

Reformulação do Projeto de Usabilidade da Ferramenta Lagarto

Samir Chaves

Universidade Federal do Ceará – UFC
Campus do Pici, s/n
Fortaleza, Ceará CEP 60451-970
samirchaves@alu.ufc.br

Fernando Trinta

Universidade Federal do Ceará – UFC
Campus do Pici, s/n
Fortaleza, Ceará CEP 60451-970
fernando.trinta@dc.ufc.br

ABSTRACT

The development of Location Based Mobile Games (LBMG) is a complex task. To overcome this problem, authoring tools have been proposed aiming at reducing the time and effort required to create these applications. One of these tools is LAGARTO, a web authoring application to help non-programmers users to create multiplayer LBMGs with augmented reality features. The first version of LAGARTO presented a usability level below end-users expectations. Therefore, this research proposes a new usability project for the LAGARTO editor to address its current problems. This document presents the results already obtained from a work in progress, with promising results of improvement in the usability of LAGARTO.

KEYWORDS

Location-Based Games, Game Modelling, Authoring Tools

1 INTRODUÇÃO

A nova era de entretenimento e interação dos jogos pervasivos, traz à tona novas possibilidades de conversação entre a tecnologia e o mundo real na experiência dos usuários, porém com uma ainda complexa metodologia de desenvolvimento. Sucessos de títulos como Pokemon Go e Ingress mostraram o potencial atrativo que jogos baseados em localização e com recursos de realidade aumentada podem trazer para diversas áreas, como por exemplo, turismo ou educação [3] [5].

Porém, a criação destas aplicações ainda demanda um grande esforço de diversos profissionais envolvidos em diversas tarefas de seu processo de desenvolvimento, como projeto do jogo, programação e testes. Neste sentido, ferramentas de autoria assumem um papel importante, uma vez que seu objetivo está em facilitar a criação de aplicações. Uma dessas ferramentas é a LAGARTO[4], uma proposta acadêmica para autoria de Jogos Móveis Baseados em Localização (JMBL), desenvolvida na Universidade Federal do Ceará como projeto de mestrado. A LAGARTO tem como objetivo maior permitir que pessoas sem conhecimento prévio de programação ou habilidades de desenvolvimento de aplicativos possam projetar e implementar JMBLs de forma facilitada.

No projeto original da LAGARTO, a interação de autores com a ferramenta é feito por meio da aplicação web chamada Editor, responsável tanto pela criação quanto pelo gerenciamento dos jogos planejados. O editor original propôs uma notação visual e uso de certas tecnologias que não permitiram uma boa usabilidade

aos usuários da ferramenta, principalmente devido ao uso *plugins* pré-estabelecidos, que impuseram um modelo de diagramação não intuitivo aos usuários. Tendo em vista este problema, foi elaborado um projeto de pesquisa de iniciação científica que se propôs a estudar e propor uma nova interface e novos modos de interação para designers de jogos com a LAGARTO.

Este documento apresenta os resultados obtidos até agora no projeto. Para isso, na seção seguinte é apresentada uma visão geral da LAGARTO. Na seção 3 é reportada a metodologia aplicada neste projeto de pesquisa. A quarta seção apresenta os resultados até obtidos com a reformulação da interface do editor da LAGARTO. Por fim, a última seção conclui o documento e apresenta próximos passos desta pesquisa.

2 LAGARTO

LAGARTO (*LocAction Games AuthoRing Tool*) [4] é uma ferramenta de autoria para JMBLs que busca jogos móveis baseados em localização que busca abstrair a complexidade de diversos aspectos do desenvolvimento de softwares móveis e sensíveis à localização. A ferramenta faz com que o usuário exerça a função de *game designer*, elaborando a lógica do jogo de forma visual por meio de um editor web. A estrutura deste jogo é recuperada e executada em *smartphone*, por meio de um aplicativo aplicativo interpretador. Além de permitir a autoria dos jogos, a LAGARTO também gerencia o ciclo de vida dos jogos em execução. Este ciclo inclui a disponibilização do jogos para os jogadores, atualização de jogos e interrupção de funcionamento do servidor.

Os jogos projetados na LAGARTO seguem um modelo de aplicação definido por [2], onde cada jogo é visto como um conjunto de missões a serem realizadas por um jogador ou grupo de jogadores. Cada missão é composta por uma ou mais mecânicas comuns à JMBLs[1], como deslocamento à um certo ponto do mapa, captura de objetos virtuais, dentre outras ações. Missões podem ser compartilhadas entre grupos de usuários, criando a ideia de colaboração entre equipes. Também é possível estabelecer prioridades entre missões e mecânicas, indicando se determinadas ações precisam ser realizadas antes que outras. Na sua versão inicial, a interface da ferramenta foi criada conforme figura 1.

Apesar de atender a criação dos jogos de forma visual, a interface de interação proposta para primeira versão da LAGARTO apresentou problemas. Os resultados obtidos a partir de testes de usabilidade revelaram que a LAGARTO possui apenas usabilidade considerada mediana em relação à outros softwares, e que alguns dos recursos da linguagem ainda confundem ou não são disponibilizados intuitivamente para os usuários. Os testes realizados com usuários não programadores foram essenciais para realizar o diagnóstico dos problemas de usabilidade, onde a busca por soluções a estes problemas é de fundamental importância para a aceitação

In: XV Workshop de Trabalhos de Iniciação Científica (WTIC 2018), Salvador, Brasil. Anais do XXIV Simpósio Brasileiro de Sistemas Multimídia e Web: Workshops e Pósteres. Porto Alegre: Sociedade Brasileira de Computação, 2018.

© 2018 SBC – Sociedade Brasileira de Computação.
ISBN 978-85-7669-435-9.

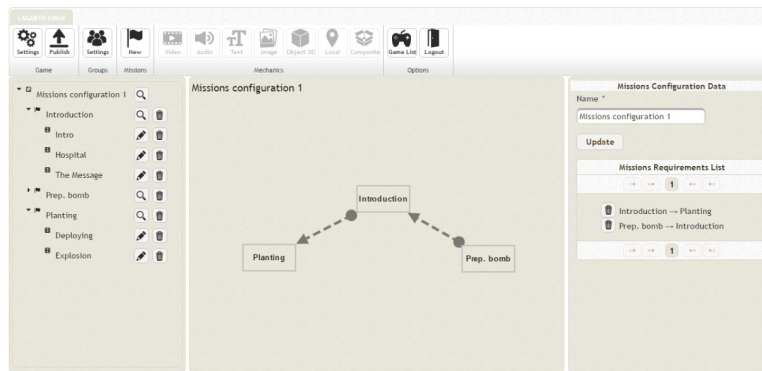


Figura 1: Componentes do Editor - LAGARTO.

e popularização da ferramenta. Dentre os problemas citados nos experimentos podemos destacar (i) a insegurança na realização de ações como alternar entre mecânicas e missões, (ii) dúvidas em relação ao significado de alguns ícones e (iii) confusão quando da implementação de recursos específicos relacionados às mídias suportadas pela ferramenta.

A melhor forma de solucionar estas questões parte pela realização de um estudo de usabilidade mais detalhado, visando implementar uma reformulação da interface gráfica da LAGARTO. Espera-se que a inclusão de recursos como arrastar-soltar (do inglês, *Drag and Drop*), a reorganização das funções da ferramenta e a remodelagem de ícones proporcionarão maior facilidade de uso da ferramenta. Consequentemente, estas mudanças farão com que os usuários sintam-se mais confiantes durante o desenvolvimento dos jogos, gerando benefícios a todos os usuários, pesquisadores e professores que utilizam a LAGARTO.

3 METODOLOGIA DA PESQUISA

A partir da detecção dos problemas de usabilidade da LAGARTO, foi elaborado um projeto de pesquisa de iniciação científica com objetivo geral de definir e implementar um novo projeto da interface gráfica do editor da LAGARTO. Para isso, o projeto (ainda em andamento) prevê a realização de cinco etapas: (i) Estudos sobre ferramentas de autoria, (ii) Estudo sobre a primeira versão da LAGARTO, (iii) Elaboração do projeto da nova interface do Editor, (iv) Implementação da nova interface em uma tecnologia adequada e (v) Testes com usuários finais. Atualmente, o projeto encontra-se na quarta etapa, com previsão de conclusão para final deste ano.

O primeiro passo foi realizado com a leitura do material existente (dissertação de mestrado e artigos científicos) sobre a própria LAGARTO, que traziam uma série de ferramentas de autoria e demais conceitos relacionados ao tema de jogos móveis baseados em localização. Destes trabalhos foram coletadas boa parte de trabalhos relacionados que serviram de base de estudos para características interessantes para a interface gráfica do Editor. Além disso, outros trabalhos também foram selecionados por meio de buscas na Internet. Dentre eles, destaca-se a TaleBlazer¹ do MIT, aplicação de objetivo bastante semelhante ao LAGARTO, porém com uma

implementação diferente dos conceitos que compõem um jogo. Os elementos inspirados no TaleBlazer são relacionados ao layout. Por exemplo, a todo instante um menu flutuante se encontra sobre a tela com opções gerais sobre o jogo. Esta abordagem foi tida como positiva, permitindo expandir a área de visualização do jogo. Além disso, uma grande área da janela de visualização é dedicada ao mapa que suporta o jogo, o que proporciona uma melhor usabilidade já que há um foco nessa região. Por fim, a inserção de um menu de contexto também foi bem avaliada. Este menu apresenta a lista de opções ao se clicar com o botão direito sobre a região do mapa, mostrando funções disponíveis relacionadas ao contexto daquele local.

Em relação à segunda etapa, o estudo sobre a proposta inicial da ferramenta indicou que o layout existente no Editor contava com tema relativamente rústico, efeito trazido tanto por elementos como fontes e ícones escolhidos, como pela aplicação, em diversos momentos, de texturas à certas áreas. Atualmente, essa abordagem não é tida como mais adequada para aplicações semelhantes à LAGARTO. Essa informação passada visualmente, por mais subjetiva que seja, leva a conquista ou não de novos usuários, sendo assim de suma importância para a popularidade da aplicação. Além disso, a organização dos elementos da página tornava o foco central do usuário destoante, comprometendo assim a usabilidade.

A maior parte da navegação baseava-se em painéis fixos que ocupavam uma parcela considerável da janela de trabalho do editor. Este recurso foi tido como um problema, pois não deixava claro, pela quantidade de opções disponíveis, qual o passo seguinte a ser tomado em meio a criação ou edição de um jogo. Outro ponto importante problemático foi em relação à modelagem do jogo, que era realizada em uma área acessível pelos painéis. Como um jogo é constituído de missões, cada missão possui sua própria modelagem, o que obriga a controlar qual missão aparece por vez na tela, trazendo um esforço maior quando o objetivo é ter um panorama geral da organização do jogo. Ainda na questão da organização das missões, estas são formadas por mecânicas, que por sua vez representam diretamente locais geográficos reais. O fato das mecânicas serem vistas em um painel opaco, sem informação palpável de onde se localizam, também abre margem a melhorias. Por fim, um problema também identificado foi o uso exagerado de janelas

¹<http://www.taleblazer.org/>

modais para interação do usuário. Estas janelas não são um recurso tão utilizados em projetos mais modernos de interfaces web.

Atualmente, esta pesquisa se encontra nas terceira e quarta etapas do projeto originalmente proposto. A partir de um processo de implementação iterativo e incremental, uma nova interface para o Editor da LAGARTO vem sendo construído, com ideias de melhoria sendo propostas à cada nova iteração. No início do processo de implementação, um rascunho com a proposta para a nova interface do usuário foi proposta. Esta interface foi inicialmente prototipada em papel para estabelecimento de princípios básicos a serem seguidos. Estes princípios incluíram manter um visual limpo da ferramenta, fazer uso extensivo de ações *drag-and-drop* e maximizar a visão do mapa do jogo pelo usuário. A seção a seguir detalha melhor os resultados até obtidos nesta etapa.

4 EDITOR 2.0 - LAGARTO

Atualmente, o Editor está sendo redesenhado de uma forma a preservar a essência e funcionalidades principais da versão original, contudo, trazendo um novo esquema de usabilidade. Essas modificações prezam por compor uma maior dinamicidade ao manejo, por meio de uma interface moderna e livre de elementos poluentes e distrativos. A Figura 2 apresenta a nova proposta da Ferramenta.

Após o desenho de alguns esboços do layout da ferramenta, foi escolhido um a ser implementado. Este prezava pela mesma ideia do Google Maps, com o elemento mapa ocupando praticamente toda a tela, porém com um sutil menu lateral. Além disso, é observado a presença de painéis arrastáveis pelos quais serão configurados os elementos dos jogos. Também existe um botão com a função de adicionar componentes dos jogos (times, missões e mecânicas) ao contexto em questão. A personalização trazida pelo posicionamento da janela, aliado à simplicidade dos elementos, e a visão panorâmica do mapa compõem a ambientação desejada. Ainda se tratando do esboço selecionado, nele já é possível notar a estilização trazida aos componentes. A iconografia com um traço moderno, o uso de leves sombras para a sensação de sobreposição, a organização de espaçamento e alinhamento dos objetos, entre outras coisas, carregam juntas a configuração da proposta visual mais limpa.

A nova versão do editor está sendo implementada com uso intensivo de ReactJS e da GoogleMaps API. O ReactJS é um framework Javascript voltado para criação de interfaces web criado pelo Facebook. A sua escolha foi graças (i) sua grande aceitação pela comunidade de desenvolvedores, (ii) sua rápida curva de aprendizado, (iii) seu ambiente de desenvolvimento simples e prático, e por, principalmente, (iv) ter a propriedade de tratar componentes web isoladamente, o que gera uma organização instantânea do código fonte e desenvolvimento mais rápido e robusto. Outra questão importante é a facilidade de integrar-se com o padrão Redux, que permite armazenar um estado global da aplicação e emitir eventos de alteração deste estado, o que traz uma montagem ideal para um webapp. É pretendido que este estado seja posteriormente usado pelo servidor da aplicação para seu armazenamento em banco de dados. Por tornar o código mais reusável, o ReactJS permitiu a mais prática criação dos componentes criados pelo esboço.

Já a GoogleMaps API permite a interação do usuário a todo o tempo com um mapa, definindo posições reais de seu interesse, e modelando toda a dinâmica de seu jogo sobre o local escolhido para

suportá-lo. Esta API é majoritariamente utilizada no projeto, desde a criação do mapa até a função de posicionamento dos marcadores das mecânicas. O objetivo é que praticamente toda a gestão do jogo seja realizada no próprio mapa. Para colocar isso em prática, alguns objetos foram criados. O primeiro deles é o *Marker*, o qual representa uma mecânica no modelo de aplicação da LAGARTO. Este se encontra no local do mapa onde a mecânica deve ser realizada e pode ser reposicionado a qualquer momento. A modelagem de prioridade e ordem de execução das mecânicas dentro de cada missão é feita através de arestas que conectam os *markers*.

Como a API não oferece uma estrutura de dados suficiente para atender a esse propósito, foi criada uma própria. Uma inovação em relação à interface original da LAGARTO, é que foi incorporada uma animação à aresta indicando que a partir de um *marker*, qual o próximo local a ser visitado. Esta era uma das principais queixas dos usuários que participaram do experimentos com a versão original do editor, pois a forma anterior não deixava claro qual mecânica ou missão precedia a outra. Cada aresta de uma missão é representada por uma *Polyline*, o segundo objeto a ser usado. O conjunto de *polylines* e mecânicas pertencentes a uma missão são formados por uma estrutura de dados na forma de um Grafo Direcionado Ponderado, permitindo armazenar a posição das mecânicas, as suas prioridades e a ordem de sua execução. O recurso de *context menu* também é utilizado para fornecer opções dos marcadores e arestas de maneira mais rápida.

4.1 Componentes do Editor

O nova versão do editor da LAGARTO possui 5 componentes principais, pelos quais, além do mapa, se dará toda a edição do jogo. As subseções a seguir descrevem cada um destes elementos.

4.1.1 Lista de Times. Na hierarquia dos componentes de um jogo, um time é o de nível mais alto, pois este é constituído por missões que por sua vez possuem mecânicas. O painel que os lista é um painel de posicionamento fixo responsável pela criação de novos times e, além disso, sua seleção para modificação das respectivas propriedades. Cada time criado possui uma cor de identificação, a qual está presente também nos marcadores de mecânicas que o compõem, com o intuito de facilitar sua identificação em meio ao mapa. Posteriormente, pretende-se adicionar funcionalidades de remoção de times e renomeio ao duplo clique a esse componente.

4.1.2 Lista de Missões. Neste menu móvel é possível visualizar toda a hierarquia do jogo. Por meio de listas retráteis, têm-se, respectivamente, os times, as missões e mecânicas organizadas de forma a tornar a árvore hierárquica intuitiva. Nele também é possível selecionar cada um desses elementos. Ao selecionar um time, há duas modificações aparentes na visão do usuário: todas as missões presentes no mapa que não o pertencem aumentam sua transparência, destacando assim as que o pertencem, e o painel de propriedades mostra as informações a ele referentes. Ao selecionar uma missão, acontece coisa semelhante: todas as mecânicas que não a constituem diminuem sua opacidade e o painel de propriedades também mostra as respectivas informações. O mesmo acontece ao selecionar uma mecânica. Em breve, almeja-se adicionar uma barra de ferramentas a esse menu, com as opções de “Criar Novo”, “Remover” e “Duplicar”, válidas para qualquer elemento selecionado.



Figura 2: Componentes do Editor - LAGARTO.

Além disso, é pretendido que seja possível mover uma missão de um time para outro apenas por meio de um *drag and drop*.

4.1.3 Painel de Propriedades. Ao selecionar um, e apenas um, item do jogo, o painel de propriedade mudará seu conteúdo de acordo com o tipo do componente selecionado, contendo suas configurações particulares. Atualmente, a única informação comum a todos é o nome descritivo. No caso de um time, suas propriedades são: nome, cor e lista de missões que o compõem. Em uma missão estão presentes o nome, o time a qual pertence e as mecânicas que a constituem. No caso da mecânica, apresentam-se nome, missão a qual pertence, tipo e localização. Ao se modificar uma informação, esta mudança é populada automaticamente em toda a aplicação tanto em dados armazenados em *cache* quanto na própria visão do usuário, permitindo ter um compreensão imediata do acabou de editar.

4.1.4 Menu de Adição de Mecânicas. Estando a todo momento sobre o mapa, este menu móvel conta com uma lista de ícones, cada um representando um tipo de mecânica a ser adicionada sendo, respectivamente, vídeo, áudio, texto, imagem, localização e composta. Este painel é, por padrão, desabilitado, estando transparente nesse estado. Para habilitá-lo, basta que uma única missão de qualquer seja selecionada. Com isso, será possível escolher o tipo a ser criado, ao clicar sobre algum, todos os outros painéis desaparecerão e espera-se que o mapa seja clicado no local que pretende-se adicionar o *marker* referente a nova mecânica como integrante da missão selecionada. Ao fazer isso e adicioná-la, ela estará disponível para fazer parte do jogo.

4.1.5 Botão de Adição. Este botão é voltado unicamente a função de adição dos novos elementos. Foi pensado de modo a proporcionar um local central onde o usuário tivesse o acesso rápido a esta funcionalidade, por mais que ela se encontre presente em outros locais.

4.1.6 Menu Lateral. O menu retrátil lateral contém opções gerais do jogo. Atualmente há: “configurações”, onde o game designer poderá optar pelo intervalo de salvamento automático, por exemplo; “publicação”, função a qual o jogo poderá ser publicado para

ser acessado pelo clientes móveis; “games”, página onde o usuário poderá ver a lista de jogos criados; e “Novo Jogo”, para a criação de um novo jogo desde o início.

5 CONCLUSÕES E TRABALHOS FUTUROS

Jogos Móveis Baseados em localização já se mostraram aplicações bastante atrativas ao longo dos últimos anos, com diversos títulos tendo alcançado um sucesso global. Estas aplicações tem potencial de alavancar outras áreas que não apenas a do entretenimento, como por exemplo, o turismo, esporte de aventura dentre outros.

A LAGARTO se insere neste contexto ao se apresentar como uma abordagem para que tais jogos possam se construídos por pessoas sem profundo conhecimento de desenvolvimento de aplicações e tecnologia. A partir de um estudo sobre as deficiências encontradas na primeira versão do editor de jogos, esta pesquisa de iniciação científica tem avançado em um projeto de refatoramento da interface gráfica deste editor. Os resultados apresentados já mostram uma interface mais limpa e concisa de ações possíveis para editores de jogos.

A conclusão desta pesquisa se dará com a finalização da ferramenta, e a realização de testes com usuários por meio de experimentos controlados para coleta da experiência destes usuários na nova versão do Editor.

REFERÊNCIAS

- [1] L.A. Lehmann. 2012. *Location-Based Mobile Games*. GRIN Verlag, Munich, Germany.
- [2] Carleandro Noletto. 2015. Uma ferramenta de autoria para o desenvolvimento de jogos móveis baseados em localização com realidade aumentada. (2015).
- [3] Janne Paavilainen, Hannu Korhonen, Kati Alha, Jaakko Stenros, Elina Koskinen, and Frans Mayra. 2017. The Pokémon GO Experience: A Location-Based Augmented Reality Mobile Game Goes Mainstream. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems (CHI '17)*. ACM, New York, NY, USA, 2493–2498.
- [4] Luís Fernando Maia Silva, Carleandro Noletto, Messias Lima, Cristiane Ferreira, Cláudia Marinho, Windson Viana, and Fernando Trinta. 2017. LAGARTO: A Location based Games AuthoRing Tool enhanced with augmented reality features. *Entertainment Computing* 22 (2017), 3–13.
- [5] Heinrich Söbke, Jannicke Baalsrud Hauge, and Ioana A. Stefan. 2017. Prime Example Ingress Reframing the Pervasive Game Design Framework (PGDF). *International Journal of Serious Games* 4, 2 (June 2017), 39–58. <https://doi.org/10.17083/ijsg.v4i2.182>