

SOFTWARE DE TRANSMISSÃO SONORA EM TEMPO REAL PARA DISPOSITIVOS MÓVEIS A PARTIR DA CAPTAÇÃO DE SONOPLASTIA AO VIVO

Breno da Costa Silva¹, Paulino Wagner Palheta Viana Ms.C²

¹Engenharia da Computação – Centro de Ensino Superior Fucapi (CESF)
Caixa Postal 69075-351 – Manaus – Am – Brasil

²Engenharia – Centro de Ensino Superior Fucapi (CESF)
dcs.breno@gmail.com, wagnerpalheta@gmail.com

Abstract. *With the advent of technology, the consolidation of social networks and the constant evolution of mobile devices have been a remarkable milestone in the last decade, with individuals increasingly becoming dependent on mobile technologies, resulting in a constant search for innovations and improvements of the handsets. Today, the vast majority of the population uses their mobile devices as real computers, performing various types of services through them - from diverse purchases in online stores, private transportation, games, movies, social media, among others. However, for a mobile device to have these capabilities, software must be designed to fit each segment because the device (hardware) does not work on its own. In this context, there was the opportunity to develop software capable of capturing live sound and at the same time transmitting on users' mobile devices, in order to streamline the way this sound is reproduced.*

Resumo. *Com o advento da tecnologia, a consolidação das redes sociais e a evolução constante dos dispositivos móveis, têm sido de um marco notável na última década, com os indivíduos, cada vez mais, tornando-se dependentes das tecnologias móveis, resultando numa busca constante de inovações e melhorias dos aparelhos. Atualmente, a grande maioria da população utiliza seus dispositivos móveis como verdadeiros computadores, executando diversos tipos de serviços através dos mesmos - desde compras diversas em lojas virtuais, transporte particular, jogos, filmes, mídias sociais entre outros. No entanto, para que um dispositivo móvel tenha essas capacidades, é necessário que exista um software desenvolvido para atender cada segmento, pois o dispositivo (hardware) não funciona sozinho. Nesse contexto, verificou-se a oportunidade de se desenvolver um software capaz de captar a sonoplastia ao vivo e, ao mesmo tempo, transmitir nos dispositivos móveis dos usuários, de modo a dinamizar a forma como esse som é reproduzido.*

1. Introdução

O mundo está se tornando cada vez mais conectado. Nas últimas três décadas, os telefones celulares passaram por diversas mudanças, evoluindo dos imensos terminais com pouca inteligência para pequenos computadores pessoais, também conhecidos

como *smartphones*, que são capazes de transmitir vozes e dados em tempo real. Isso abriu um universo de oportunidades para operadoras, fabricantes e integradores. Com isso iniciou-se o desenvolvimento de um modelo de negócios que abrange desde o entretenimento por meio de jogos, músicas e toques, até serviços de segurança com transmissão de imagens e localização, comércio usando comunicação entre máquinas, propaganda com o SMS, “Serviços de Mensagens Curtas”, e incrementada com o surgimento do MMS, “Serviços de mensagens multimídia”.

Nesse sentido, Araujo (2006) destaca a multiplicidade de recursos que agregam valor a estes aparelhos eletrônicos, os quais não se limitam apenas a fazer ligações telefônicas. Dentre as funções atualmente realizadas pelos *smartphones*, Araujo (2006) destaca: a) manipulação de dados; b) produção e edição de fotos; c) execução de aplicativos, cada um com seu enfoque específico. Além disso, o autor reitera que a capacidade de executar jogos mais simples até aplicativos de controle pessoal fascina os clientes do mundo tecnológico. Os dispositivos móveis se notabilizam pela junção entre as capacidades de processamento e rede, o que sugere ser esta a tendência destes aparelhos no futuro próximo.

O objetivo deste estudo, portanto, é desenvolver um aplicativo para dispositivos móveis, que tenha a capacidade de captar o som proveniente de caixas de som em shows ao vivo e, em tempo real, retransmitir esse som nos aparelhos que detêm o aplicativo.

Para tal, será utilizada a tecnologia para desenvolvimento baseada no framework Ionic. Construído com base na tecnologia padrão da *Web*, o *Ionic* ajuda as equipes a criar e distribuir belos aplicativos híbridos e progressivos da *Web* em várias plataformas usadas por fabricantes de celulares, abrangendo uma maior variedade de aparelhos. Antes, qualquer programa que precisasse ser incluído em celulares deveria ser escrito na linguagem nativa do próprio dispositivo, provavelmente utilizando bibliotecas proprietárias do mesmo. O *Ionic* é independente de plataforma.

2. Ionic

Boniati e Wahlbrinck (2017) consideram que o *Ionic Framework* consiste num SDK de código aberto, cuja principal característica é permitir aos seus usuários a criação de aplicativos móveis através do emprego de tecnologias largamente conhecidas, como por exemplo, HTML, CSS e o *JavaScript*.

Moura e Melo descrevem a sigla SDK como *System Development Kit*, que consistem em pacotes voltados para o desenvolvimento de softwares. E quanto ao *Ionic Framework*, estas autoras ressaltam que através da utilização deste pacote é possível realizar a criação de softwares caracterizados por apresentar elevado desempenho e reconhecidas qualidade. Em adição a isto, Ionic (2018) destaca o uso da base *Apache Cordova*, a qual proporciona aos aplicativos criados uma aparência mais profissional, além de tornar mais simples o seu processo de elaboração e materialização.

Andrade (2019) destaca a questão da conectividade e interatividade deste *framework*, através de ferramentas como: gestos, controles, animações e demais interações. Por sua vez, Ferreira (2019) evidencia a facilidade de manuseio do *Ionic*, o qual pode ser utilizado de maneira autônoma, ou, noutras palavras, sem o uso de uma estrutura *frontend*, por meio de um *script* de inclusão. Outra maneira de utilização do

referido *framework* é através de sua integração com bibliotecas e outras estruturas, como por exemplo, o *Angular*, o qual, segundo Moura e Melo (2017) é um *framework Java Script* cuja principal característica é o auxílio na realização de *single-page applications*, o qual é mantido pela empresa *Google*. Em relação as principais vantagens do *Ionic*, Ferreira (2019) e Lange e Mercado (2019) listam os seus benefícios de utilização conforme abaixo:

- Framework de código aberto e poderoso conjunto de ferramentas e serviços do desenvolvedor, onde permitindo a execução de um aplicativo em qualquer plataforma ou dispositivo, tudo a partir de um único código.
- Continuidade de plataforma, a aparência do seu aplicativo não é de tamanho único.
- Tecnologia é toda baseada na web, você pode até executar o seu aplicativo em um navegador de desktop como um aplicativo da *web*.
- A comunidade de desenvolvedores da web é cerca de 30 vezes maior em tamanho do que o número de desenvolvedores de aplicativos móveis nativos.
- Os usuários têm uma ótima experiência através de plataformas e dispositivos, e o desenvolvedor tem um único código para se preocupar.
- Velocidade, construir para múltiplas plataformas a partir de uma única base de código, o processo é geralmente 2 a 3 vezes mais rápido do que o nativo.
- Eficiência estima-se que o híbrido pode diminuir de 75-80% em custos de suporte e portabilidade em comparação com o nativo.
- Aplicativos híbridos são capazes de acessar quase todos os recursos nativos de um dispositivo, como a câmera ou o giroscópio, usando *plugins* nativos.

3. Plataforma

Inicialmente, o aplicativo foi desenvolvido para a plataforma Android e iOS, através do CLI do *Ionic* para ter uma aplicação híbrida com uma interface prioritariamente intuitiva. Massita (2018) explica que a ideia da interface intuitiva é fazer com que qualquer usuário que tem contato como o *app* pela primeira vez consiga interagir com ele sem maiores dificuldades.

Assim, foi realizado o desenvolvimento híbrido, já que se busca garantir o que na visão de Massita (2018) significa assegurar uma boa performance da aplicação, propiciando ao usuário uma experiência única, por meio de um produto construído especialmente para a necessidade dos clientes.

Gonçalves (2012) explica que a maior vantagem desta abordagem está na questão da interface (MASSITA, 2018), uma vez que os aplicativos móveis devem se notabilizar pela propiciação de ótimas experiências na utilização, com vias de obter a atenção por parte do usuário. Além disso, Gonçalves (2018) assevera que a experiência de utilização é única em cada plataforma, o que significa que a mesma não pode ser transferida para outras plataformas, dada a singularidade que permeia cada experiência de utilização por parte do usuário.

É preciso, por conseguinte, que a interface do aplicativo seja simples e funcional e, principalmente, que os ícones escolhidos se destaquem, uma vez que, só a AppStore, conta com mais de 3,5 milhões de outros aplicativos e a *PlayStore*, conta com mais de 4 milhões de outros aplicativos. É preciso lembrar, ainda, que existem smartphones de diferentes tamanhos, resoluções e desempenhos, passando, constantemente, por atualizações.

3.1. Ciclo de vida do software de desenvolvimento móvel

Consoante CICLOS (2011), o ciclo de vida pode ser considerado como a estrutura que descreve todos os processos englobados no desenvolvimento, na operação e manutenção de um software, o qual abarca, dentre outros aspectos, como serão definidos os seus requisitos, além do término da utilização.

Ainda segundo CICLOS (2011), o modelo de ciclo de vida é a primeira escolha a ser feita no processo de software, que é o conjunto de atividades que constituem o desenvolvimento de um sistema computacional. Estas atividades são agrupadas nas seguintes fases:

- **Concepção** – todos os aplicativos começam com uma ideia. Essa ideia geralmente é refinada para tornar-se uma base sólida para um aplicativo.
- **Design** – a fase de design consiste em definir a UX (experiência do usuário) do aplicativo, por exemplo, como é o layout geral, como ele funciona, etc., bem como transformar essa experiência do Usuário em um design de UI (Interface do Usuário) adequado, geralmente com a Ajuda de um designer gráfico.
- **Desenvolvimento** – geralmente a fase com uso mais intensivo de recursos, costuma ser a verdadeira criação do aplicativo.
- **Estabilização** – quando o desenvolvimento é suficiente, o setor de garantia de qualidade geralmente começa a testar o aplicativo e os erros são corrigidos. Muitas vezes, um aplicativo entra em uma fase beta limitada na qual um público mais amplo recebe uma oportunidade para usá-lo, fornecer comentários e informar sobre eventuais alterações.
- **Implantação**: ocorre após o sistema ter passado satisfatoriamente por testes de aceitação. O sistema é transferido do ambiente de desenvolvimento para o ambiente de produção. O sistema antigo (se existir) deve migrar para o novo.
- **Manutenção** refere-se a todas as atividades relacionadas a um sistema depois que ele é implementado. Deve incluir atividades tais como a correção de software que não funcione corretamente, a adição de novos recursos aos sistemas em resposta às novas demandas dos usuários.

Os autores Gordon e Gordon (2006) ressalta que não há modelo de SDLC (*Systems Development Life Cycle*) uniformemente aceito. Alguns modelos combinam desenvolvimento e implementação em uma única etapa. Outros combinam o levantamento e a análise das necessidades também em uma única etapa. Alguns modelos dividem o projeto em projeto lógico e projeto físico.

Dessa forma, em cada fase são definidas, além das suas atividades, as funções e responsabilidades de cada membro da equipe, e como produto resultante, os artefatos.

3.2. Projetando o aplicativo

O primeiro passo para projetar uma aplicação consiste em definir as características a serem embutidas na interface do dispositivo que é o responsável por guiar o usuário.

Higor (2014) compreende que esta condição deve ser executada para desenvolver a interface de um dispositivo móvel com o intuito de padronizar a experiência do usuário na plataforma definida. Estes padrões atualmente podem ser baixados e consultados, uma vez que são disponibilizados pelos próprios desenvolvedores de sistemas operacionais.

Horvath, 2012 relata que para o caso específico do software em desenvolvimento, as guias de interface humana do IOS narram as diretrizes que auxiliam na projeção de uma interface para os usuários. Além disso, percebe-se que na plataforma IOS, os seus usuários demonstram preferência que causam a impressão de que foram desenvolvidos exclusivamente para esta plataforma. Ainda que de forma inconsciente e sem conhecer os princípios básicos de design de interface humana, os usuários conseguem perceber quando um aplicativo é fácil de ser manuseado e é interativo, bem como a ausência destes requisitos. Por este motivo, é essencial que os desenvolvedores de softwares ao criar aplicativos para IOS compreendam o que faz eles serem únicos em relação ao seu design e a sua interface.

3.3. Prototipando o aplicativo

Silvestre (2017) narra que a finalidade um protótipo é proporcionar uma experiência de utilização do artefato tecnológico com seus usuários. O propósito de sua criação é a realização de testes, que, por sua vez, tem o objetivo de identificar as falhas de performance e as oportunidades de melhorias existentes no protótipo do software.

A ferramenta utilizada para a prototipação será o figma. Machado (2019) explica que este é um software de prototipagem completo, o qual se caracteriza pela facilidade de utilização. Através do figma, pode-se definir as telas do software para o modelo navegável, o que reduz os custos de desenvolvimento do software. Além disso, o Figma propicia a retroalimentação dos usuários, o que torna o produto mais apto a responder de forma adequada as necessidades dos clientes.

Ainda sobre o figma, Higor (2014) aponta que esta ferramenta se caracteriza por viabilizar a prototipação em *wireframes* interativos, além de proporcionar interfaces tanto para sites quanto para aplicativos móveis, o que significa que pode ser utilizado tanto no modo on line como também no próprio sistema operacional.

Jefferson (2019) destaca que o figma tem se tornado uma ferramenta cada vez mais utilizada pelos times de desenvolvimento de softwares. Segundo este autor, os motivos que justificam este crescimento em relação ao uso são: a) pela facilidade de manuseio; b) pela sua flexibilidade; c) pelo seu baixo custo. Porém, consoante Castro (2019), o principal diferencial está no seu acesso, bastando o usuário fazer o seu cadastro no site da figma. Em adição a isso, as demais razões que fundamentam o seu uso são: a) interface simples; e, b) navegabilidade intuitiva. Posteriormente, após acessar o site e criar uma cota é necessário criar um projeto. Após a criação do projeto,

surge a opção de criar um protótipo para plataformas como iOS, Android, iPad ou Windows Phone; além de outras plataformas também disponíveis.

Higor (2014) reitera que após o protótipo finalizado, é necessário que seja realizado o seu teste diretamente no aparelho, pois, se o propósito é propiciar uma experiência única ao usuário, é primordial que sejam realizados testes de performance nos ambientes nativos.



Figura 1. Tela do software

3.4. Implantação do aplicativo

O Resonare poderá fazer uso tanto do microfone interno do *smartphone* quanto de um microfone externo adaptado ao aparelho, para captar o som, transmitido em um show vivo e, reproduzi-lo no dispositivo móvel, transformando o celular em uma espécie de caixa de som.

Ponte (2018) alerta que, embora os *smartphones* atuais contem com diversos microfones integrados, ainda não é possível uma boa captura de áudio em situações de muita pressão sonora, como, por exemplo, um show de rock. Isto faz com que o áudio gravado em ambientes de muito barulho não saia com uma qualidade satisfatória, o que dificulta a distinção entre o que é som ambiente e o que é ruído.

Ainda sobre essa questão do áudio e dos microfones em *smartphones*, Ponte (2018) explica que microfones externos compatíveis com estes aparelhos podem obter um melhor rendimento na captura de som, como por exemplo, os da marca *Shure*, os quais são elaborados para utilização em aparelhos IOS e *Android*.

O aplicativo desenvolvido também conta com recursos como botão de *Play* (iniciar a reprodução do áudio), pause (pausar a reprodução, permitindo que o usuário continue do ponto que parou através do botão *play*) e *stop*, o qual para a reprodução, sendo que nesta situação, ao clicar em *play*, o áudio é reproduzido do início novamente. Haverá um certo atraso na reprodução em alguns casos, mas isso é mais uma limitação tecnológica do que do aplicativo.

4. Considerações finais

A evolução tecnológica busca cada vez mais criar dispositivos menores em dimensão e superiores em qualidade. Apesar de ser bastante atrativa aos consumidores, esta tendência tecnológica de priorizar a miniaturização implica em problemas ergonômicos de manipulação do aparelho. Tais problemas aparecem na redução da quantidade das teclas presentes nos celulares, indo de encontro ao crescente aumento de funções nesses dispositivos.

O som, que teve uma evolução com menos etapas do que a resolução da tela do celular, também gerou bastante diferenças. Nesse sentido, levando-se em conta o avanço constante das configurações funcionais dos dispositivos móveis, o presente trabalho teve por objetivo desenvolver um software para captação de sonoplastia ao vivo, para reprodução no dispositivo móvel do usuário, transformando-o em uma espécie de caixa de som portátil.

Para tanto, a tecnologia para desenvolvimento baseada no framework Ionic, que permite criar e distribuir aplicativos híbridos e progressivos da *Web* em várias plataformas utilizadas por fabricantes de celulares, abrangendo uma maior variedade de aparelhos. Depreende-se a facilidade de utilização deste *framework* no desenvolvimento de aplicativos. Permitiu o desenvolvimento do aplicativo para a plataforma Android e iOS, através do CLI do Ionic e, ter uma aplicação híbrida com uma interface prioritariamente intuitiva.

A ferramenta utilizada para a prototipação será o Figma, um software de prototipação completo, rápido, que roda no navegador, e ainda é bastante simples de usar, com uma interface clean e autoexplicativa.

Assim, o recurso poderá ser baixado tanto para Android quanto para iOS, proporcionando ao usuário uma experiência mais completa e mais proveitosa de um show ao vivo, por exemplo, onde, dependendo de onde se está localizado, o som fica cada vez mais comprometido e inaudível, de modo que, a partir da captação do som, será possível reproduzi-lo, quase simultaneamente, no dispositivo móvel.

5. Referencias

- ANDRADE, Ana Paula De. O que é Ionic?. **Blog Treinaweb**, 24 de julho de 2019. Disponível em: < <https://www.treinaweb.com.br/blog/o-que-e-ionic/> >. Acesso em: 19 de setembro de 2019.
- ARAUJO, Fábio Bicalho de. **Desenvolvimento de software para dispositivos móveis**. [s.l.], 2006. Disponível em: < http://www.puc-rio.br/pibic/relatorio_resumo2006/relatorio/CTC/Cetuc/F%E1bio%20Bicalho%20de%20Araujo.pdf>. Acesso em: 23 de julho de 2019.
- BONIATI, B.B.; WAHLBRINCK, K.A. Aplicações *mobile* híbridas: um estudo de um caso de *framework Ionic* para a construção de um diário de classe. In: ENCONTRO ANUAL DE TECNOLOGIA DA INFORMAÇÃO.8.2017. **Anais...**Frederico Westphalen: IFFCFW-RS, 2017, p.69-86.
- CASTRO, Daniela. Wireframes e Protótipos: testando o Figma. **Medium**, 6 de junho de 2019. Disponível em: < <https://medium.com/@danielacfcastro/wireframes-e-prot%C3%B3tipos-testando-o-figma-8041993035b9> >. Acesso em: 30 de setembro de 2019.
- CICLOS de Vida do Software. **Devmedia**, 2011. Disponível em: < <https://www.devmedia.com.br/ciclos-de-vida-do-software/21099> >. Acesso em: 26 de junho de 2019.
- FERREIRA, Alexandre dos Santos Freire. *Ionic Framework*. **AlexandreFreire (AF)**, 2019. Disponível em: < <https://www.alexandrefreire.me/ionic-framework/>>. Acesso em: 28 de setembro de 2019.
- GONÇALVES, João Paulo. Desenvolvimento para dispositivos móveis: Nativo x Cross Platform. **Teclógica**, 7 de novembro de 2012. Disponível em: < <https://blog.teclogica.com.br/desenvolvimento-para-dispositivos-moveis-nativo-x-cross-platform/> >. Acesso em: 12 de junho de 2019.
- GORDON, Judith R.; GORDON, Steven R. Sistemas de informação: uma abordagem gerencial. Rio de Janeiro: LTC, 2006
- HIGOR. Projetando e criando Aplicativos para Dispositivos Móveis. **Devmedia**, 2014. Disponível em: < <https://www.devmedia.com.br/projetando-e-criando-aplicativos-para-dispositivos-moveis/30671>>. Acesso em: 24 de setembro de 2019.
- HORVATH, Eduardo. Guideline iOS – Introdução. **Redspark**, 26 de fevereiro de 2012. Disponível em: < <http://redspark.io/guideline-ios/> >. Acesso em: 01 de setembro de 2019.
- JEFFERSON, Alex. Já ouviu a palavra do Figma hoje, designer?. **Design culture**, 21 de agosto de 2019. Disponível em: < <https://designculture.com.br/ja-ouviu-a-palavra-do-figma-hoje-designer>>. Acesso em: 27 de setembro de 2019.
- LANGE JR, Neri Norberto. MERCADO, Neyza Bibiana Guzman. **Vantagens e desvantagens da utilização do ionic framework para o desenvolvimento de aplicativos móveis**. [s.l.], 2019. Disponível em: < https://semanaacademica.org.br/system/files/artigos/artigo_74.pdf>. Acesso em: 18 de setembro de 2019.

- MASSITA, Tiago. 5 Desafios no Desenvolvimento Mobile. **Monitora**, 22 de janeiro de 2018. Disponível em: < <https://www.monitoretec.com.br/blog/5-desafios-no-desenvolvimento-mobile/>>. Acesso em: 29 de setembro de 2019.
- MOURA, Simony C.R. de; MELO, Jucylene H.A. de. Mobilidade urbana na perspectiva das mulheres: hactivismo no mapeamento de assédio em transportes públicos. *In: WOMEN IN INFORMATION TECHNOLOGY, (WIT-SBC), 11, 2017. Anais...IX Women in Information Technology*. Porto Alegre: Sociedade Brasileira de Computação, julho de 2017.
- PONTE Adriano. Como gravar áudio de show com o celular [CT Responde]. **Canal Tech**, 19 de junho de 2018. Disponível em: < <https://canaltech.com.br/video/canaltech-responde/como-gravar-audio-de-show-com-o-celular-ct-responde-10474/>>. Acesso em: 13 de setembro de 2019.
- SILVESTRI, Gabriel. O que é UX Design: TUDO sobre como começar nessa área. **Gabriel Silvestri**, 17 de abril de 2018. Disponível em: < <https://gabrielsilvestri.com.br/o-que-e-ux-design/>>. Acesso em: 08 de agosto de 2019.