização de missões, o que torna esta aplicação dinâmica e, especialmente, valiosa. Essa funcionalidade permite que este projeto possa ser facilmente adaptado para diversas outras aplicações, como a navegação pelo *campus* de uma universidade ou no centro de uma cidade. Ademais, o presente trabalho serve como ponto de partida para a criação de *apps* análogos para outras cidades brasileiras, abrindo uma linha de investigações promissora sobre o uso de tecnologias para o fortalecimento do turismo.

Destacamos que, embora o *app* “*Roraima Tour*” ainda seja um protótipo, verificamos que ele avança em relação ao estado da arte, uma vez que funciona em um smartphone comum, sem a necessidade de capacete ou óculos especiais, como no caso do trabalho de Reitmayr e Schmalstieg (2004). É preciso considerar, também, que o smartphone é um dispositivo prático para carregarmos, que as pessoas já utilizam e que estão habituadas a usar. Além disso, o smartphone permite ampla mobilidade do turista no processo de exploração de uma cidade e agrega, em um único dispositivo, outros serviços de que o turista possa precisar, como *app* para pedido de motorista, *app* de troca de mensagens, *app* de fotos e *apps* de redes sociais. Na proposta de Fritz et al. (2005), por exemplo, a mobilidade e a oferta de outros serviços fica prejudicada, uma vez que a RA foi adicionada aos binóculos de turismo, que ficam instalados em uma posição fixa.

Outro ponto que merece destaque em relação ao presente *app*, é que ele visa favorecer o comportamento de “explorar pontos turísticos de uma cidade”. Malca et al. (2019), por exemplo, buscaram facilitar o acesso a informações sobre atrativos turísticos, a partir da exibição de modelos 3D desses atrativos, acompanhada por informações relevantes. Contudo, não trabalharam com a organização de condições para favorecer a exploração da cidade e, ainda, para consequenciar esse comportamento, o que foi feito no presente trabalho, por meio de técnicas de gamificação.

Ressaltamos que, de fato, esse tipo de tecnologia pode trazer um impacto positivo para o turismo e para a economia de um município, merecendo ser aperfeiçoada e colocada em prática (Cranmer et al., 2020; Elliot, 2021). Assim, entendemos que o *app* que desenvolvemos merece ser avaliado em estudos futuros, tendo sido a ausência de teste com usuários a principal limitação desta pesquisa.

7. Considerações Finais

A aplicação finalizada, “*Roraima Tour*”, é funcional e entrega o que foi proposto, oferecendo um ambiente em RA baseado em geolocalização por meio de uma interface de navegação por *Geofences* cadastradas no banco de dados. Como trabalho futuro, para que o *app* possa ser disponibilizado para *download*, é sugerido um novo módulo capaz de realizar persistência de dados e tratar os momentos de falta de contato com a Internet. Também é necessário que novos estudos avaliem a usabilidade da aplicação e,



principalmente, o seu grau de efetividade em relação à facilitação e promoção de engajamento do comportamento de “explorar pontos turísticos de uma cidade”.

Por fim, ressaltamos que, com base em nossa revisão de literatura, este é o primeiro trabalho no extremo norte a desenvolver uma aplicação para favorecer o comportamento do turista ao explorar atrações de uma cidade. Esperamos, assim, estimular o interesse de outros pesquisadores para colaborar com a construção de soluções tecnológicas e na investigação de efetividade dessas ferramentas, em relação ao fortalecimento do turismo brasileiro.

Referências

- Azuma, R., Baillot, Y., Behringer, R., Feiner, S., Julier, S., and Macintyre, B. (2001). Recent advances in augmented reality. Disponível em: <https://www.cc.gatech.edu/~blair/papers/ARsurveyCGA.pdf>. Acesso em: 10 jan. 2018.
- Burnette, E. (2009). *Hello, Android: Introducing Google's Mobile Development Platform*. Raleigh, North Carolina: Pragmatic Bookshelf.
- Busarello, R. I. (2016). Gamification: Princípios e estratégias. Recuperado de https://edisciplinas.usp.br/pluginfile.php/4455428/mod_resource/content/1/Gamificat ion.pdf. Acesso em: 25 ago. 2019.
- Cranmer, E. E., Dieck, M. C. T., e Fountoulaki, P. (2020). Exploring the value of augmented reality for tourism. *Tourism Management Perspectives*, 35:100672. <https://doi.org/10.1016/j.tmp.2020.100672>
- Folha de Boa Vista. (2016). Economia: Em RR, turismo representa mais de 10% do volume de serviços. Folha de Boa Vista, Editora Boa Vista. Recuperado de <http://www.folhabv.com.br/noticia/Em-RR-turismo-representa-mais-de-10--do-volume-de-servicos/13277>. Acesso em: 21 ago. 2017.
- Fritz, F., Susperregui, A., and Linaza, M. T. (2005). Enhancing cultural tourism experiences with augmented reality technologies. In: M. Mudge, N. Ryan, e R. Scopigno (Editors), *The 6th International Symposium on Virtual Reality, Archaeology and Cultural Heritage VAST*. Recuperado de <https://bit.ly/3Rnz793>
- Gonçalves, C. C. S., Faria, D. M. C. P., e Horta, T. A. P. (2020). Metodologia para mensuração das atividades características do turismo: Uma aplicação para o Brasil e suas unidades da federação. *Revista Brasileira de Pesquisa em Turismo*, 14(3):89-108. <http://dx.doi.org/10.7784/rbtur.v14i3.1908>
- Haris, M., Jadoon, B., Yousaf, M., and Khan, F. H. (2018). Evolution of Android Operating System: A review. *Asia Pacific Journal of Contemporary Education and Communication Technology*, 4(1), 178-188. <https://www.doi.org/10.25275/apjcectv4i1ict2>
- IBGE. (2017). Portal Cidades@ – Roraima. Recuperado de <https://cidades.ibge.gov.br/brasil/rr/panorama>. Acesso em: 03 jul. 2022.



- Krevelen, D. W. F. van, and Poelman, R. (2010). A Survey of Augmented Reality Technologies, Applications and Limitations. *International Journal of Virtual Reality*, 9(2):1-20. <https://doi.org/10.20870/IJVR.2010.9.2.2767>
- Liang, L. J., and Elliot, S. (2021). A systematic review of augmented reality tourism research: What is now and what is next? *Tourism and Hospitality Research*, 21(1):15-30. <https://doi.org/10.1177/1467358420941913>
- Malca J. Y., Carrasco B. C., Guamán, P. V., Delgado J. A., Vinueza P. G. N., Guevara B. B., and Ruiz F. O. (2019). ARTOUR: Augmented Reality for Tourism - A Case Study in Riobamba, Ecuador. In: *International Conference on Information Systems and Computer Science* (INCISCOS, pp. 116-123). <https://doi.org/10.1109/INCISCOS49368.2019.00027>
- Namiot, D., and Sneps-Sneppe, M. (2013). Geofence and Network Proximity. In: Balandin, S., Andreev, S., Koucheryavy, Y. (eds.), *Internet of Things, Smart Spaces, and Next Generation Networking* (pp. 117-127). ruSMART NEW2AN 2013. Lecture Notes in Computer Science, 8121. Springer, Berlin, Heidelberg. https://doi.org/10.1007/978-3-642-40316-3_11
- Negrão, A., e Alves, A. (2017). GPS: Conceitos, funcionamento e utilidade. Disponível em: <https://bit.ly/3tb53Wc>. Acesso em: 25 ago. 2019.
- Negrusa, A. L., Toader, V., Sofica, A., Tutunea, M. F., and Rus, R. V. (2015). Exploring gamification techniques and applications for sustainable tourism. Disponível em: <http://www.mdpi.com/2071-1050/7/8/11160/htm>. Acesso em: 10 set. 2017.
- Parnas, D. L. (1972). On the criteria to be used in decomposing systems into modules. *Communications of the ACM*, 15(12):1053-1058. Recuperado de <https://bit.ly/3GMpoUN>. Acesso em: 13 set. 2017.
- Reitmayr, G., and Schmalstieg, D. (2004). Collaborative augmented reality for outdoor navigation and information browsing. In: *Proceedings of the Symposium on Location Based Services and TeleCartography*. Recuperado de <https://bit.ly/48csygi>. Acesso em: 17 jan. 2018.
- Zichermann, G., and Cunningham, C. (2011). *Gamification by design: Implementing game mechanics in web and mobile app*. Sebastopol, Canada: O'REILLY.