

Avaliando o Tardigrade: um framework para o desenvolvimento de jogos de cartas ubíquos

Willian Almeida Rodrigues¹, Cecília Costa^{2*}, Nécio de Lima Veras³,
Vitor Cortez⁴, Ricardo Andrade⁵, Fernando Trinta⁶
Glaudiney M. Mendonça Junior⁷, Windson Viana⁸

¹Universidade Federal do Ceará (UFC)
Campus do Pici - Bloco 942-A
Fortaleza — Ceará — Brasil

w.almeida.w@gmail.com, ceciliaaguilar@great.ufc.br

necio.veras@ifce.edu.br, vitoracortez@gmail.com

ricardoalr@ufpi.edu.br, fernandotrinta@great.ufc.br

glaudiney@virtual.ufc.br, windson@great.ufc.br

Abstract. *Researchers in Ubiquitous Games aim at discovering how computing devices and sensors can promote entertainment. They try to extend game experience beyond the virtual world by extrapolating its boundaries to the player's physical world. These games inherit traditional challenges of both domains (i.e., game development and ubiquitous systems), such as device heterogeneity, resource discovery, communication, and limitation of available resources. In this paper, we propose a framework for the development of ubiquitous card games, called Tardigrade. It seeks to minimize the challenges of implementing this kind of ubiquitous game. We implemented an evaluation experience with eight mobile developers. The results are discussed further in this article.*

Resumo. *Pesquisadores de Jogos Ubíquos buscam descobrir como dispositivos de computação e sensores podem promover entretenimento. Eles tentam estender a experiência do jogo para além do mundo virtual extrapolando seus limites para o mundo físico. Esses jogos herdam desafios tradicionais de ambos os domínios (i.e., desenvolvimento de jogos e sistemas ubíquos), como heterogeneidade, descoberta de recursos, comunicação, e limitação de recursos. Neste artigo, é proposto e avaliado um framework de jogos de cartas ubíquos, chamado Tardigrade. O framework minimiza os desafios de implementar este tipo de jogo. Por fim, o artigo apresenta uma avaliação com oito desenvolvedores de dispositivos móveis e discute os resultados alcançados.*

*Bolsista de mestrado pela Funcap

1. Introdução

Os jogos ubíquos ou pervasivos representam uma nova era para o entretenimento e a interação, estendendo a experiência dos jogos para além do mundo virtual [Silva et al. 2017]. Jogos ubíquos borram a fronteira dos mundos virtual e real do jogador nos quais um ou mais elementos do jogo são influenciados pelos limites espaciais, temporais e sociais do jogador [Montola et al. 2009]. Esses jogos usam sensores, informações contextuais e elementos tangíveis (e.g., lugares do mundo real, brinquedos, cartas) e contemplam um amplo espectro: desde jogos *singleplayer* em smartphones até jogos massivos de realidade mista como o recente sucesso do Pokemon Go [Paavilainen et al. 2017]. Os jogos ubíquos herdam, ao mesmo tempo, desafios tradicionais do desenvolvimento de jogos (e.g., gerenciamento de equipes multidisciplinares, autoração de multimídia) e problemas intrínsecos aos sistemas ubíquos, tais como: a heterogeneidade dos dispositivos, a descoberta de dispositivos, a comunicação volátil e a limitação dos recursos disponíveis.

Esta pesquisa endereça um subtipo de jogos ubíquos focado no aumento da experiência de jogos de mesa (i.e., do inglês *boardgames*). Neste domínio, é possível encontrar alguns trabalhos que relatam aplicações e jogos com características ubíquas [Römer and Domnitcheva 2002] [Sakamoto et al. 2014] [Cheung et al. 2013]. Contudo, esses trabalhos não focam em nenhuma abordagem que promova o reuso ou a redução da complexidade de desenvolvimento, possuindo basicamente o objetivo de construir um ambiente de jogo e a integração com dispositivos distintos. Por exemplo, o trabalho [Barhaug 2017] que faz uso de uma ferramenta com propósito similar ao *framework*, porém seu foco é em técnicas de *design* de jogos híbridos para ensino.

Nesse sentido, esta pesquisa propõe o conceito de jogo de cartas ubíquos, pautado em um mecanismo que auxilia sua utilização por meio de dispositivos móveis. Neste artigo, em especial, o principal objetivo é propor e avaliar um *framework* para auxiliar o desenvolvimento de um tipo de jogo ubíquo que envolve cartas tangíveis (físicas). A pesquisa reflete a materialização de um modelo para criação de jogos de cartas ubíquos, denominado Tardigrade, cujas ideias iniciais são descritas em [Rodrigues et al. 2016]. O *framework* foca em minimizar o esforço de implementação deste tipo de jogo na plataforma Android. Uma das contribuições diretas obtidas a partir desse tipo de aplicação é o aumento da experiência de um jogo analógico com aplicativos móveis para reconhecer cartas, apresentar o estado do tabuleiro ou mostrar animações digitais. Estes podem ser vistos como exemplos da materialização dos princípios de Computação Ubíqua no domínio dos jogos de cartas, tendo em vista que a Computação Ubíqua pode incorporar dispositivos computacionais em objetos comuns, de uma forma que a interação com o usuário se torna natural e quase imperceptível [Weiser 1999].

2. Cenário Motivador

”Com seu baralho de cartas colecionáveis no bolso, David caminha em um shopping de Fortaleza. Seu Android vibra anunciando que existe um oponente nas proximidades dele, Jack. David começa o combate juntando uma combinação de cartas na parte de trás do seu aparelho, fazendo Jack contra-atacar colocando também uma carta e começando a balançar seu celular com afinco. David na verdade utilizou uma série de cartas de ataque, e Jack utilizou uma carta de defesa que tem seu resultado aprimorado com o movimento de chacoalhar o aparelho. Os ruídos do shopping permitiram Jack contra-atacar utilizando uma das suas cartas que aumenta o ataque de acordo com o ruído do local.

David nota que a partida está perdida, porém decide continuar defendendo-se do combate cobrindo os sensores de iluminação e proximidade com o intuito de impedir os danos do próximo ataque. Como recompensa, David deveria entregar uma de suas cartas usadas em combate a Jack. David entrega a carta que já não o pertence mais e parabeniza Jack pela vitória, mas deixando claro que voltará para reconquistar a carta perdida. Durante a batalha, informações de contato dos jogadores foram trocadas, permitindo com que David envie mensagens para Jack e, assim, possa ter a chance de uma revanche.”

O cenário motivador apresenta mecânicas de combate baseadas em cartas tangíveis e um aplicativo de combates virtuais. Cartas são reconhecidas multi ou unicamente por proximidade pelos dispositivos, que reagem de acordo com os efeitos de cada carta. Além disso, ele incorpora o conceito de cartas sensíveis ao contexto apresentado em [Rodrigues et al. 2016]. Essas cartas necessitam de informações contextuais (i.e., que descrevem a situação do jogador) para serem bem-sucedidas, como aumentar o dano baseado no clima do local ou no nível de ruído onde o combate está sendo realizado. Já outras cartas utilizam sensores, como o de proximidade, iluminação e acelerômetro, para terem seus efeitos maximizados ou minimizados. Existem ainda cartas que são apenas identificadas pela aplicação para que seus efeitos sejam aplicados. Essas e outras funcionalidades foram implementadas no *framework* descrito a seguir.

3. Tardigrade

O *framework* Tardigrade tem como propósito disponibilizar uma forma de desenvolver jogos de cartas ubíquos que apresentem interação do real com o virtual, podendo ou não fazer uso do contexto do jogador como componente de jogo (e.g., localização física). O *framework* proporciona a detecção das cartas, a associação delas ao estado do jogo virtual, a captura do contexto do jogador e o compartilhamento dessas informações entre múltiplos dispositivos. O modelo base do Tardigrade foi projetado para três possíveis situações de jogo. Na primeira situação, tem-se um único dispositivo móvel sendo compartilhado por todos os jogadores. Uma segunda situação, que representa uma variação da anterior, indica que mais de um dispositivo permitiria acessar o estado de jogo a partir de alguma referência compartilhada fisicamente (e.g., Qr Code). Ampliando o conceito de compartilhamento da segunda situação, tem-se uma terceira visão de jogos suportados pelo modelo, que é composta por múltiplos dispositivos conectados em rede. Nessa situação, os dispositivos podem compartilhar e trocar o estado de jogo e compartilhar eventos detectados em cada dispositivo de forma individual, ou seja, cada jogador tem uma visão distinta do jogo.

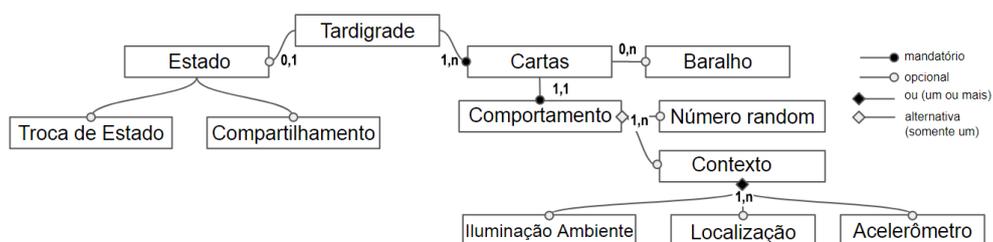


Figura 1. Diagrama de *Features* do Modelo de Jogos do Tardigrade.

No diagrama, as ligações terminadas em círculos preenchidos representam relações mandatórias. Dessa forma pode-se observar que um jogo de cartas ubíquo base-

ado no modelo do Tardigrade tem as seguintes características: **a)** estados que descrevem o jogo, **b)** esses estados podem ser compartilhados e/ou trocados entre diferentes instâncias (jogadores); **c)** todos os jogos devem ter cartas que **d)** podem ser gerenciadas e armazenadas em um baralho; e **e)** essas cartas possuem comportamento de um ou mais tipos.

As visões de sistemas ubíquos [Spínola and Travassos 2012] e de jogos ubíquos [Cheung et al. 2013] foram utilizadas para guiar as características do modelo proposto. O framework Tardigrade e sua documentação associada estão disponíveis no repositório ¹. O *framework* foi desenvolvido para plataforma móvel Android, onde o SDK da referida plataforma serviu de fonte de recursos para as funções providas pelo *framework*. O Tardigrade se caracteriza como um *framework* do tipo caixa cinza, no qual alguns classes já são fornecidas completas e outras precisam ser implementadas ou estendidas para cada jogo instanciado. A Figura 2 ilustra os módulos principais do Tardigrade e suas relações.

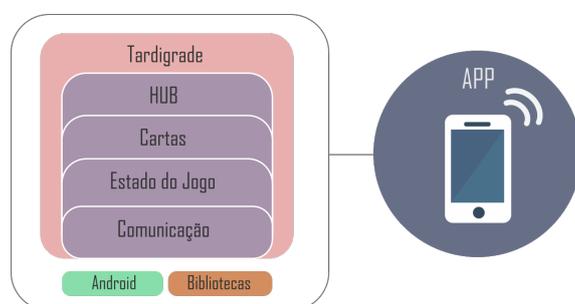


Figura 2. Organização estrutural dos módulos principais do Tardigrade.

3.1. Cartas Ubíquas

O Tardigrade materializa o conceito de jogos de cartas ubíquos proposto em [Cheung et al. 2013] que foi estendido em pesquisa anterior [Rodrigues et al. 2016] por meio de três categorias de cartas descritas a seguir.

Cartas tangíveis-identificáveis são cartas ubíquas que possuem um ou mais elementos de identificação que permitem a passagem de suas propriedades da realidade para a virtualidade. O elemento de identificação pode ser uma etiqueta NFC, um *QR Code* ou um padrão visual. Uma aplicação que reconhece as cartas do baralho de UNO e faz sua transcrição em áudio é um exemplo de uso de cartas tangíveis-identificáveis que pode criar interação de jogadores com deficiência visual com jogo sem a necessidade de modificações físicas.

Cartas tangíveis-interativas são aquelas que, além da identificação, possuem propriedades que variam ou são ativadas de acordo com elementos de interação (e.g., sacudir o celular) ou elementos do mundo digital (e.g., número aleatório dentro de um intervalo). Essas cartas permitem jogos com características híbridas, isto é, jogos analógicos que se misturam com o virtual, como um jogo que abre uma janela de interação onde o jogador precisa sacudir o dispositivo para obter um resultado.

Por fim, as **Cartas sensíveis ao contexto** usam a ideia de que elementos do contexto de um ou mais jogadores podem ser incorporados ao jogo para aumentar, diminuir,

¹<https://github.com/walmeidaw/tardigrade>

ativar ou desativar efeitos de cartas físicas. Por exemplo, em um jogo inspirado no *Yu-gi-oh!*, cartas de polimerização só poderiam ser utilizadas se o jogador estivesse em um contexto específico (e.g., o jogador deve estar sorrindo em uma manhã chuvosa). O dispositivo digital funciona como verificador dessa condição contextual (detecta o sorriso do jogador e checa a meteorologia daquela localização).

Esses três tipos de cartas apresentadas permitem o desenvolvimento de diversos jogos de cartas que podem ser entendidos como ubíquos. Uma das características que permite isso é a tangibilização de componentes reais para o meio virtual, que podem ou não fazer o uso do contexto dos jogadores como componentes do jogo. Um vídeo de execução desses conceitos de carta pode ser acessado em no link².

3.2. Módulo Hub

O Hub funciona como um agregador de informações e funções externas ao *framework*, uma espécie de padrão *Facade* para o acesso a informações (e.g., dados contextuais capturados pela *Context API*). Sua responsabilidade é agrupar métodos de integração, sejam com sensores nativos, plataformas de *middleware* ou bibliotecas externas. Este módulo foi implementado seguindo o padrão *Singleton*, de forma que, com sua instância, é possível acessar o mesmo serviço de diferentes partes da aplicação.

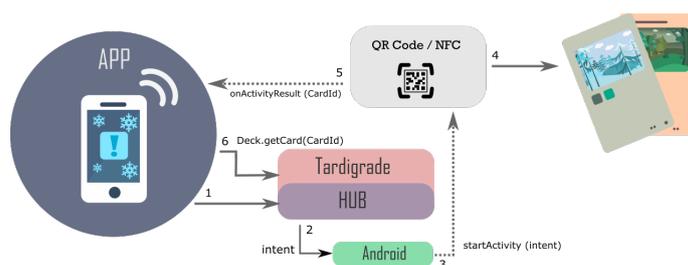


Figura 3. Fluxo do Hub para integração de serviços.

A Figura 3 apresenta o fluxo realizado na leitura de uma carta por *QR Code*. O *Hub* é quem intermedia a interação de leitura de uma carta física. O *Hub* inicia uma intenção de leitura de *QR Code* e a solicitação abre a aplicação que faz a leitura e retorna o resultado para o *Hub* que a repassa para a aplicação. O último passo consiste em tratar o resultado obtido para recuperar as informações da carta identificada. Uma outra alternativa permite a leitura de cartas de forma mais simples, utilizando etiquetas NFC, permitindo que as cartas sejam identificadas apenas aproximando-as de dispositivos compatíveis com a tecnologia.

3.3. Módulo de Cartas

O módulo de cartas é composto pelas classes *Deck* e *Card*. A classe *Deck* representa o baralho e é responsável por agregar e gerenciar um conjunto de objetos *Card*, podendo fornecer informações desses objetos. O *Tardigrade* pode suportar três tipos de cartas com comportamentos e utilidades distintas descritos anteriormente. Todas essas cartas podem ter efeitos de aumento de realidade, revelando sons, animações e interações gráficas que tornem a experiência de jogo mais imersiva. O comportamento dessas cartas

²<https://goo.gl/dccWSJ>

é encapsulado na classe `Card` seguindo o padrão de projeto *Command*. A implementação desse comportamento é feita no método `execute()` do objeto `Card`. O comportamento ubíquo e/ou interativo tem o `Hub` como fonte, onde também acontece a interação com uma plataforma de *middleware*, biblioteca externa, sensores, ou com o serviço responsável por adquirir o contexto.

3.4. Módulo de Estado de Jogo e Módulo de Comunicação

O estado de um jogo no Tardigrade representa a memória do andamento e do progresso do jogo. Esse estado descreve informações do cenário, dos objetos envolvidos e dos jogadores em um dado momento do jogo. O estado possui uma estrutura única para as regras de cada jogo, ficando a cargo do desenvolvedor do jogo explicitar suas peculiaridades ao usar o *framework*. Já o Módulo de Comunicação é responsável por descobrir, conectar e gerenciar a troca de mensagens entre dispositivos conectados. Esse módulo é composto por quatro componentes: a classe `Network` que ativa um canal de recebimento de mensagens; a classe `Discovery` que implementa um serviço de rede e a descoberta de outros dispositivos que possuam o mesmo serviço ativado (integrando-se com a classe `Network` para criar uma conexão entre dispositivos); cada conexão é representada por um objeto `Channel` que envia mensagens para o canal de rede correspondente; e a classe `Manager` que implementa o padrão *Observer* e envia mensagens a um grupo de canais observadores deste objeto. Nos eventos em que um dispositivo é descoberto ou perdido, o conteúdo do objeto de mensagem é um objeto `Channel` que pode ser utilizado para manter comunicação com o dispositivo encontrado. O serviço de descoberta disponível no *framework* limita-se a encontrar dispositivos que estejam conectados numa mesma WLAN.

4. Avaliação Inicial com Desenvolvedores

Foi realizada uma avaliação inicial do *framework* Tardigrade e suas funcionalidades para o desenvolvimento de jogos de cartas ubíquos, tendo como foco verificar se desenvolvedores com alguma experiência em aplicações Android conseguiriam utilizar o *framework* e os seus componentes disponíveis com facilidade. O experimento de avaliação foi inspirado nas metodologias de avaliação de usabilidade propostas por [Brad A. Myers 2016]

4.1. Perfil dos Usuários

Foi escolhido de forma não probabilística, e por conveniência, um grupo de oito voluntários (um graduando, cinco mestrandos e dois doutorandos) com experiência prévia em desenvolvimento Android, onde apenas um tinha algum conhecimento no desenvolvimento de jogos digitais. A Figura 4 apresenta mais informações sobre o perfil do grupo de avaliadores. Sobre o perfil de programador dos avaliadores, todos reportaram ter conhecimento relevante em programação Java e desenvolvimento de aplicações Android.

4.2. Procedimento

Para cada um dos voluntários, foi solicitado executar uma sequência de atividades e o preenchimento de formulários de avaliação. O tempo que cada um dos voluntários levou para realizar as tarefas foi coletado durante cada avaliação.

A aplicação do experimento foi dividida em quatro módulos. Cada um deles consistia na capacitação sobre as principais funcionalidades do *framework* seguido por uma

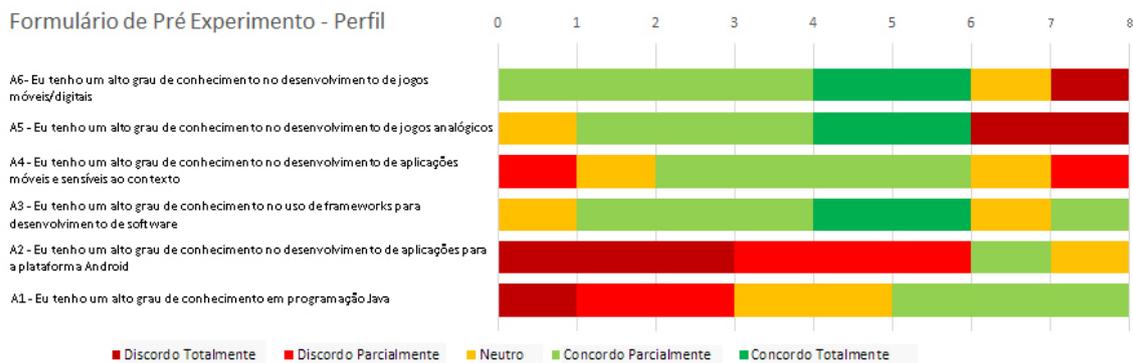


Figura 4. Perfil dos voluntários.

atividade relacionada às características elencadas no módulo em avaliação. As atividades consistiam em codificar os módulos conforme as instruções, sendo que a atividade era considerada como concluída quando o voluntário autodeclarava que as instruções dadas tinham sido codificadas por completo. O conjunto realizado tinha como objetivo criar uma aplicação simples que fizesse uso dos componentes do *framework*.

Ao fim de cada atividade, o comportamento esperado do jogo implementado era validado como relatado a seguir. A **Atividade 1** deveria reconhecer uma carta e apresentar seu valor na interface. A **Atividade 2** deveria reconhecer uma carta identificável, uma carta interativa e uma carta sensível ao contexto, apresentando o resultado baseado na interação realizada na interface. A **Atividade 3** deveria somar o valor das cartas reconhecidas ao estado do jogo e apresentar a soma na tela. Por fim, a **Atividade 4** deveria realizar a descoberta de um segundo dispositivo, conectar-se a ele e trocar informações de cartas reconhecidas, devendo apresentar a soma das cartas em ambos os dispositivos.

4.3. Materiais e Instrumentos

Durante o experimento, foram utilizados seis formulários digitais³ criados no *Google Forms* para realizar a coleta das informações durante as etapas da avaliação.

O formulário de pré-experimento tinha como objetivo coletar informações sobre o perfil dos voluntários. Ao fim de cada atividade, era aplicado um formulário de acompanhamento com o objetivo de saber quanto o voluntário compreendeu da atividade e das dificuldades encontradas. O formulário pós-experimento tinha como objetivo coletar informações sobre a utilização do *framework* e foi inspirado no SUS (*System Usability Scale*) [Brooke 2013]. Os desenvolvedores deviam responder usando a escala Likert de cinco níveis: “discordo totalmente”, “discordo parcialmente”, “indiferente” ou “neutro”, “concordo parcialmente” e “concordo totalmente”. O objetivo desse formulário é obter uma avaliação preliminar a cerca do uso do *framework*. Para o experimento, foi criado um documento de capacitação dos voluntários para o uso do *framework*. O documento⁴ apresenta o conceito de jogos de cartas ubíquos, a utilização das funcionalidades do *framework* e as atividades a serem realizadas. Para desenvolver as atividades, um ambiente de desenvolvimento no *Android Studio* foi preparado com um projeto base já configurado. Esse projeto consistia no *framework* Tardigrade já implantado com um projeto de

³Os formulários estão disponíveis em <https://sites.google.com/site/avaliacaotardigrade/>

⁴O documento está disponível em <https://sites.google.com/site/avaliacaotardigrade/>

interface já pronta. Assim o experimento consistia apenas no que tangia as funcionalidades do *framework*. Os voluntários tinham disponível um dispositivo Android para testar a aplicação ao fim de cada atividade. Na última atividade, relacionada a descoberta de dispositivos, um segundo dispositivo era exigido para completar o teste. Para este fim, foi fornecido um outro dispositivo, com uma versão completa e funcional da aplicação já previamente instalada.

4.4. Resultados e Discussão

Os voluntários de 1 ao 6 executaram cada tarefa em duas etapas, uma de capacitação e outra para desenvolver a atividade. Os voluntários tiveram o maior gasto de tempo na capacitação do primeiro módulo (média de 10 minutos), que incluía também uma introdução aos jogos ubíquos, uma visão geral do *framework* para então abordar o módulo de reconhecimento de cartas de forma detalhada. Nos demais módulos, a média foi de quatro minutos e meio para cada um dos três módulos finais. Já para desenvolver as instruções das atividades, eles gastaram em média 11 minutos e meio.

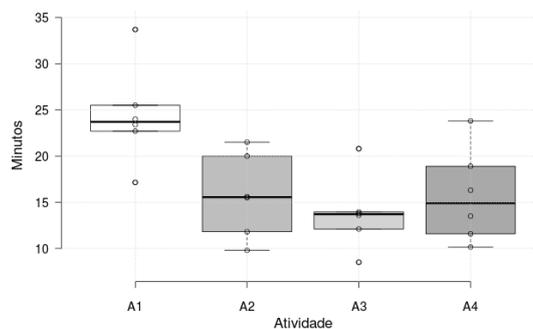


Figura 5. Medições da primeira sessão (voluntários 1 à 6)

O experimento foi feito com mais quatro voluntários. Apenas dois, considerados como voluntários 7 e 8, para fins de análise, puderam concluir o experimento. Os demais tiveram de se ausentar devido compromissos. Essa sessão do experimento foi feita de forma conjunta e sem uma etapa de capacitação separada para cada módulo. Os voluntários receberam o mesmo documento com a descrição do *framework*, módulos e atividades. O tempo total para a conclusão foi superior ao dos demais voluntários. Assim como nos demais testes, foi observado um tempo maior na execução da tarefa referente ao primeiro módulo. Uma voluntária que não pode concluir o experimento sofreu com atrasos adicionais para a conclusão do módulo 1 devido à problemas no ambiente de desenvolvimento. Ela foi capaz de executar a tarefa do módulo 2 em apenas 11 minutos, tempo inferior ao de todos os demais candidatos. Para os 3 módulos restantes, a média foi de 29 minutos. Foi possível avaliar o tempo de execução da tarefa sem uma etapa prévia de capacitação. Para concluir cada módulo com sucesso, era necessário que o voluntário extraísse as informações pertinentes à atividade, da descrição do módulo respectivo.

O primeiro módulo tinha o objetivo de avaliar o reconhecimento de cartas. Apesar de cinco dos voluntários afirmarem terem compreendido a atividade, sete deles apresentaram problemas no desenvolvimento deste módulo, principalmente em relação ao uso da classe `Card`. Já o voluntário 4 não teve dificuldades em implementar a atividade e não apresentou problemas na compilação. Ele teve um tempo total de 17 minutos no módulo

1, uma vez que o tempo médio dos voluntários 1 ao 6 foi de 24 minutos. Os voluntários 7 e 8 tiveram um tempo médio de 45 minutos para finalizar o primeiro módulo.

O segundo módulo tinha como foco os diferentes comportamentos das cartas reconhecidas. Dentre os voluntários 1 à 6, a média de tempo finalização do módulo foi de 16 minutos. Para os voluntários 7 e 8, essa média foi de 30 minutos. Diferente da primeira atividade, os voluntários não tiveram problemas para executar as instruções que desenvolviam o comportamento das cartas, com exceção do voluntário 7 que informou não ter compreendido o que devia ser feito no passo a do item 1, os voluntário 2, 3 e 6 tiveram mais dificuldades na abstração das instruções e precisaram de orientações durante esta etapa do experimento, necessitando de mais tempo para concluí-la. Os voluntários 1, 4 e 5 levaram aproximadamente metade do tempo para finalizar as mesmas instruções.

O módulo 3 desenvolveu um estado de jogo e a interação de cartas com esse estado. Com tempo médio total para os voluntários 1 à 6 de 13 minutos e de 26 minutos para os voluntários 7 e 8, todos tiveram sucesso ao realizar todas as instruções. O voluntário 4 apresentando novamente um tempo consideravelmente inferior aos demais (apenas oito minutos e 27 segundos).

O quarto módulo tinha como propósito apresentar a implementação da comunicação. Neste módulo, os voluntários 1 à 6 demonstraram menos dificuldade na compreensão do seu funcionamento. Com uma média de 15 minutos e 46 segundos, todos foram capazes de finalizar a atividade com eficiência. Os voluntários 7 e 8, concluíram em 22 e 40 minutos, respectivamente. Em ambos os casos apenas a parte de descoberta do dispositivo não funcionou como esperado. A tarefa foi dada como concluída devido ao tempo já estendido. A Figura 6 apresenta os resultados obtidos com o formulário de avaliação pós-experimento.

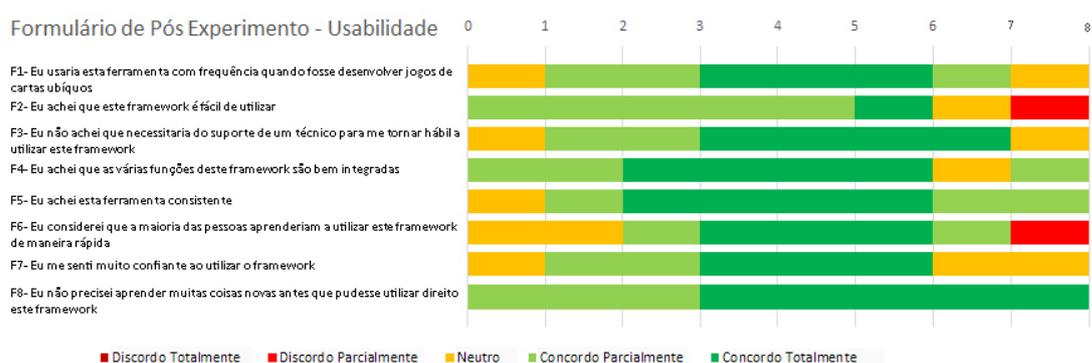


Figura 6. Resultado do questionário pós-experimento.

5. Conclusões e Trabalhos Futuros

Este trabalho apresentou o Tardigrade, um *framework* Android para o desenvolvimento de jogos de carta com características de aplicações ubíquas. A concepção de um *framework* para jogos de carta ubíquos encapsula conhecimento deste domínio e proporciona uma base para o desenvolvimento mais rápido desses jogos. O Tardigrade implementa três tipos de cartas que podem ser utilizadas como mecânicas que envolvem interações com o

mundo real dos jogadores. Um procedimento de experimentação do *framework* foi realizado com um grupo de 8 desenvolvedores para verificar a qualidade das funcionalidades oferecidas, a usabilidade dessas funcionalidades e a facilidade no desenvolvimento de novos jogos ubíquos. Com os resultados da experimentação, foi possível observar que o *framework* foi bem aceito pelos desenvolvedores. No entanto, apesar da avaliação positiva, é importante mencionar que a experimentação foi realizada com um baixo número de desenvolvedores, e que o desenvolvimento de um único jogo de carta ubíquo não é o suficiente para evidenciar a completude do Tardigrade. Além disso, a subjetividade das respostas e a relação próxima entre os candidatos e avaliadores, pode ser um fator de ameaça ou validade da abrangência para os resultados. Para minimizar esse elemento, foram coletados os tempos durante cada atividade. Assim, é possível comparar as impressões coletadas nos formulários com os resultados práticos. Essas limitações estabelecem possíveis problemas a serem explorados em trabalhos futuros.

Referências

- Barhaug, M. (2017). Serious games: improving the learning effect with hybrid games. Master's thesis.
- Brad A. Myers, J. S. (2016). Improving api usability. *Magazine Communications of the ACM*.
- Brooke, J. (2013). Sus: A retrospective. *J. Usability Studies*, 8(2):29–40.
- Cheung, G., Lee, A., Cheng, K., and Lee, H. J. (2013). Dispensable, tweakable, and tangible components: Supporting socially negotiated gameplay. *Games and Culture*.
- Montola, M., Stenros, J., and Waern, A. (2009). *Pervasive Games: Theory and Design*. Morgan Kaufmann Publishers Inc.
- Paavilainen, J., Korhonen, H., Alha, K., Stenros, J., Koskinen, E., and Mayra, F. (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, pages 2493–2498, New York, NY, USA. ACM.
- Rodrigues, W., Mendonça, G., and Viana, W. (2016). Cartas sensíveis ao contexto: Aumentando a experiência em jogos de carta ubíquos. In *SBC – Proceedings of SBGames*, pages 630–638.
- Römer, K. and Domnitcheva, S. (2002). Smart playing cards: A ubiquitous computing game. *Personal Ubiquitous Comput.*, 6(5-6):371–377.
- Sakamoto, M., Alexandrova, T., and Nakajima, T. (2014). Introducing virtuality to enhance game-related physical artifacts. *International Journal of Smart Home*, 8(2):137–152.
- Silva, L. F. M., Noletto, C., Lima, M., Ferreira, C., Marinho, C., Viana, W., and Trinta, F. (2017). LAGARTO: A location based games authoring tool enhanced with augmented reality features. *Entertainment Computing*, 22:3–13.
- Spínola, R. O. and Travassos, G. H. (2012). Towards a framework to characterize ubiquitous software projects. *Inf. Softw. Technol.*, 54(7):759–785.
- Weiser, M. (1999). The computer for the 21st century. *SIGMOBILE Mob. Comput. Commun. Rev.*, 3(3):3–11.