



UNIVERSIDADE DE BRASÍLIA

SOFIA SOARES DIAS

**DESENVOLVIMENTO DE APLICAÇÕES MÓVEIS ACESSÍVEIS:
ANÁLISE DA PLATAFORMA THUNKABLE X**

Brasília

2019

SOFIA SOARES DIAS

DESENVOLVIMENTO DE APLICAÇÕES MÓVEIS ACESSÍVEIS:
ANÁLISE DA PLATAFORMA THUNKABLE X

Trabalho de Conclusão de Curso apresentado ao Departamento de Línguas Estrangeiras e Tradução da Universidade de Brasília, como requisito para a obtenção de grau no curso de bacharelado em Línguas Estrangeiras Aplicadas ao Multilinguismo e à Sociedade da Informação.

Orientadora: Helena Santiago Vigata

Brasília

2019

SOFIA SOARES DIAS

DESENVOLVIMENTO DE APLICAÇÕES MÓVEIS ACESSÍVEIS:
ANÁLISE DA PLATAFORMA THUNKABLE X

Trabalho de Conclusão de Curso defendido e aprovado no Departamento de Línguas Estrangeiras e Tradução da Universidade de Brasília, como requisito para a obtenção de grau no curso de bacharelado em Línguas Estrangeiras Aplicadas ao Multilinguismo e à Sociedade da Informação, aprovado pela seguinte comissão examinadora:



Prof.^a Helena Santiago Vigata
Professora-Orientadora



Prof. Tiago Barros Pontes e Silva
Professor-Examinador



Prof.^a Patrícia Tuxi dos Santos
Professora-Examinadora

Brasília, julho de 2019

AGRADECIMENTOS

Primeiramente, agradeço à minha família, especialmente, à minha mãe, Sônia, que me apoia nos meus estudos e nas minhas escolhas. Obrigada por tudo que você fez por mim.

Agradeço a Helena Santiago Vigata, como minha orientadora, por ter abraçado o trabalho e me acompanhado nessa jornada. Também a agradeço como a professora que ajudou a consolidar meu interesse por acessibilidade em suas aulas sobre o tema.

Meu muito obrigado à professora Patrícia Tuxi e ao professor Virgílio Soares por terem me apresentado a Libras. Agradeço também a todos os outros professores e amigos que cederam tempo, durante aulas, congressos e férias, e comigo praticaram, corrigiram e ensinaram sobre essa rica língua. São muitas pessoas para nomear, mas vocês sabem quem são.

Agradeço calorosamente a Saulo Machado Mello de Sousa e Viviane Santos Almeida Queiroz que se dispuseram a colaborar com meu trabalho e participaram da avaliação de acessibilidade da plataforma aqui analisada. Agradeço a Casa Thomas Jefferson por ter cedido o espaço e os dispositivos móveis para testar o aplicativo aqui desenvolvido.

Agradeço a todos os membros da Empresa Júnior de Letras Quimera, os quais compartilharam os desafios e ansiedades da universidade e que contribuíram para meu crescimento profissional.

Por fim, sou grata a todos meus amigos e familiares de Brasília e de Belo Horizonte por estarem sempre ao meu lado.

RESUMO

Os *smartphones* têm ganhado uma considerável fatia do mercado tecnológico por serem um recurso que auxilia o usuário em vários aspectos do seu dia-a-dia. Contudo, nem sempre os aplicativos presentes nesses dispositivos são acessíveis a toda a população. Cabe aos desenvolvedores desses programas, solucionar esses obstáculos, porém, programar aplicativos acessíveis não é uma tarefa fácil. A Thinkable X é uma plataforma que propõe democratizar e popularizar o desenvolvimento dessas aplicações móveis, pois possibilita a criação de *apps* por meio de blocos lógicos sem programação. Este trabalho analisa se os recursos oferecidos pela plataforma conseguem cumprir as principais recomendações nacionais e internacionais de acessibilidade. Para isso, propõe-se criar uma aplicação na plataforma Thinkable X para a exposição multissensorial Entreates, um evento idealizado na Universidade de Brasília que proporciona um ambiente artístico acessível a todos. Após o levantamento de requisitos, a criação do protótipo e o desenvolvimento do *app*, usuários com diversidade funcional testaram o aplicativo. Com as avaliações, percebeu-se que, apesar da iniciativa possuir qualidades na criação de interfaces organizadas e atrativas, ainda necessita desenvolver melhorias para satisfazer algumas diretrizes de acessibilidade. Por fim, espera-se que mais desenvolvedores usem a plataforma para criar seus próprios aplicativos e reivindicar a ampliação de recursos acessíveis para fomentar o número de projetos digitais inclusivos no Brasil e no mundo.

Palavras-chave: Acessibilidade. Aplicação móvel. Exposição.

ABSTRACT

Smartphones have gained a significant share of the technology market, as they are a resource that assists users in various aspects of their day-to-day lives. Nevertheless, the applications in those devices are not always accessible to all people. The solution of these obstacles lies with those who develop software programs, yet programming accessible applications is not an easy task. Thunkable X is a platform that proposes the democratization and popularization of mobile application development, for it enables app creation through the usage of logic blocks and without programming. This paper analyzes if the resources offered by this platform can comply with the main international and national Brazilian recommendations for accessibility. To achieve this, we propose the development of an application on Thunkable X for the multisensory exhibition *Entreartes*, an event conceived in the University of Brasilia that provides an accessible artistic environment to all. After the elicitation and specification of the requirements, the creation of the prototype and the app development, functional diversity users could test it. Based on the evaluations, we noticed that, despite being able to create an organized and appealing interface, the initiative still needed to develop enhancements to meet a few accessibility guidelines. Finally, we expect more developers to use the platform to develop their own apps and to demand for an increase in accessible resources to foster the number of inclusive digital projects in Brazil and in the world.

Keywords: Accessibility. Mobile application. Exhibition.

RESUMEN

Los teléfonos inteligentes han ganado una considerable cuota del mercado tecnológico por ser un recurso que ayuda al usuario en varios aspectos de su cotidiano. Sin embargo, no siempre las aplicaciones contenidas en estos dispositivos son accesibles a toda la población. Recae sobre los desarrolladores de esos programas solucionar esos obstáculos, pero programar aplicaciones accesibles no es una tarea fácil. La Thunkable X es una plataforma que propone democratizar y popularizar el desarrollo de aplicaciones móviles, pues posibilita la creación de apps por medio de bloques lógicos sin programación. Este trabajo analiza si los recursos ofrecidos por la plataforma logran cumplir las principales recomendaciones brasileñas e internacionales de accesibilidad. Para hacerlo, se propone crear una aplicación en la plataforma Thunkable X para la exposición multisensorial Entreates, un evento idealizado en la Universidad de Brasilia que proporciona un ambiente artístico accesible a todos. Después de la obtención y análisis de los requisitos, la creación del prototipo y del desarrollo de la app, usuarios con diversidad funcional evaluaron la aplicación. Con las evaluaciones, se notó que, a pesar de la iniciativa poseer calidades en la creación de interfaces organizadas y atractivas, aún necesita desarrollar mejorías para satisfacer algunas directrices de accesibilidad. Finalmente, se espera que más desarrolladores utilicen la plataforma para crear sus propias aplicaciones y reivindicar la ampliación de recursos accesibles para fomentar el número de proyectos digitales inclusivos en Brasil y en el mundo.

Palabras clave: Accesibilidad. Aplicación móvil. Exposición.

RÉSUMÉ

Les smartphones ont gagné une considérable part du marché technologique, puisqu'ils sont une ressource qui aide l'utilisateur en plusieurs aspects de son quotidien. Toutefois, les applications de ces dispositifs ne sont pas toujours accessibles à toute la population. C'est incombé aux développeurs de ces logiciels de résoudre ces obstacles, pourtant programmer applications accessibles n'est pas une tâche facile. La Thunkable X une plateforme qui propose démocratiser et populariser le développement des applications mobiles, car elle permet la création de applis à travers des blocs logiques sans programmation. Ce travail analyse si les ressources offertes par la plateforme parviennent les principales recommandations brésiliennes et internationales d'accessibilité. Pour cela, on propose la création d'une application sur la plateforme Thunkable X pour l'exposition multisensorielle Entreates, un événement conçu à l'Université de Brasilia qui fournit un environnement artistique à tous. Après l'élicitation et analyse des exigences, la création du prototype et le développement de l'appli, des utilisateurs avec diversité fonctionnelle ont testé l'application. Basé sur l'évaluation, on a remarqué que, bien que l'initiative ait des qualités dans la création d'une interface organisée et attrayante, elle a encore besoin de développer des améliorations pour satisfaire quelques lignes directrices d'accessibilité. Finalement, on attend que les développeurs utilisent la plateforme pour créer ses propres applications et revendiquer l'augmentation des ressources accessibles pour promouvoir le nombre des projets numériques inclusifs aux Brésil et au monde.

Mots clefs : Accessibilité. Application mobile. Exposition.

LISTA DE FIGURAS

Figura 1 – Interface da Thunkable X e Thunkable Classic	31
Figura 2 – Diagrama de Entidade e Relacionamento (DER) do aplicativo Entreates	35
Figura 3 – Exemplo de entidade do DER	35
Figura 4 – Exemplo de relações do DER	36
Figura 5 – Exemplo de atributos do DER	36
Figura 6 –Protótipo da página de configuração de idiomas.....	37
Figura 7 – Protótipo da página de configuração de acessibilidade	38
Figura 8 – Protótipo da página principal	39
Figura 9 – Protótipo das demais páginas do aplicativo Entreates.....	40
Figura 10 – Componente personalizado para apresentar a função de imagens ao leitor de tela.....	43
Figura 11 – Comparação da tela do aplicativo com diferentes configurações de fonte do Android.....	44
Figura 12 – Bloco de vibração presente na plataforma Thunkable X.....	46
Figura 13 – Tela de montagem de blocos da plataforma Thunkable X	50
Figura 14 – Tela de design da página de artistas do aplicativo Entreates na plataforma Thunkable X	52

SUMÁRIO

1 INTRODUÇÃO	10
2 REFERENCIAL TEÓRICO.....	13
2.1 ACESSIBILIDADE: O QUE É E PARA QUEM É	13
2.2 DIREITO À ACESSIBILIDADE.....	16
2.3 OBSTÁCULOS PARA A ACESSIBILIDADE DIGITAL.....	18
2.4 COMO SOLUCIONAR ESSES OBSTÁCULOS.....	20
2.4.1 Tecnologia assistiva	20
2.4.2 <i>Design</i> universal e experiência de usuário	22
2.4.3 Guias e recomendações de acessibilidade.....	24
3 CONSTRUÇÃO DO APLICATIVO.....	29
3.1 METODOLOGIA.....	31
3.2 IDEALIZAÇÃO DO APLICATIVO.....	33
3.2.1 Levantamento e análise dos requisitos	33
3.2.2 Prototipagem	37
3.3 INVESTIGAÇÃO DOS RECURSOS DA THUNKABLE.....	40
3.4 AVALIAÇÃO DO APLICATIVO COM USUÁRIOS	46
3.5 AVALIAÇÃO DA THUNKABLE PELA DESENVOLVEDORA	49
4 CONSIDERAÇÕES FINAIS	53
REFERÊNCIAS.....	55

1 INTRODUÇÃO

Aparelhos tecnológicos, como computadores e *laptops*, conquistaram um grande espaço no cotidiano de muitas pessoas como artefatos essenciais para a realização de tarefas do dia-a-dia, especialmente com o advento da internet. Mas a predominância do computador pessoal no mercado da tecnologia ganhou um competidor à altura. Os celulares *smartphones* estão sendo adquiridos por muitos devido à facilidade de transporte que o seu tamanho reduzido proporciona. Em 2010, por exemplo, mais *smartphones* foram vendidos que computadores pessoais (ZUMO NOTÍCIAS, 2011). Desde o despertador até a calculadora, as facilidades viabilizadas pelos *softwares* presentes nesses dispositivos que cabem no bolso vêm preconizando sua expansão na sociedade moderna.

Entretanto, uma boa parte da população mundial não consegue usufruir do extenso potencial de benefícios e facilidades desses aparelhos. Motivados pelos problemas de acesso à informação, vários pesquisadores, empresas e governos estão produzindo diretrizes de acessibilidade para guiar desenvolvedores na criação de produtos e *sites* acessíveis a todos. Isto é, produtos que não apresentem obstáculos que impeçam sua utilização por pessoas com diferentes níveis de mobilidade, visão, audição e cognição, por exemplo. Entre as diversas iniciativas criadas para fomentar a acessibilidade digital, podemos citar as Diretrizes de Acessibilidade para Conteúdo Web (WCAG), que tiveram sua primeira versão produzida pelo consórcio do World Wide Web (W3C) em 1999 e são reconhecidas internacionalmente até hoje e, no Brasil, o Modelo de Acessibilidade do Governo Eletrônico lançado em 2004, conhecido como eMAG. Porém, em sua declaração de acessibilidade, o projeto GNU (2019) estima que 85% dos aplicativos de *software* e *sites* não cumprem as normas e orientações de acessibilidade.

Dentre os motivos que dificultam a implementação dessas políticas acessíveis e que fazem as empresas se absterem de seguir as diretrizes de acessibilidade em seus produtos estão a falta de conscientização, de tempo e de conhecimento dos desenvolvedores sobre as próprias diretrizes (LAZAR; DUDLEY-SPONAUGLE; GREENIDGE, 2004). Por outro lado, muitas pessoas que participam ativamente de comunidades surdas ou cegas, por exemplo, e que são instruídos quanto às recomendações de acessibilidade, não têm uma formação na área tecnológica e,

conseqüentemente, não têm o conhecimento necessário para desenvolver *softwares*.

Diante desse cenário, destacam-se projetos que têm como proposta a criação de *sites* e aplicativos sem o uso de programação. Na área de *sites*, citamos plataformas como Wix e WordPress, enquanto para aplicativos, uma companhia nascida de um projeto incubado no Instituto de Tecnologia de Massachusetts (MIT) com parceria da Google e fundada por Arun Saigal e WeiHua Li em 2015, chamada Thunkable, está na liderança global do mercado de criação de aplicativos sem nenhum código (PRNEWSWIRE, 2018).

A Thunkable propõe a criação de aplicativos com o uso de blocos lógicos, os quais substituem a programação. Esse mercado em crescimento pode abrir as portas para não-programadores criarem soluções para suas comunidades. Segundo Winn (2019), do jornal eletrônico *MIT News*, mais de 1,5 milhão de aplicações móveis já foram criadas com a plataforma, como um aplicativo que ajuda iraquianos a aprender inglês, um sistema de alerta de rodovias na Tailândia e um aplicativo que ajuda na compra e instalações de painéis solares no Iêmen. Saigal, o CEO e cofundador da Thunkable, acredita que “a tecnologia móvel só irá perceber seu total potencial para ajudar pessoas quando qualquer um puder criar um aplicativo para solucionar seus problemas” (WINN, 2019, tradução nossa¹).

A Thunkable é uma forma de democratizar o desenvolvimento de aplicativos e, se for possível projetar uma aplicação acessível com a plataforma, poderá haver um fomento de inovações na área da acessibilidade digital de aplicativos móveis, pois não-programadores poderão criar seus próprios aplicativos. Além disso, é notável a importância da criação de tecnologias acessíveis a todos os usuários, pois, além da acessibilidade da tecnologia de comunicação e informação ser um direito de todo cidadão brasileiro, o desenvolvimento de novos aplicativos acessíveis pode melhorar o dia-a-dia de idosos, pessoas com baixa escolaridade, não fluentes nas línguas empregadas, com conexão lenta à internet, que utilizam aparelhos antigos e novos usuários (THATCHER *et al.*, 2006).

É por isso que nesse trabalho nos propomos a buscar se os recursos disponíveis na plataforma Thunkable conseguem alcançar as recomendações de

¹ Do inglês: “[...] *mobile technology will only realize its full potential for helping people when anyone can build an app to solve their problems*”.

acessibilidade *web*. Para analisar esses recursos, desenvolveu-se um aplicativo para a exposição multissensorial *Entreartes*, nascida de um projeto de extensão e pesquisa do grupo *Acesso Livre*, da Universidade de Brasília, com intuito de organizar eventos artísticos acessíveis a todos.

Nesse contexto, analisamos os recursos oferecidos pela plataforma *Thunkable* com o propósito de verificar a capacidade de desenvolver aplicativos acessíveis sem o uso de programação e de expor possíveis barreiras encontradas na propagação dessa técnica. Para alcançar esse objetivo, levantamos os requisitos de acessibilidade necessários para o aplicativo teste, o qual contém informações da exposição *Entreartes*. Esses requisitos foram analisados com base nas diretrizes e recomendações pesquisadas e utilizados para criar um protótipo, o qual guiou o desenvolvimento do aplicativo na plataforma *Thunkable*.

Após o desenvolvimento do aplicativo, ele foi examinado por dois usuários e pela pesquisadora e, assim, avaliamos sua acessibilidade e sugerimos possíveis melhorias aos desenvolvedores da *Thunkable*. Essa popularização de aplicativos proposta pela *Thunkable* pode ser muito benéfica para usuários com diversidade funcional, caso os resultados desse trabalho mostrem que os aplicativos criados na plataforma podem ser acessíveis.

2 REFERENCIAL TEÓRICO

Para investigar se os recursos fornecidos pela plataforma Thinkable são suficientes para criar um aplicativo acessível sem o uso de programação, foi essencial compreender quais características tornam uma tecnologia acessível. Para isso, a seguir debateremos o significado atual de acessibilidade na legislação brasileira e como esse conceito ainda pode ser aprimorado. Também veremos a importância da acessibilidade e como esse direito é assegurado no Brasil e no mundo, bem como apontaremos os principais obstáculos para a acessibilidade digital e, conseqüentemente, como podemos evitá-los.

2.1 ACESSIBILIDADE: O QUE É E PARA QUEM É

Originado por volta do ano de 1940, o termo *acessibilidade* era e ainda é utilizado para instituir uma garantia de serviços que pudessem desobstruir o acesso a recursos na sociedade para pessoas com diversidade funcional (SILVA; LIMA, 2013). Isto é, ser acessível é facilitar o cotidiano de uma pessoa para que ela possa participar da vida em sociedade com independência.

No Brasil, conceitos como o de acessibilidade vêm sendo gradativamente introduzidos à legislação brasileira com o avanço dos anos. Uma das leis regulamentadas pelo Decreto 5.296, aprovado em 2004, a Lei 10.098/00, definiu o termo como a:

[...] possibilidade e condição de alcance para utilização, com segurança e autonomia, de espaços, mobiliários, equipamentos urbanos, edificações, transportes, informação e comunicação, inclusive seus sistemas e tecnologias, bem como de outros serviços e instalações abertos ao público, de uso público ou privados de uso coletivo, tanto na zona urbana como na rural, por pessoa com deficiência ou com mobilidade reduzida (BRASIL, 2000).

Com isso, nota-se que o conceito de acessibilidade abrange recursos de aspectos físicos e também não-físicos, como a *web*. Dessa forma, rampas, elevadores e vagas de fácil acesso são consideradas soluções acessíveis, assim como também o são as legendas e os atributos `<alt>`² para descrições de imagens em *websites*. Portanto, é necessário que a legislação seja idealizada levando em

² O atributo `<alt>` é utilizado em documentos em HTML, uma linguagem frequentemente utilizada na construção de *websites*. Esse atributo serve para inserir um texto alternativo para certos elementos, como em imagens. Assim, se o usuário utiliza um leitor de tela, ou mesmo se houver contratempos no carregamento de uma imagem, o texto que a descreve será apresentado.

consideração todas as facetas do termo. Durante esse projeto, no entanto, daremos maior ênfase aos princípios da acessibilidade digital.

Para compreender o termo por completo, deve-se considerar ainda o público-alvo de projetos acessíveis. A Convenção Internacional sobre os Direitos das Pessoas com Deficiência incorporada na legislação brasileira como emenda constitucional aprovada pelo Decreto nº 6.949/2009 fornece o conceito legal de “pessoa com deficiência”, conforme explica Maranhão (2014, p. 146). Essa Convenção, realizada em 2006 pela Organização das Nações Unidas (ONU), teve como objetivo “promover, proteger e assegurar o exercício pleno e equitativo de todos os direitos humanos e liberdades fundamentais por todas as pessoas com deficiência” (BRASIL, 2009a) e contribuiu para uma legislação mais completa sobre acessibilidade no Brasil.

Nesse conceito é estabelecido que a “pessoa com deficiência possui “impedimentos”, os quais encontram barreiras que obstruem a plena participação do indivíduo na sociedade (BRASIL, 2009a). São eles o impedimento motor, cognitivo e sensorial. Similarmente, no Decreto nº 5.296/2004, há a definição legal de cada “deficiência” (física, auditiva, visual, intelectual e múltipla) com informações técnicas para o enquadramento do indivíduo naquela categoria (BRASIL, 2004). Porém, ainda sem uma definição geral da “pessoa com deficiência”, como acontece no texto da Convenção.

Esses também são os tipos de diversidade funcional empregados pelo Instituto Brasileiro de Geografia e Estatística (IBGE) em seu Censo Demográfico de 2010. De acordo com os dados do IBGE (2010), mais de 45,6 milhões de pessoas possuem pelo menos uma das diversidades funcionais investigadas. Isso significa que quase um quarto da população brasileira enfrenta diversas barreiras físicas e tecnológicas diariamente. Esse número tende a crescer ainda mais com o envelhecimento da população, visto que com o avanço da idade, casos de diminuição da mobilidade, visão, audição e habilidades cognitivas ficam mais comuns (THATCHER *et al.*, 2006).

Assim, compreende-se que a acessibilidade também pode beneficiar a produtividade da população como um todo, não só das pessoas com diversidade funcional. Além dos idosos, pessoas com baixa escolaridade, não fluentes nas línguas empregadas, com conexão lenta à internet, que utilizam aparelhos antigos e

novos usuários podem ser auxiliados, respectivamente, por textos com frases menos complexas, textos alternativos para multimídias que pesam no *download* e um *design* simples e consistente para navegação, conforme demonstram Thatcher *et al.* (2006). Ademais, há situações corriqueiras em que a acessibilidade digital também favoreceria os demais usuários. Como exemplifica Kavcic (2005), o contraste e a escolha de cores podem ajudar a diminuir o cansaço visual causado pela utilização de aparelhos por horas prolongadas devido ao trabalho e estudo, bem como estímulos visuais na tela são ideais para usuários em ambientes muito barulhentos ou onde o som é proibido e não podem ouvir os sons do aparelho.

Levando isso em consideração, a definição legal de acessibilidade no Brasil, ainda mantida na Lei Brasileira de Inclusão, nº 13.146/2015, que apresenta o conceito de possibilidade de utilização autônoma de espaços físicos e não-físicos apenas para “pessoas com deficiência”, aparenta não abarcar todo o potencial desse termo. Essa ampliação do conceito acontece, por exemplo, na legislação espanhola com o Real Decreto Legislativo 1/2013, o qual unifica leis como a lei nº 51/2003, que estabelece o conceito de acessibilidade universal. Em sua definição, o direito à acessibilidade é estendido para além da pessoa com diversidade funcional, sendo assegurado a todos os cidadãos e, portanto, considerado um direito universal (ESPANHA, 2013).

Outra mudança proveniente da adoção da Convenção Internacional na legislação brasileira foi a substituição da expressão “pessoas portadoras de deficiência” para “pessoas com deficiência”, termo de preferência internacional, aprovado pela Portaria da Secretaria de Direitos Humanos da Presidência da República, nº 2.344, em 3 de novembro de 2010 (BRASIL, 2010). Todavia, é importante ressaltar que muitas vezes a expressão “pessoa com deficiência” não é a utilizada pela própria comunidade. Dentro da comunidade surda, por exemplo, diz-se “surdo” para pessoas com pouca ou nenhuma audição e “não-surdo” para um indivíduo com plena capacidade auditiva. Há até mesmo diferentes nuances no uso da palavra “surdo”, sendo Surdo (com maiúscula) não só o significado para uma pessoa com dificuldades auditivas, mas aquele que se apropria do substantivo e se declara como tal, por adotar a Libras e a identidade Surda (CASTRO, 2011, p. 12). E, em vez de deficiência visual, muitos preferem utilizar os termos cego ou com baixa-visão.

Há também novos termos surgindo internacionalmente, como o termo *diversidade funcional*, cunhado pelos pesquisadores espanhóis Javier Romañach e Manuel Lobato e apresentado no Foro de Vida Independiente em 2005. Os idealizadores da expressão explicam, por exemplo, que uma pessoa com lesão muscular usa uma cadeira de rodas para se deslocar, enquanto a maioria da população o faz usando as pernas, porém, a finalidade ainda é a mesma, a locomoção. Ambos têm a mesma função, porém atuando de maneiras diversas, por isso, o termo retrata essa realidade de diversidade funcional (ROMAÑACH; LOBATO, 2005).

Palavras como “deficiência” carregam em si uma conotação negativa de “falta de capacidade” e estabelecem um padrão de “normalidade” determinado por uma comparação com uma maioria. Por isso, nesse trabalho, optou-se por utilizar o termo “diversidade funcional” devido à sua atitude de celebração das diferenças, em vez de tratar as pessoas com diversidade funcional como parte do problema. Ademais, encoraja-se debates mais extensos acerca dos diferentes termos e suas aceitações pelas comunidades concernentes em trabalhos vindouros com enfoque linguístico.

Em suma, a acessibilidade traz consigo a possibilidade universal de interação com espaços, serviços e meios de informações, sejam eles físicos ou não, e, assim, garantir que todos os cidadãos tenham acesso e possam usufruir de forma clara, fácil e completa de todos os recursos presentes na sociedade.

2.2 DIREITO À ACESSIBILIDADE

O estabelecimento de conceitos legais, como o de *acessibilidade*, é vital para que o direito dos cidadãos seja assegurado, pois, a partir deles, serão formuladas medidas que os estimulem. No Brasil, começaram a ser instituídos os primeiros direitos às “pessoas portadoras de deficiência” na Constituição de 1988. Um ano depois, em 1989, a Lei nº 7.853 vem garantir a integração social desse grupo por meio da definição de formas de apoio, crimes puníveis e disciplina a atuação do Ministério Público. Para regulamentar a Lei nº 7.853, promulga-se, em 1999, o Decreto nº 3.298, o qual delinea as disposições da lei e viabiliza o direito às ajudas técnicas, termo hoje substituído por tecnologia assistiva. Esses recursos e serviços que auxiliam na superação de limitações ganham também um capítulo no Decreto nº 5.298/04, no qual se pormenorizam os planos do Governo na área de tecnologia assistiva. Nesse decreto ainda se regulamentam as leis nº 10.048 e nº 10.098, as

quais estabelecem, respectivamente, a prioridade de atendimento a pessoas com diversidade funcional, idosos e gestantes e as normas gerais para a promoção de acessibilidade.

Além disso, com o crescimento da tecnologia de informação, o legislativo começa a promulgar leis que abordam também a acessibilidade digital, como a Lei nº 13.146, sancionada em 2015. Conhecida como a Lei Brasileira de Inclusão da Pessoa com Deficiência (LBI), ela vem para alterar leis como o Código Eleitoral, Código de Defesa do Consumidor e a Consolidação das Leis do Trabalho, adequando-as para com o texto da Convenção da ONU. Segundo Gabrilli (2016, p. 12), deputada relatora da LBI, foram alteradas “leis que não atendiam ao novo paradigma da pessoa com deficiência ou que simplesmente a excluía de seu escopo”.

A LBI, “que veio para assegurar e promover, em condições de igualdade, o exercício dos direitos e das liberdades fundamentais por pessoa com deficiência” (NAVES *et al.*, 2016, p. 7), destina um de seus capítulos ao acesso à informação e comunicação de forma acessível, já que a acessibilidade, como definida pela Lei nº 10.098, engloba diversas esferas sociais, não só em espaço físico, mas também *online*. Em seu Art. 67, a Lei Brasileira de Inclusão determina a utilização de recursos como legendagem, audiodescrição e janela de intérprete de Libras, reforçando o direito à inclusão digital. No entanto, apesar do avanço das leis que regulamentam e estimulam a acessibilidade na área da tecnologia de informação e comunicação, especialmente de *sites* governamentais, ainda não existe legislação brasileira que estabeleça normas para aplicativos móveis.

Enquanto isso, os Estados Unidos foram um dos primeiros países a implementar políticas e diretrizes de acessibilidade, incluindo acessibilidade digital. Em 1986, o acesso à informação para pessoas com diversidade funcional se tornou obrigatório com a emenda da Seção 508 na *Federal Rehabilitation Act* de 1973, na qual foi estabelecido que as tecnologias de informação utilizadas e produzidas pelo Governo fossem projetadas com a acessibilidade em mente (BURGSTAHLER, 2019). Indubitavelmente, a legislação dos EUA tem mais abrangência nas áreas de acessibilidade digital comparada à brasileira. Em 2010, por exemplo, o país estende as disposições de acessibilidade às novas tecnologias informacionais, como

smartphones e tablets, por meio da *21st Century Communication and Video Accessibility Act*.

Já na legislação europeia, Kavcic (2005, p. 1024) cita a iniciativa eEurope, proposta pelo Comissão Europeia em 1999 como uma maneira de integrar pessoas com diversidade funcional e idosos na sociedade da informação e, assim, modernizar a economia europeia. Com planos de ação em 2002 e 2005 para estimular esses serviços e programas acessíveis, a União Europeia realizou um relatório, em 2007, intitulado *Measuring progress of e-Accessibility in Europe* para avaliar o cumprimento dessas políticas pelos Estados-membros e definir futuras necessidades de melhoria dessas políticas (NARASIMHAN, 2012). Outro exemplo a ser seguido em relação à avaliação e monitoramento de políticas acessíveis é a Coreia do Sul. Narasimhan (2012, p. 24) relata que o país conduz pesquisas anuais sobre os usos de computadores e internet por pessoas com diversidade funcional, bem como sobre a conformidade com as normas de acessibilidade estabelecidas nacionalmente e a conscientização da população sobre o tema. Levando em consideração as informações expostas no presente trabalho, considera-se necessário que o Brasil também faça esse tipo de levantamento para avaliar o cumprimento da legislação de acessibilidade por parte de órgãos do Governo e empresas privadas, atualizar termos defasados, adicionar políticas para suprir lacunas em relação às novas tecnologias e avaliar a conscientização da sociedade brasileira sobre a acessibilidade digital.

2.3 OBSTÁCULOS PARA A ACESSIBILIDADE DIGITAL

Para a criação de um produto acessível, é indispensável compreender a diversidade funcional e as dificuldades enfrentadas por certos usuários. Desse modo, o desenvolvedor poderá identificar possíveis empecilhos e evitá-los de forma mais consciente. Para isso, o Trace Center da Universidade de Wisconsin, nos EUA, compôs as *Application Software Design Guidelines* (VANDERHEIN, 1994), que explanam sobre os principais motivos de exclusão tecnológica para usuários com:

- a) diversidade funcional auditiva: surdos parciais ou totais não conseguiriam distinguir ou detectar sons durante o uso do dispositivo tecnológico. Textos muito longos e complexos na língua escrita podem também não ser completamente compreendidos, visto que grande parte dos surdos tem a língua de sinais do país

como a primeira língua. Outro obstáculo seria a entrada de voz, já que há muitos surdos não-oralizados;

b) diversidade funcional intelectual: pessoas com déficit de atenção, dislexia, dificuldade de aprendizagem ou demência podem ter um comprometimento da memória, linguagem, percepção ou aprendizagem. Conseqüentemente, recuperar informações, compreender conceitos abstratos e solucionar problemas em sequência pode ser desafiador;

c) diversidade funcional física: as restrições de movimento ou de controle de braços, mãos e dedos podem afetar a manipulação de objetos como *mouses* e teclados. Logo, tecnologias que requerem respostas em uma quantidade (especialmente curta) de tempo específica ou o pressionamento de duas ou mais teclas ao mesmo tempo poderiam trazer problemas ao usuário; e

d) diversidade funcional visual: informações dispostas em formato de gráfico e imagem na tela comprometeriam a utilização do aparelho por pessoas com baixa visão e cegueira. Dispositivos apontadores, como o *mouse*, que exigem coordenação olho-mão também representariam um problema.

Enfim, o desenvolvimento de uma experiência de interação exitosa com o produto tem relação direta com a compreensão profunda das necessidades e desejos do usuário (FORLIZZI; BATTARBEE, 2004, p. 265). A empatia é parte chave do processo. Em razão disso, o convívio com pessoas com deficiência é um diferencial na concepção de tecnologias acessíveis a esses usuários, porque de tal modo criamos um produto que eles possam não apenas utilizar, mas, acima de tudo, desfrutar. Compreendemos também que há uma individualidade mesmo dentro da diversidade, pois cada usuário tem a sua necessidade. Por exemplo, cada surdo tem um diferente grau de surdez e também pode ter outras diversidades como baixa visão e dificuldade de mobilidade.

Ademais, além dos obstáculos com os quais os usuários normalmente se confrontam, há ainda dificuldades técnicas enfrentadas pelos desenvolvedores. Para criar um aplicativo acessível, é necessário o domínio de programação e dos parâmetros de acessibilidade. No Brasil, contudo, ainda são poucas as pessoas que têm a oportunidade de adquirir esse tipo de conhecimento. Por isso, uma ferramenta como a Thunkable, que não requer programação para o desenvolvimento de um

app, poderia ser um caminho para a democratização e a popularização do desenvolvimento de aplicações acessíveis.

2.4 COMO SOLUCIONAR ESSES OBSTÁCULOS

Depois de compreender as dificuldades normalmente enfrentadas por diferentes tipos de usuários no mundo digital, o desenvolvedor deve procurar meios de evitar essas barreiras. Apresentamos, em seguida, formas de amparar o usuário, com tecnologias assistivas e *design* universal, e o desenvolvedor, com diretrizes e recomendações de acessibilidade.

2.4.1 Tecnologia assistiva

Em virtude da expansão da legislação que estimula os direitos à acessibilidade, outra expressão vem ganhando espaço nas pesquisas técnicas da área. A tecnologia assistiva (TA), também referida como ajudas técnicas, é, segundo o Comitê de Ajudas Técnicas (CAT):

[...] uma área do conhecimento, de característica interdisciplinar, que engloba produtos, recursos, metodologias, estratégias, práticas e serviços que objetivam promover a funcionalidade, relacionada à atividade e participação, de pessoas com deficiência, incapacidades ou mobilidade reduzida, visando sua autonomia, independência, qualidade de vida e inclusão social (BRASIL, 2009b).

O ajuste da expressão ocorreu em agosto de 2007, pelo CAT, porque o novo termo, ainda considerado como um sinônimo do predecessor, já estava firmado nacionalmente no meio acadêmico, em setores governamentais e em organizações, conforme explica Bersch (2017, p. 14). Como resultado, atualmente é o termo considerado como mais adequado para definir os equipamentos e serviços desenvolvidos especificamente para se adaptar a aspectos do dia-a-dia dos usuários, auxiliando-os na execução de tarefas diárias que, sem o auxílio dessa tecnologia, não conseguiriam realizar.

Bersch (2017, p. 11) nos elucida que a tecnologia assistiva não é um “recurso do profissional”, e sim do usuário. Seja um talher modificado para alguém com dificuldades motoras, um sistema de controle remoto para controlar as luzes da casa, uma bengala, uma órtese para correção postural ou uma lupa, a TA sempre tem como foco principal a pessoa que a utiliza. De modo que é indispensável o envolvimento do usuário desde o planejamento até a avaliação do produto final (BERSCH, 2017, p. 13).

Na área de tecnologia de informação e comunicação, especificamente no uso de recursos como computadores e celulares, a tecnologia assistiva vem possibilitando para essas pessoas uma maior comunicação, aprendizado e inserção no mercado de trabalho. Esses recursos podem ser divididos em três grupos para ambientes computacionais, como propostos pelo Programa Informática, Educação e Necessidades Especiais (InfoEsp), das Obras Sociais Irmã Dulce (GALVÃO FILHO, 2009):

- a) adaptações físicas ou órteses: são ajustes ou aparelhos corporais feitos para a necessidade particular do indivíduo que o auxiliam no manuseio e utilização do dispositivo, como aparelhos auditivos e estabilizadores de punho com ponteira para digitação;
- b) adaptações no *hardware*: são adaptações aos componentes físicos do computador ou do celular. Por exemplo, teclados com alteração na inclinação, *mouses* adaptados e impressoras de Braille; e
- c) *softwares* especiais de acessibilidade: são programas presentes em computadores e aparelhos móveis que possibilitam a interação com os aparelhos. Alguns exemplos disso são opções de alto-contraste, simuladores de teclado e *mouse* e programas de dicionários online em língua de sinais³

Por conseguinte, percebemos que a ferramenta Thunkable não pode ser descrita como uma TA, pois é um recurso para os desenvolvedores. No entanto, os aplicativos produzidos na plataforma podem fazer parte dessa área como *softwares* de acessibilidade se forem acessíveis e ajudarem a fornecer uma melhoria na qualidade de vida de homens e mulheres com diversidade funcional.

Aliás, os sistemas operacionais Android e iOS já apresentam alguns *softwares* acessíveis de forma nativa, isto é, sem a necessidade de fazer *download* de aplicativos de terceiros, para facilitar a utilização de aparelhos móveis, como *tablet* e celulares. Por exemplo, nos produtos da Apple, com o sistema iOS, disponibiliza-se uma TA de leitura de tela chamada VoiceOver, enquanto nos Androids temos o TalkBack, produzido pela Google, para realizar essa função desde

³ Dicionários e tradutores online de línguas de sinais podem auxiliar na interpretação de textos simples de, por exemplo, português para Libras. Porém, essas iniciativas, apesar de pioneiras no Brasil, podem não ser muito precisas, como também acontece com a maioria das ferramentas que se baseiam em traduções automáticas, devido à falta de reflexão quanto à contextualização dos sinais e palavras.

2009. Com ambos, pode-se deslizar o dedo pela tela do aparelho e uma voz reproduzirá o conteúdo pressionado. Dentro de aplicativos, é necessário que as informações estejam programadas de maneira acessível, com rótulos e sequências lógicas, para que o leitor de tela consiga passá-las ao usuário de forma detalhada e compreensível. Características que tentamos emular nesse trabalho, sem programação, com o uso da Thunkable. Embora ainda haja melhorias a serem adicionadas nesses sistemas operacionais, tais recursos preparam o caminho para outras inovações no mercado de tecnologia de informação e comunicação acessível em aparelhos móveis.

2.4.2 *Design* universal e experiência de usuário

Assim como a tecnologia assistiva, o *design universal* é um conceito que começa, recentemente, a aparecer em leis, pesquisas e referenciais teóricos sobre acessibilidade no Brasil, destacando-se como uma maneira de superar os obstáculos enfrentados pelas pessoas com diversidade funcional. A Lei Brasileira de Inclusão (BRASIL, 2015) aponta o *design universal* como a “concepção de produtos, ambientes, programas e serviços a serem usados por todas as pessoas, sem necessidade de adaptação ou de projeto específico”. Em outras palavras, seria uma maneira de criar ambientes acessíveis ao maior número de pessoas possíveis simultaneamente. E, uma vez que a acessibilidade do ambiente ou produto for desenvolvida de forma concomitante ao resto do projeto, ela ganhará, por um lado, mais atenção e tempo da equipe, além de contar com uma investigação específica para incorporá-la naquele contexto. Por outro lado, é uma maneira de evitar alterações posteriores para adaptar o projeto a, por exemplo, idosos, pessoas com diversidade funcional e gestantes, o que seria custoso.

E para guiar o processo de planejamento e avaliar os planejamentos existentes, um grupo de arquitetos, engenheiros e *designers* compilaram, em 1997, os sete princípios do *design universal* (CENTER OF UNIVERSAL DESIGN, 1997):

- a) baixo nível de esforço físico: o usuário deve poder usufruir do produto de forma eficiente, confortável e com o mínimo de fadiga. Por exemplo, é recomendável permitir ao usuário manter uma posição neutra do corpo e minimizar ações repetitivas;

- b) perceptibilidade e informação: faz com que as informações essenciais sejam passadas de forma efetiva ao usuário, independentemente das capacidades do usuário. É recomendado o uso de contraste e tamanho de fonte adequados e diferentes modos (verbal, tátil, pictórico) de apresentação do mesmo conteúdo;
- c) tamanho e espaço adequado para utilização: estabelecimento de um espaço apropriado para alcançar e manipular o produto, levando em consideração o tamanho, postura e mobilidade do usuário. É interessante contar com espaço para dispositivos assistivos e se adaptar à mão do usuário, se for preciso;
- d) tolerância a erros: providenciando avisos de erros e riscos ao utilizador do produto e organizando elementos de modo a minimizar possíveis erros, os desenvolvedores podem amenizar as consequências de ações acidentais e não-intencionais;
- e) uso equitativo: torna o *design* útil e comercializável para todos. Para isso, é necessário, por exemplo, disponibilizar políticas de privacidade e segurança equitativas, evitar segregações e proporcionar um *design* atraente aos usuários;
- f) uso flexível: suporta diversas preferências individuais e capacidades, provendo escolhas nos modos de utilização, facilitando a precisão e se adaptando ao ritmo do usuário; e
- g) uso simples e intuitivo: deve ser fácil de se entender, independentemente da experiência, conhecimento, competência linguística ou níveis de concentrações atuais do usuário. Assim sendo, é aconselhável eliminar complexidades desnecessárias no texto e no *design*, ser consistente durante a utilização, organizar as informações por importância e oferecer sugestões e comentários durante e depois da execução de tarefas.

Nota-se que esses princípios, reunidos em 1997, são os mais atuais em questão de *design* universal. A falta de conscientização sobre essa temática coíbe estudos que contribuam com atualizações que agreguem a esses princípios com base em reflexões sobre as mudanças tecnológicas mais recentes. Além disso, o desenvolvimento de um *design* universal tem como base a usabilidade do produto, isto é, a eficiência e a facilidade com que o usuário consegue navegar e completar tarefas usando a interface do produto. Entretanto, a prática do *design* vai além da usabilidade. Deve-se incorporar outras considerações, como questões financeiras e

culturais, no processo de concepção do *design* para todos (CENTER OF UNIVERSAL DESIGN, 1997). Logo, surge a ideiação do desenvolvimento de um *design* de interação que evocaria emoções e seduziria seus usuários por meio da experiência do produto.

Assim, o conceito de “experiência do usuário”, popularizado em 1998 por Donald Norman, é mais abrangente do que o da usabilidade, pois analisa as expectativas, sentimentos e satisfações dos usuários em relação à interação com o produto (GUIMARÃES; TAVARES, 2014). Atualmente, muitas marcas têm apostado na criação de experiências de usuário diferenciadas, pois é uma forma de cativar os consumidores pelo apelo emocional e, então, se distinguir dos concorrentes (IIDA; SILVA; SARMET, 2008, p. 24).

O *design* de um aplicativo acessível, por exemplo, pode gerar vários tipos de emoções no usuário. Ele pode ficar feliz com a possibilidade de explorar um conteúdo ao qual não conseguia ter acesso anteriormente. Ou pode ficar frustrado caso algo não funcione como o esperado. Essas emoções são, ao final do uso, traduzidas como uma experiência de interação usuário-produto e também podem afetar a avaliação final do produto. Segundo Hassenzahl e Tractinsky (2006, p. 95, tradução nossa⁴), essas experiências são

[...] consequência do estado interno do usuário (predisposições, expectativas, necessidades, motivação, humor etc.), das características do sistema planejado (complexidade, propósito, usabilidade, funcionalidade etc.) e do contexto (do ambiente) no qual as interações ocorrem (organizacional/contexto social, significação da atividade, voluntariedade do uso etc.).

Por conseguinte, é necessário considerar todos esses aspectos tanto no planejamento inicial, para criar a experiência que o desenvolvedor deseja passar ao usuário, quanto na avaliação final do aplicativo, para analisar se essa experiência foi alcançada com sucesso.

2.4.3 Guias e recomendações de acessibilidade

Para fomentar a criação de *softwares* e *sites* acessíveis, empresas, governos e pesquisadores começaram a produzir guias e diretrizes com sugestões de acessibilidade. Kavcic (2005, p. 1.026) demonstra que muitos desses documentos

⁴ Do inglês: “[...] consequence of a user’s internal state (predispositions, expectations, needs, motivation, mood, etc.), the characteristics of the designed system (e.g. complexity, purpose, usability, functionality, etc.) and the context (or the environment) within which the interaction occurs (e.g. organisational/social setting, meaningfulness of the activity, voluntariness of use, etc.)”.

aderem recomendações similares e compila e descreve, com base nas *Irish National Disability Authority IT Accessibility Guidelines*, *IBM Software Accessibility Checklist*, *Sun Microsystems Accessibility Design Guidelines* e *Section 508 Guidelines*⁵, as cinco áreas de *software* com os maiores problemas de acessibilidade:

- a) sistema e conteúdo: é crucial o reconhecimento dos recursos de acessibilidade do sistema operacional mesmo dentro de um programa. Para evitar obstáculos, é também essencial que a interface do usuário seja consistente em todo o *software*. Para o conteúdo, deve haver um indicador visual do foco atual do cursor, assim como são necessários rótulos (no código) em controles, objetos e ícones para os leitores de tela. Ademais, a linguagem utilizada precisa ser simples e o tempo de resposta deve ser suficiente para até o mais lento dos usuários;
- b) sons e multimídia: para impedir problemas de acessibilidade, deve se assegurar que informações como gráficos e imagens sejam descritas no código e que vídeos e animações tenham alternativas acessíveis, como as legendas e a audiodescrição. Para alertas de áudio, devem ser exibidos estímulos visuais simultaneamente. É desejável haver um suporte para língua de sinais;
- c) teclado e dispositivos apontadores: a possibilidade de navegação somente pelo teclado necessita ser fornecida para usuários que não podem utilizar o *mouse*. Assim, todas as ações realizadas precisam ter equivalente no teclado, com combinações simples de teclas. A tecla *tab* sempre serve para navegar, em sequência lógica, pelos elementos. Enquanto isso, o uso de um dispositivo apontador, como o *mouse*, deve ser opcional. Em suma, é interessante sempre haver mais de uma forma de realizar a mesma tarefa;
- d) tela: com relação à exibição de conteúdo, é imprescindível que não sejam anuladas as configurações atuais do sistema para fonte, tamanho e cor. Por isso, não é aconselhável que esses elementos sejam *hard-coded*, ou seja, predefinidos para um valor exato no código. Além disso, deve haver contraste entre as cores utilizadas e elas não podem ser a única maneira de transmitir uma informação. E devem-se evitar elementos piscantes;

⁵ As versões mais atuais das [diretrizes de acessibilidade irlandesas para software da Irish National Disability Authority](#); da [listagem de acessibilidade software da empresa IBM](#); das [diretrizes de acessibilidade do design da empresa Sun Microsystems](#) adquirida pela Oracle Corporation em 2009; e das [diretrizes da seção 508 dos Estados Unidos da América](#).

e) verificação de acessibilidade e documentação: é fundamental que os produtos sejam testados em sua acessibilidade, tanto de forma automatizada em *sites* especializados, quanto por usuários. Outra forma de fazê-lo seria testar o *software* com tecnologia assistiva ou mesmo desligando o som ou a tela do dispositivo. Por fim, é de praxe fornecer instruções e descrições dos recursos acessíveis, assim como para a instalação, documentação e materiais de suporte.

Em consonância com o avanço legislativo da acessibilidade e do direito ao acesso à informação a pessoas com deficiência, o Governo brasileiro desenvolveu, em 2004, o Modelo de Acessibilidade em Governo Eletrônico (eMAG). O documento contém um conjunto de recomendações, baseadas “no estudo de 14 normas existentes em outros países acerca da acessibilidade digital” (eMAG, 2014, p. 8). Dentre essas normas, se destacam: as Diretrizes de Acessibilidade para Conteúdo Web 2.0 (WCAG 2.0), do World Wide Web Consortium (W3C); o guia de padronização da acessibilidade de *software*, estabelecido na Seção 508 da legislação dos Estados Unidos da América; as Diretrizes Irlandesas de Acessibilidade; o canadense *Common Look and Feel Standards for the Internet*, conhecido como padrões CLF; e documentos de Portugal e Espanha. O eMAG inclui seis recomendações (marcação, comportamento, conteúdo/informação, apresentação/*design*, multimídia e formulários) que auxiliam na programação de um *website* para uma implementação padronizada e coerente da acessibilidade digital.

Entre as normas utilizadas no eMAG, a mais difundida internacionalmente são as diretrizes WCAG. Criadas em 1999 e depois atualizadas em 2008 e 2018, fazem parte da iniciativa de acessibilidade *web*, conhecida como WAI, lançada pelo organismo responsável por desenvolver padronizações mundiais relacionadas à *web*, o consórcio World Wide Web (W3C). E, além das WCAG, que propõem recomendações para a acessibilidade de conteúdo *web* e *sites*, a iniciativa também publicou as *Authoring Tool Accessibility Guidelines* (ATAG) e as *User Agent Accessibility Guidelines* (UAAG), as quais descrevem como fazer, respectivamente, *softwares* acessíveis de produção de conteúdo e navegadores e reprodutores (*media players*) acessíveis (THATCHER *et al.*, 2006).

A versão mais recente da WCAG é a 2.1⁶, a qual substitui a anterior desde junho de 2018. Porém, a última versão do eMAG, a versão 3.1⁷, foi disponibilizada em 2014 e por isso ainda não apresenta as recomendações atualizadas da WCAG 2.1. As diretrizes WCAG são organizadas em quatro princípios (perceptível, operável, compreensível e robusto) e cada um possui recomendações a serem seguidas. São atribuídos a cada recomendação uma lista de critérios de sucesso (A, AA e AAA). Isto é, há um detalhamento progressivo do que precisa ser feito para alcançar cada recomendação e pelo qual podemos avaliar o nível de conformidade do *site* quanto à sua acessibilidade. Quanto mais específico for o nível (sendo considerado mais específico o nível AAA), maior será o público que conseguirá ter acesso ao conteúdo da página. Para cumprir as recomendações são descritas, por fim, técnicas específicas para a programação do *site*.

Há, ainda, guias para aspectos mais específicos, como para a padronização de mídias presentes na *web*. O *Guia para Produções Audiovisuais Acessíveis*⁸, lançado por um grupo de pesquisadores em Brasília (NAVES *et al.*, 2016), sugere recomendações de padronização de elementos como legendagem para surdos e ensurdecidos, audiodescrição e janela de interpretação de Língua de Sinais. Outro exemplo de um guia mais específico é o documento publicado pelo Governo espanhol intitulado *Guías multimedia accesibles: el museo para todos*⁹ (RUIZ *et al.*, 2008). No texto, há recomendações tanto para o *software* e *hardware* de aplicações interativas que seriam utilizadas durante a visita de exposições e museus, quanto para a acessibilidade física do espaço da exposição.

Entretanto, apesar do número crescente de recomendações, guias e diretrizes ainda são escassos os documentos específicos para acessibilidade de dispositivos móveis. Muitas vezes, para preencher essa lacuna, os desenvolvedores utilizam as mesmas diretrizes WCAG para *sites* e as adaptam ao uso de aplicativos. Em vista disso, a pesquisadora espanhola Olga Carreras Montoto, consultora em acessibilidade *web*, publicou, em fevereiro de 2018, um artigo em seu *blog* onde aborda a acessibilidade em aplicativos nativos do sistema operacional Android¹⁰. Nele menciona requisitos obrigatórios aos aplicativos para serem acessíveis

⁶ A versão mais recente da WCAG, a [versão 2.1](#) (ainda sem tradução oficial para o português).

⁷ A versão mais recente do eMAG, a [versão 3.1](#).

⁸ O [Guia para Produções Audiovisuais Acessíveis](#) em versões em português e inglês.

⁹ O [Guia multimídia acessível: o museu para todos](#) em versão em espanhol.

¹⁰ Postagem de 2018 intitulada "[Apps nativas de Android accesibles](#)".

segundo a norma europeia de Accesibilidad para Productos y Servicios de Tecnologías de la Información y Comunicación (EN 301 549), boas práticas e ferramentas para a revisão de acessibilidade (CARRERAS MONTOTO, 2018).

Esses guias auxiliam o desenvolvedor a compreender que há um extenso número de diferentes necessidades, pois cada indivíduo tem uma diversidade e elas precisam ser contabilizadas e qualificadas antes mesmo do processo de desenvolvimento do *software*. Todavia, essa automatização de recomendações não devem ser a única forma de avaliação da acessibilidade, sendo fundamental contar com o parecer de usuários com diversidade funcional em todas as etapas da criação do produto.

3 CONSTRUÇÃO DO APLICATIVO

A acessibilidade digital é, além de um direito previsto na legislação brasileira às pessoas com diversidade funcional, uma maneira de auxiliar idosos, falantes não nativos, pessoas com baixa escolaridade, novos usuários e pessoas com conexão lenta que usam essas tecnologias (THATCHER *et al.*, 2006), além de ajudar com aspectos do dia-a-dia, pois a usabilidade de um *design* universal e acessível auxilia em um uso mais intuitivo, flexível e com menos esforço (CENTER OF UNIVERSAL DESIGN, 1997). Com isso, verifica-se que a adoção desses produtos e recursos, como tecnologia assistiva, *design* universal, pesquisas de experiência do usuário e recomendações de acessibilidade, em um projeto pode trazer aos desenvolvedores uma maior abrangência de público e, conseqüentemente, um maior retorno financeiro.

Contudo, para desenvolver um projeto de um produto tecnológico acessível é vital ter uma equipe de especialistas em acessibilidade, *design*, programação, entre outras áreas. Para os aplicativos móveis, por exemplo, a equipe de programadores pode escolher o desenvolvimento de uma aplicação específica para uma plataforma, chamada de aplicação nativa (PREZOTTO; BONIATI, 2019). Ou seja, criar um aplicativo que funcione somente em um sistema operacional dentre os vários existentes (Android, iOS, Windows Phone, Symbian, MeeGo etc.). Porém, para um maior alcance de usuários, o ideal seria desenvolver para múltiplas plataformas. O problema é que cada sistema operacional tem uma ferramenta de desenvolvimento (*framework*) própria que requer uma linguagem de programação específica. Por exemplo, no iOS temos a *framework* da Apple com a linguagem Objective-C, enquanto o *framework* do Android usa, tipicamente, o Java (HEITKÖTTER; HANSCHKE; MAJCHRZAK, 2013, p. 122). Portanto, seria necessária uma equipe especialista em várias linguagens de programação para desenvolver e manter cada aplicativo nativo de cada plataforma diferente. Isso tornaria o projeto caro e difícil.

Em virtude dessa necessidade de expansão de mercado de forma menos custosa, surge o desenvolvimento conhecido como multiplataforma (*cross-platform*). Heitkötter, Hanschke e Majchrzak (2013) explicam que a abordagem multiplataforma permite aos desenvolvedores implementar aplicações com um único código-fonte que funcionaria em mais de um dispositivo móvel, podendo ser categorizadas em *webapps* e aplicações híbridas. Atualmente, há várias ferramentas

de desenvolvimento multiplataforma no mercado, *opensource* e comerciais, como Adobe PhoneGap, Rhodes, Apache Cordova, Titanium, MoSync e React Native. Segundo Palmieri, Singh e Cicchetti (2012, p. 180), essas ferramentas trouxeram benefícios como a redução do código, redução de tempo de desenvolvimento, redução de custos de manutenção a longo prazo e aumento de participação do mercado.

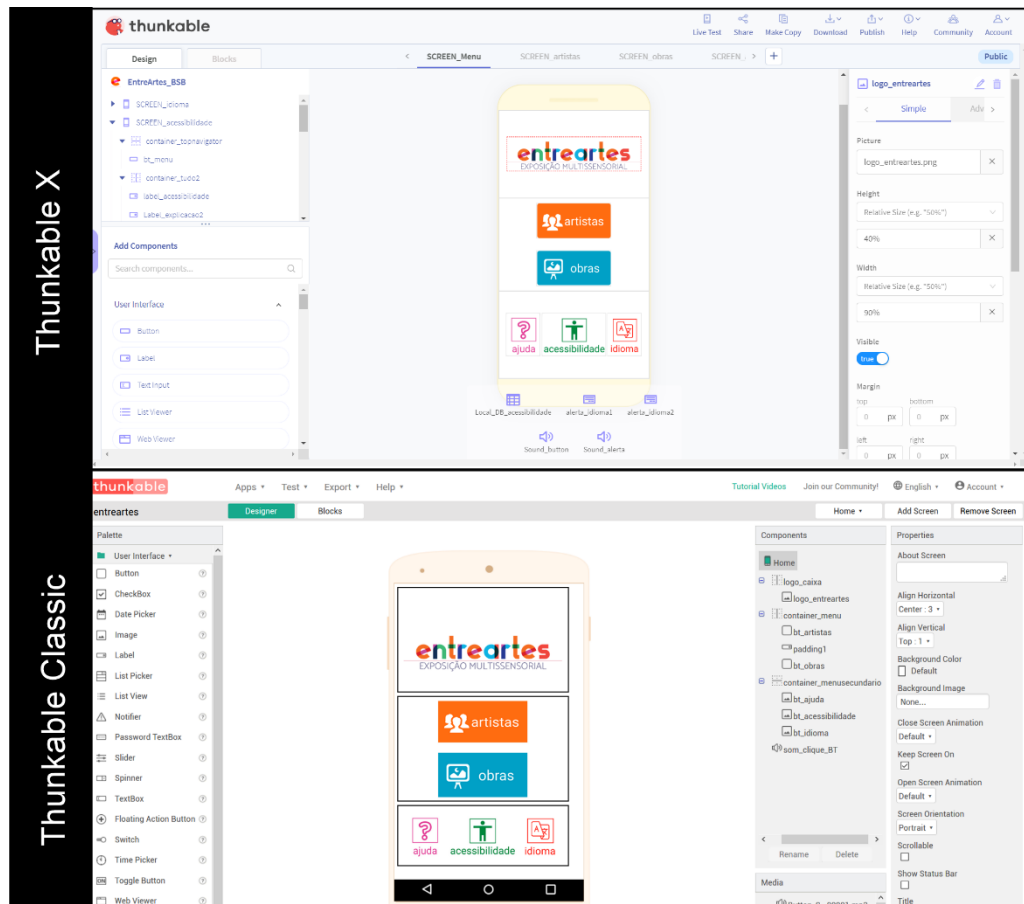
Levando em consideração as dificuldades do aprendizado de tantas linguagens de programação e a competição de aplicativos nativos e multiplataforma, Arun Saigal e WeiHua Li fundaram a companhia Thunkable em 2015, baseados em um projeto incubado no Instituto de Tecnologia de Massachusetts (MIT) com parceria da Google do qual faziam parte. A Thunkable é uma plataforma que permite a não-programadores criar seus próprios aplicativos, pois utiliza um sistema de *drag-and-drop* (arrastar e soltar) para criar a interface da aplicação e um sistema de blocos no lugar da programação (THUNKABLE, s.d.).

A primeira plataforma criada pela Thunkable, em 2015, é a Thunkable Classic for Android, uma ferramenta para o desenvolvimento de aplicativos nativos para aparelhos Android. E em 2018, a companhia lançou uma nova plataforma, a Thunkable X, a qual pode ser usada como um *framework* multiplataforma para os sistemas operacionais do Android e da Apple. Nesse trabalho, utilizamos a plataforma X para o desenvolvimento do aplicativo, pois os aparelhos utilizados na exposição Entreates serão *tablets* da Apple. Contudo, algumas comparações entre as versões Classic e X são feitas durante o tópico de utilização e análise da Thunkable. Podemos observar a interface da Thunkable X e da Thunkable Classic, respectivamente, na Figura 1.

Há uma versão gratuita e uma versão profissional, paga, da Thunkable. Na versão paga, os projetos podem ser privados ou públicos e têm 1Gb de armazenamento de dados, enquanto na versão gratuita os projetos serão públicos e com menos capacidade de armazenamento. Além disso, por ser uma versão profissional, há componentes de publicidade e opções de personalizar a identidade de marca dos aplicativos criados. Testamos os recursos e funcionalidades da plataforma em sua versão gratuita.

Antes de analisar os recursos da plataforma Thinkable, é fundamental descrever a maneira como conduzimos esta investigação. Por esse motivo, expomos no próximo tópico nossa metodologia.

Figura 1 – Interface da Thinkable X e Thinkable Classic



Fonte: Elaborado pela autora (capturas de tela do site da Thinkable)

3.1 METODOLOGIA

A pesquisa é um “processo formal e sistemático de desenvolvimento do método científico” (GIL, 2008, p. 26). Assim que, para desenvolvê-la, é necessário seguir técnicas em cada etapa do processo. Para isso, utilizamos em nossa pesquisa o método científico definido por Karl Popper, conhecido como hipotético-dedutivo. Nele, formula-se um problema, como a falta de aplicativos móveis acessíveis no mercado, e cria-se uma ou mais hipóteses para solucioná-lo. Em nosso projeto, trabalhamos com a hipótese de que a plataforma Thinkable pode democratizar o desenvolvimento de aplicativos acessíveis para não-programadores. Com o teste dessa hipótese, por meio da análise dos recursos da plataforma Thinkable, avalia-se o resultado e refuta-se ou corrobora-se a hipótese inicial. A seguir detalharemos como gerimos cada etapa dessa pesquisa.

Para desenvolver o método hipotético-dedutivo, realizamos uma pesquisa exploratória de abordagem qualitativa aplicada à acessibilidade digital de aplicações móveis no Brasil, pois essa área específica ainda é recente e pouco desenvolvida, como percebemos no estudo das leis brasileiras para assegurar e estabelecer o direito à acessibilidade às pessoas com diversidade funcional. O procedimento técnico utilizado na pesquisa possui as características de um estudo de caso, tendo como objeto os recursos da Thunkable. Com base nas investigações das legislações, barreiras e recomendações sobre a acessibilidade no mundo tecnológico, investigamos esses recursos na construção de aplicativos acessíveis por meio do desenvolvimento de um aplicativo teste e da avaliação desse aplicativo.

Para realizar a análise proposta, procedemos a desenvolver um aplicativo com os recursos oferecidos pela plataforma. Para isso, utilizamos as etapas do processo de desenvolvimento de *software* da área de Engenharia de *Software*. Conhecido como o ciclo de vida de um *software*, esse processo se divide em: levantamento e análise de requisitos, projeto, implementação, testes e entrega (FALBO, 2014). Em um primeiro momento, fizemos o levantamento e a análise dos requisitos, ou seja, das características que o aplicativo acessível necessitava para ser utilizado no evento Entreates.

Esse levantamento se deu através de reuniões com a equipe do projeto, assim como por meio de levantamento bibliográfico documentado no referencial deste trabalho. A análise foi representada por um modelo conceitual, o qual descreve as características do aplicativo. Ademais, em um segundo momento, há a etapa de projeto, na qual elaboramos a prototipagem do aplicativo. A documentação de todo esse processo é fundamental, por isso, os passos estão retratados no capítulo de idealização do aplicativo.

Já nos próximos capítulos, realizamos o desenvolvimento do *app* na plataforma Thunkable, na fase de implementação, e a avaliação do aplicativo pela desenvolvedora e pelos usuários participantes da pesquisa. Esse último passo está previsto no ciclo de vida do *software* como etapa de testes. Para a avaliação dos usuários, duas pessoas com diversidade funcional e experiência com avaliações de acessibilidade, sendo uma surda e uma cega, exploraram o aplicativo em conjunto com as tecnologias assistivas presentes no sistema do iOS e foram entrevistadas, na sua primeira língua, para coletar as impressões sobre a acessibilidade do

produto. Para a avaliação do desenvolvedor, pondera-se sobre as recomendações do eMAG e sobre os dados da avaliação dos usuários. Dessa maneira, respondemos se é possível o desenvolvimento de aplicativos acessíveis com o uso da Thunkable, bem como descrevemos possíveis dificuldades encontradas e recomendamos melhorias e adições de recursos na plataforma.

3.2 IDEALIZAÇÃO DO APLICATIVO

Para desenvolver um *software*, é preciso seguir certos passos pré-definidos na área de Engenharia de *Software*. Neste capítulo, documentamos cada etapa desse processo. Naturalmente, para implementar um *software* para um cliente é necessário conhecer seus objetivos para o projeto, qual questão ele deseja resolver, qual o comportamento esperado para o sistema, qual o custo, tempo e idioma do projeto. Isto é o que chamamos de levantamento de requisitos, pois busca descobrir as necessidades do cliente para aplicá-las ao sistema (FIGUEIRA, 2012). Em nosso caso, precisamos nos familiarizar com o projeto *Entreartes* e os desafios da acessibilidade.

3.2.1 Levantamento e análise dos requisitos

O *Entreartes* é um projeto de extensão vinculado ao grupo de pesquisa e extensão da Universidade de Brasília Acesso Livre. Planeja-se projetar uma exposição acessível para todos os públicos. Com artistas, mediadores e participantes com ou sem alguma diversidade funcional, a equipe espera oferecer oficinas e apresentações multissensoriais de dança, pintura, poesia e escultura, misturando vários sentidos em uma mesma performance. O objetivo do aplicativo seria disponibilizar informações aos visitantes sobre os cinco artistas e as dezoito obras da exposição. Espera-se que o público possa acessar essas informações durante o evento por meio de *tablets* disponíveis em um estande, por conseguinte, a informação precisa estar acessível a todos. Essa seria a questão a ser resolvida com o uso do aplicativo. O *app* é de uso complementar, ou seja, contém informações adicionais para o público e, por isso, não é obrigatório usá-lo para apreciar a exposição.

Segundo Falbo (2014), há várias técnicas de levantamento de requisitos, como entrevistas, questionários, observação, investigação de documentos, entre outros. Neste trabalho utilizamos tanto a entrevista com a equipe para levantar os

requisitos das funcionalidades do aplicativo, quanto a investigação de documentos para conhecer os requisitos de acessibilidade necessários para o público da exposição. Esse levantamento foi documentado como parte do referencial teórico, visto a importância dessa pesquisa para o tema do trabalho. Serão usados como requisitos as recomendações de diretrizes nacionais e internacionais, assim como os princípios do *design* universal (CENTER OF UNIVERSAL DESIGN, 1997).

A análise de requisitos pode acontecer de forma simultânea com o seu levantamento, pois, uma vez que se entendeu as necessidades do cliente, é necessário descobrir requisitos conflitantes, ambíguos, redundantes, esquecidos e duplicados e, assim, solucioná-los (FIGUEIRA, 2012). Um exemplo prático disso é que no levantamento entendemos que o conteúdo seria disponibilizado em português, inglês, espanhol e na Língua de Sinais Brasileira, por isso esses seriam os idiomas requisitados. No entanto, devido a restrições do tempo disponível para o trabalho, esse requisito foi alterado. Desenvolveremos aqui somente a versão do aplicativo em português e Libras. Entretanto, posteriormente espera-se poder expandi-lo para outras línguas.

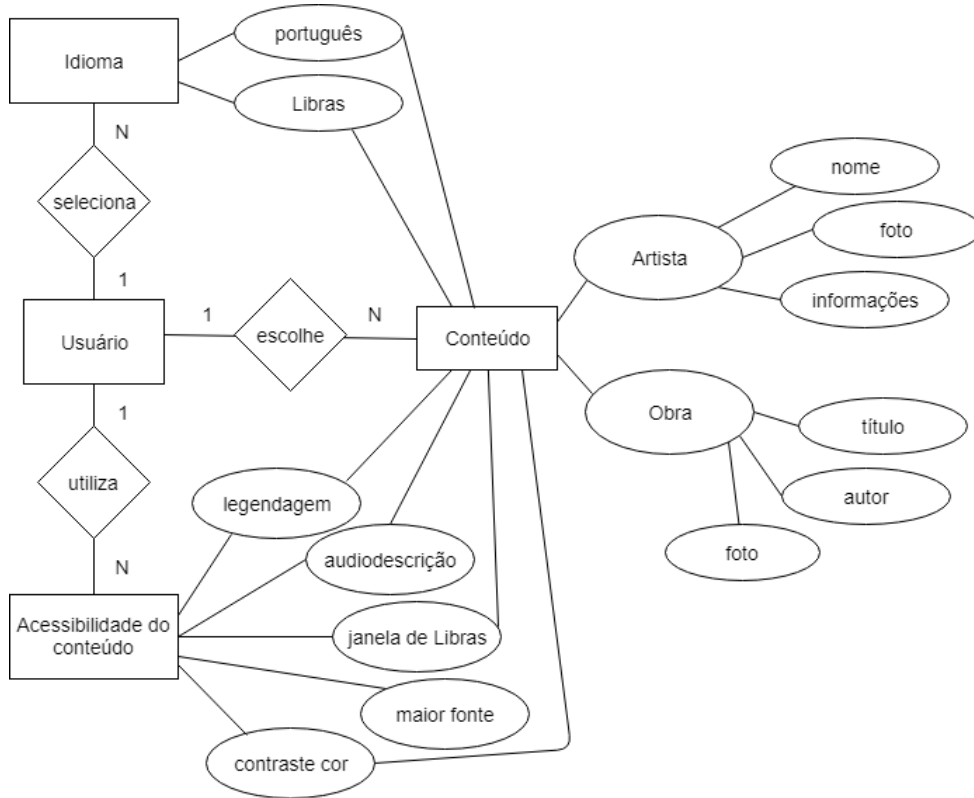
É importante ressaltar que nesse trabalho tratamos somente do *software*. O conteúdo presente nele não é de nossa autoria, nomeadamente os textos, imagens e vídeos com as informações dos artistas e obras. A produção do conteúdo é realizada pela organização do evento, bem como a legendagem, audiodescrição e interpretação das informações para Libras.

Por fim, constatamos que o comportamento almejado do aplicativo seria que os usuários pudessem escolher o conteúdo que irão acessar. Escolhendo, não só os idiomas em que lerão o conteúdo, mas também a forma de apresentação desse conteúdo, se em texto, imagem ou vídeo, e a acessibilidade neles presente, como legenda, audiodescrição e janela de Libras.

Representamos esse comportamento por meio de um modelo conceitual, chamado de Modelo Entidade-Relacionamento (MER), mostrado na Figura 2. Segundo Falbo (2014), o MER é muitas vezes utilizado para a construção de bancos de dados, pois representa os objetos do sistema e o relacionamento entre esses objetos através de um diagrama de entidades e relacionamento (DER). Há vários modelos conceituais na Engenharia de *Software*, porém, escolhemos o MER, pois uma das principais funcionalidades do nosso aplicativo é guardar informações da

exposição Entreates, sendo muito parecida com a funcionalidade de um banco de dados.

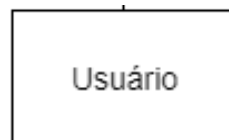
Figura 2 – Diagrama de Entidade e Relacionamento (DER) do aplicativo Entreates



Fonte: Elaborado pela autora

Na Figura 2, podemos perceber que os objetos, chamados de *entidades*, são representados por um retângulo. São elas, “idioma”, “usuário”, “acessibilidade do conteúdo” e “conteúdo”.

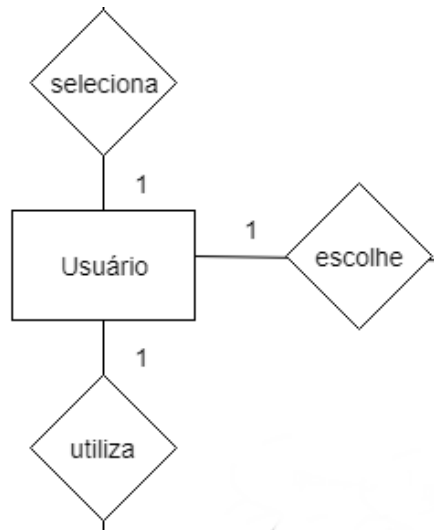
Figura 3 – Exemplo de entidade do DER



Fonte: Elaborado pela autora

O DER descreve o comportamento dessas entidades por meio de relacionamentos entre elas, os quais são exibidos nos losangos que interligam os retângulos. Por exemplo, vemos que a entidade “usuário” tem relações com várias entidades, como, selecionar os idiomas, utilizar a acessibilidade de conteúdo e escolher o conteúdo.

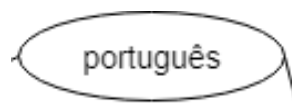
Figura 4 – Exemplo de relações do DER



Fonte: Elaborado pela autora

As entidades podem ter atributos, retratados por elipses ligadas ao retângulo, os quais descrevem as propriedades e características dessas entidades.

Figura 5 – Exemplo de atributos do DER



Fonte: Elaborado pela autora

Por exemplo, a entidade “idioma” possui somente dois atributos, o português e a Libras. Já a entidade “acessibilidade de conteúdo” possui cinco características listadas, representadas pelos atributos de legendagem, audiodescrição, janela de Libras, contraste de cor e maior fonte. O conteúdo tem várias características, herdando atributos das entidades “acessibilidade de conteúdo” e “idiomas” e tendo características próprias como “autor” e “obra”. Os atributos “autor” e “obra” são compostos, o que significa que têm seus próprios atributos – por exemplo, o autor tem seu nome, foto e demais informações e a obra tem seu título, autor e foto.

Também foi definido que o sistema operacional utilizado seria o iOS, visto que seriam concedidos *tablets* da Apple para o evento. Assim, analisamos que a melhor plataforma para desenvolver o aplicativo seria a versão Thinkable X, uma vez que ela é uma multiplataforma para Android e iOS, enquanto a versão Thinkable Classic só desenvolve aplicativos nativos ao Android. É interessante apontar que a Thinkable X é multiplataforma, pois utiliza em seu sistema o React Native, o qual é

uma ferramenta de desenvolvimento de aplicativo multiplataforma que utiliza a linguagem JavaScript.

3.2.2 Prototipagem

A etapa de projetos tem como objetivo fazer um esquema da arquitetura do aplicativo com base nos modelos feitos na etapa anterior de levantamento e análise de requisitos. Para essa etapa, produzimos um protótipo do aplicativo, o que ajudaria o cliente a validar os requisitos e visualizar a interface do projeto e, posteriormente, facilitaria a etapa de implementação em que se programa o aplicativo.

Para essa etapa, nos baseamos nas diretrizes do documento brasileiro sobre acessibilidade digital, eMAG (2014), e nas recomendações internacionais para a acessibilidade de conteúdo *web* e *sites*, expostos nas WCAG 2.1 (2019). Como a implementação do aplicativo tem como cenário uma exposição, utilizamos também o documento *Guías multimedia accesibles: el museo para todos* (RUIZ *et al.*, 2008) para aclarar sobre o uso real do produto pelo usuário em um ambiente de exposições artísticas.

Quando o aplicativo fosse aberto pela primeira vez, a primeira página seria a de configuração do idioma, como mostra a Figura 6. A escolha do idioma é o único processo que se adianta à configuração de acessibilidade (RUIZ *et al.*, 2008). Nessa versão do aplicativo, haverá a opção de seleção da língua portuguesa e da Língua Brasileira de Sinais. Se a segunda for escolhida, o português também ficará ativo como língua de suporte, mas o conteúdo terá uma janela de interpretação em Libras.

Figura 6 –Protótipo da página de configuração de idiomas



Fonte: Elaborado pela autora

Assim que o idioma for escolhido, será feita a configuração dos elementos de acessibilidade na tela seguinte. Como explica Ruiz et al. (2008), essas duas primeiras páginas podem ser configuradas pelo próprio usuário, ou, por um mediador, já que estamos tratando de uma exposição onde o aplicativo será utilizado por vários usuários diferentes. Na Figura 7, percebemos que a opção de janela de Libras está ativada, devido a língua selecionada anteriormente.

Figura 7 – Protótipo da página de configuração de acessibilidade



Fonte: Elaborado pela autora

Só então apareceria a página principal (Figura 8) com a logo do projeto e cinco botões que levam, respectivamente, ao catálogo dos artistas, ao catálogo das obras, à página de ajuda, à página de acessibilidade e à página de idioma. Há uma clara diferenciação entre os botões que levam ao conteúdo (artistas e obras) e os botões de configurações (acessibilidade e idioma) e de ajuda, pois são menus com diferentes funcionalidades. Essa diferenciação deve estar presente tanto no visual dos botões quanto no agrupamento desses elementos no código. Enquanto os botões de conteúdo levam a páginas de listagem de artistas e obras, os botões de configuração de idioma e acessibilidade são as mesmas das páginas iniciais, caso o usuário queira trocar as configurações no meio do uso do aplicativo. E o botão de ajuda leva para um conteúdo de auxílio à exposição, como a programação e a explicação do local físico, além de informações sobre a equipe e uma explicação sobre qual é a serventia de cada recurso de acessibilidade disponível no *app*.

Também foi levada em conta a identidade visual do Entreates utilizando botões coloridos como a logo do projeto e criando uma interface mais agradável. Quanto às cores dos botões, elas não interferem no significado desses elementos, garantindo que os daltônicos não percam nenhuma informação, como explicitado no critério de sucesso 1.4.1 das diretrizes WCAG 2.1 (2018). Além disso, o contraste entre o *background* e as fontes com menos de dezoito pontos e mais de dezoito pontos cumprem o critério de sucesso 1.4.3 de nível AA das diretrizes WCAG 2.1 (2018), sendo a fonte escolhida sem serifa, ou seja, sem as hastes prolongadas em cada letra, pois é a de mais fácil leitura. O tamanho dos botões foi idealizado para fornecer uma maior área de seleção para pessoas com dificuldade de mobilidade e tem pouca precisão no movimento. Ademais, para auxiliar os usuários surdos que não possuem domínio elevado do português, cada botão possui, além de seus títulos, um ícone representativo.

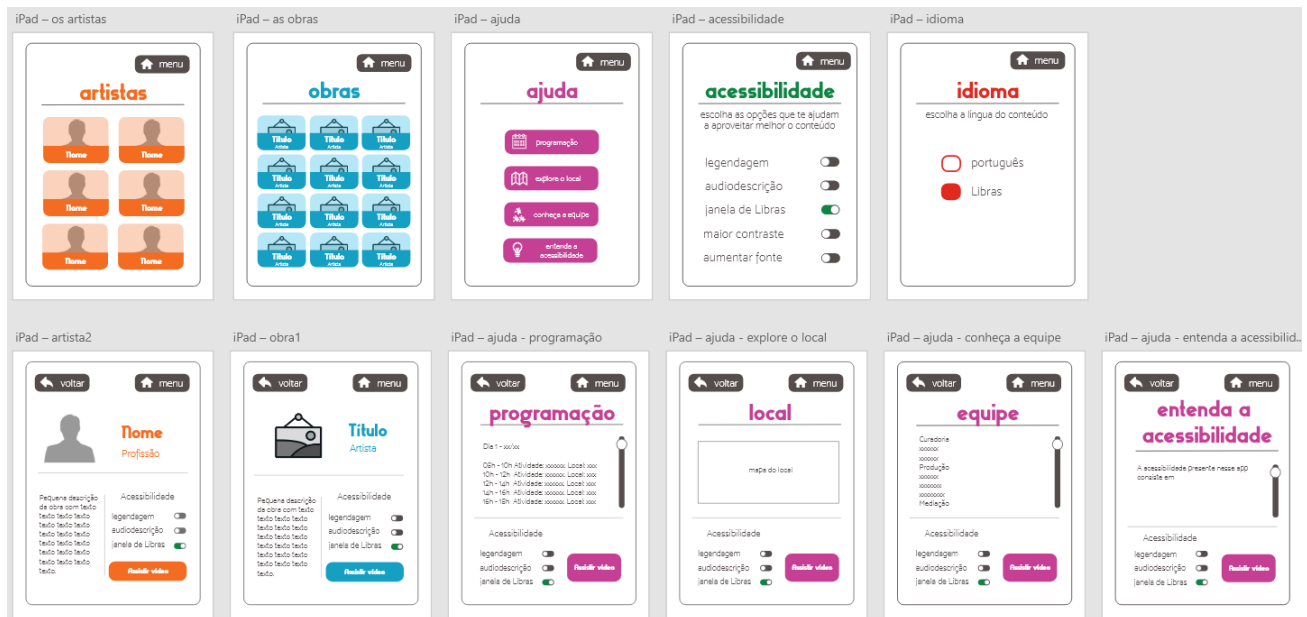
Figura 8 – Protótipo da página principal



Fonte: Elaborado pela autora

Os protótipos das demais páginas do aplicativo foram feitos com o princípio de consistência em mente, como podemos ver na Figura 9. Usamos as mesmas posições, fontes e cores para fazer com que a interface se torne rapidamente familiar ao usuário e, assim, possibilitar uma navegação mais clara e intuitiva. Podemos destacar, no menu de “ajuda”, a página de descrição dos recursos acessíveis como uma forma de tornar a documentação mais acessível, como explica Kavcic (2005, p. 1.026).

Figura 9 – Protótipo das demais páginas do aplicativo Entreates



Fonte: Elaborado pela autora (captura de tela)

3.3 INVESTIGAÇÃO DOS RECURSOS DA THINKABLE

Na etapa de implementação, fizemos, simultaneamente, o desenvolvimento do *app* e a análise dos recursos da plataforma Thinkable X. Assinalamos que a proposta não é avaliar se a plataforma em si é acessível aos desenvolvedores que a utilizam, mas, sim, testar se os recursos disponíveis na Thinkable têm a capacidade de desenvolver um aplicativo acessível a todos os usuários. No desenvolvimento do aplicativo, ponderamos sobre as maneiras de implementar as diretrizes do eMAG (2014) e da WCAG 2.1 (2019) propostas no protótipo. Citaremos também as recomendações presentes no artigo sobre acessibilidade em aplicativos nativos do sistema operacional Android de Carreras Montoto (2018). Além disso, durante o processo de desenvolvimento do aplicativo demos enfoque em: tela, sistema e conteúdo, sons e multimídia, teclado e dispositivos apontadores e verificação de acessibilidade e documentação, pois são as principais áreas de dificuldades no desenvolvimento de *softwares* acessíveis, segundo Kavcic (2005, p. 1.026).

Uma das primeiras coisas a se fazer antes de começar a trabalhar com um *framework* é ler sua documentação. Com a documentação, o desenvolvedor fica ciente das funções e recursos atuais da ferramenta em questão. Na documentação da Thinkable, são explicados a lógica dos blocos que substituem a programação, assim como os componentes presentes na plataforma. Percebemos que há intenções de acrescentar mais recursos na versão Thinkable X, pois é a versão

mais recente da plataforma. Isso deixa espaço para o crescimento de uma gama de recursos, inclusive aqueles que auxiliam a acessibilidade do aplicativo. Um exemplo de um novo recurso seria a rotação de tela, a qual está prevista, segundo a documentação, para ser implementada em breve na versão X. Atualmente, só se pode desenvolver um aplicativo na orientação retrato na versão X, enquanto na Classic há opções para orientação retrato e paisagem. Na documentação atual, porém, não há nenhuma menção à acessibilidade ou a futuros recursos para contribuir com essa área.

Já dentro do programa, há duas telas principais: a tela de *design* e a de montagem dos blocos lógicos. Na tela de *design*, onde fazemos a edição da interface do aplicativo, percebe-se que a disposição de conteúdo é de *drag-and-drop*, o que torna mais fácil a organização da interface concebida no protótipo. Há como escolher a tipografia por estilo serifado e sem serifa, sendo as fontes sem serifa mais claras para leituras rápidas. Pode-se também escolher as cores pelos valores HEX ou RGB, sendo assim, há uma gama de cores a serem escolhidas, tornando possível cumprir as taxas de contraste estabelecidas pelas diretrizes WCAG 2.1 (2019). Entretanto, os desenvolvedores podem não conhecer as taxas de contrastes da WCAG e isso pode levá-los a escolher cores sem grande contraste entre si. O ideal seria, então, disponibilizar, além das game de cores, um aviso aos desenvolvedores sobre a taxa de contraste entre a paleta utiliza do *background* e na tipografia do *app* ou levá-los a sites de terceiros que verificam o contraste de cor.

Apesar de poder escolher a tipografia e as cores presentes no aplicativo, a Thinkable não anula as configurações atuais do dispositivo. Isto é, se as configurações do aparelho estiverem ligadas para, por exemplo, fontes de tamanhos maiores ou inversão de cores, elas substituirão as fontes e cores do aplicativo para as preferências e necessidades específicas daquele usuário.

Quanto aos componentes do sistema em si (como botões, notificações, interruptores, caixa de entrada de texto, imagens etc.), todos deveriam apresentar seu estado, função, valor e etiqueta segundo a recomendação 1.10 de Carreras Montoto (2018). Na versão X e Classic, todos os componentes apresentam seu estado ao leitor de tela, ou seja, quando focado pelo leitor, lhe será informado sua propriedade. Por exemplo, o leitor dirá se o interruptor está ligado, desligado ou, ainda, desabilitado. Contudo, na versão X, eles não apresentam sua função ao leitor

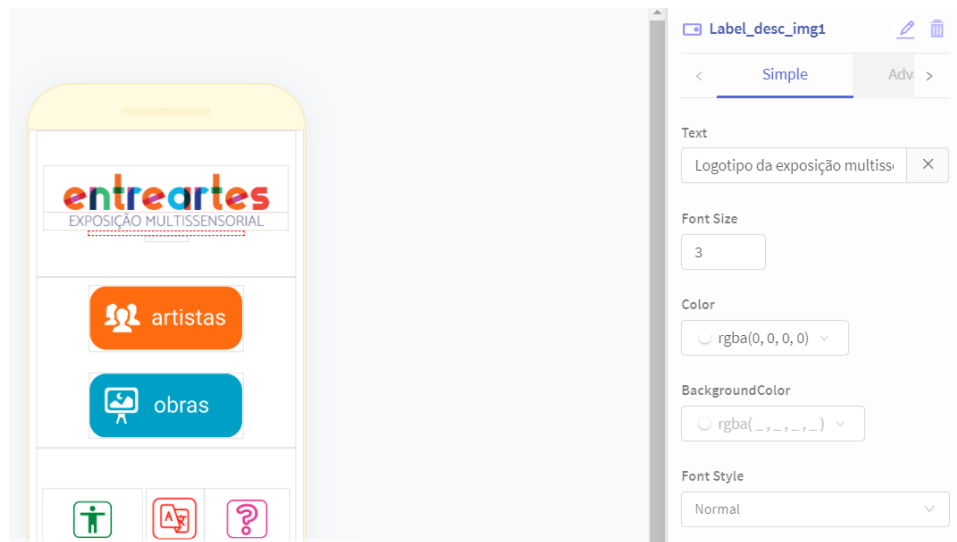
de tela. Ou seja, o leitor não informa ao usuário se o elemento em foco é um botão ou uma caixa de entrada de texto, por exemplo, e, com isso, o usuário fazendo uso do leitor não saberá que aquele elemento é interativo e qual interação deveria ser feita. Enquanto isso, na versão Classic da plataforma os componentes apresentam sua função ao leitor de tela. Quanto ao valor, todos os componentes o informam, fazendo com que o leitor de tela diga o texto contido em um campo de texto, por exemplo. Além disso, não há como etiquetar os componentes em ambas as versões. Essa é uma característica imprescindível para imagens, pois poderíamos descrever seu conteúdo dessa maneira.

Há uma única diferença do componente de imagem entre as versões. Na versão Classic, o leitor de tela informará ao usuário que há uma imagem (pois informa sua função), porém não a descreve (pois não há etiqueta), enquanto, na versão X, o leitor de tela ignorará as imagens (sem função e etiqueta). Há uma opção para imagens na versão X chamada de “acessibilidade”. Se marcada, oferecerá a função da imagem e o leitor de tela passará a reconhecê-la, mas ainda não oferece uma forma de descrevê-la. Uma das maneiras encontradas de personalizar esse componente para que fosse mais acessível foi adicionar um componente de texto (*label*) com a cor do texto transparente abaixo da imagem e nele colocar sua descrição. Logo, o leitor de tela pulará a imagem, mas focará no componente não-visível de texto que contém sua descrição.

Os botões, além de não apresentarem sua função de botão, não possuem configurações de edição que permitem adicionar um ícone ao lado de seu texto, como havíamos concebido no protótipo. Há, porém, como adicionar uma imagem como *background* do botão em vez de uma cor sólida. Para contrapor a falta de função desse componente, decidimos adicionar todo o *design* do botão, sendo ícone e texto, como uma imagem de *background* e utilizamos o texto do botão em cor transparente para expor sua função ao leitor, como apresentado na Figura 10. Assim, quando em foco, o leitor de tela lerá o texto do botão onde se lê “botão” e seu título, mas não afetará a aparência do botão. Carreras Montoto (2018) explica que se criamos componentes personalizados precisaremos torná-los acessíveis de forma manual, definindo todas suas funcionalidades e informações. É uma abordagem criativa, porém que demanda muito mais trabalho e tem suas restrições, sendo ainda o melhor método configurar sua acessibilidade pelo código. Além do mais, como o

botão é responsivo, o *background* ficará maior em aparelhos de maior tamanho, assim, a fonte também ficará maior. Contudo, caso o usuário selecione a configuração do aparelho para uma fonte maior, o texto do botão não aumentará, pois é uma imagem.

Figura 10 – Componente personalizado para apresentar a função de imagens ao leitor de tela



Fonte: Elaborado pela autora (captura de tela)

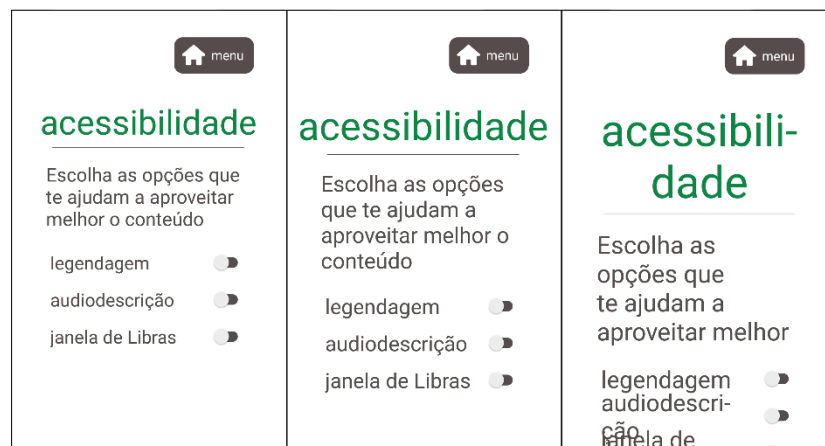
Para tornar os botões ainda mais acessíveis aos surdos, seria importante oferecer, além das cores e ícones, vídeos em Libras que explicam a funcionalidade do botão. Em alguns *websites* acessíveis esses vídeos são colocados em formato *gif*, por serem leves e capazes de se repetirem do início, e implementados no código pelo método “hover”, o qual possibilita que o vídeo em *gif* apareça quando o usuário passar o *mouse* pelo botão. Assim, o botão teria duas ações, o clique para ir a uma nova página e o “hover” para explicar sua função aos usuários surdos. Porém, esse método não existe em codificação de *smartphones*, uma vez que o toque na tela equivale ao clique e não há como somente passar o dedo sem desencadear esse clique. O desafio de implementar esses *gifs* para a acessibilidade da língua de sinais em aplicativos ainda se mantém para todas as ferramentas de desenvolvimento com ou sem o uso da programação.

Quanto à disposição dos componentes, eles são organizados em caixas chamadas linhas e colunas (*rows* e *columns*). Essas caixas são ignoradas pelo leitor de tela e só servem para o agrupamento visual dos componentes que estão dentro dessas caixas, e não para um agrupamento funcional. Não há nenhuma maneira de agrupar funcionalmente os componentes com os blocos lógicos da Thunkable

Classic e X, por exemplo, agrupando todos os elementos do topo da página ou todo o conteúdo de temática similar. Tampouco há como colocar uma hierarquia funcional nos títulos, ou seja, sem ser somente pelo aspecto visual da diferença de tamanho entre títulos. Por conseguinte, não cumpre as recomendações 1.3 e 1.8 do eMAG (2014), o que torna a organização do conteúdo menos clara para um usuário do leitor de tela. Como não há acesso ao código, não podemos afirmar como essas caixas são programadas, porém a dificuldade de agrupamento funcional dos elementos nos aponta para uma possível diagramação por tabela. A disposição do conteúdo por meio da programação de tabelas (*tables*) é um método obsoleto e não acessível, pois dificulta o sequenciamento lógico do conteúdo pelo leitor de tela.

Outra característica da disposição dos componentes adotado na Thunkable X é o emprego de *padding*s e *margins* para auxiliar na interface do aplicativo. Esses elementos de *design* são utilizados para criar espaçamento entre componentes e não existiam na Thunkable Classic. Também há como colocar tamanhos relativos (*relative size*) nos componentes, exceto nas fontes. Isto significa que o aplicativo será responsivo, ajustando seu tamanho para a dimensão do aparelho, podendo ser utilizado tanto em celulares quanto em *tablets*. Todavia, como as fontes ainda não podem ter tamanhos relativos na Thunkable X, elas não serão responsivas e nem sempre o texto configurado em uma fonte maior no dispositivo caberá em seu contêiner, ficando cortado na tela. Na Figura 11 comparamos como três diferentes configurações de tamanho de fonte do celular Android afetam uma das telas do aplicativo.

Figura 11 – Comparação da tela do aplicativo com diferentes configurações de fonte do Android



Fonte: Elaborado pela autora (capturas de tela)

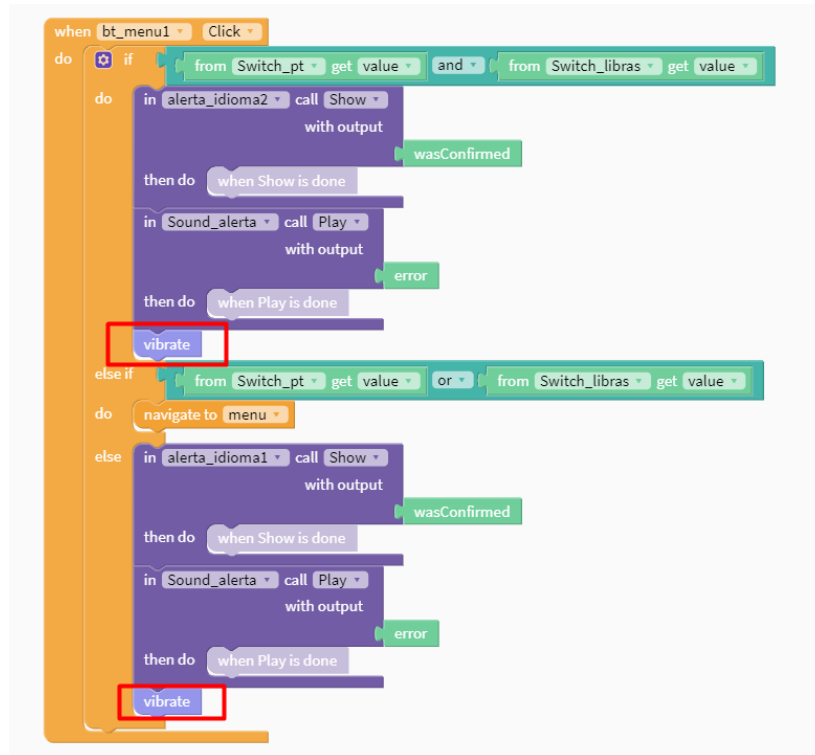
Segundo a recomendação 3.1 do eMAG (2014), o idioma principal do produto deve ser identificado por meio do código, porém não há como fazê-lo com os recursos da Thinkable. Assim o leitor de tela não consegue reconhecer se o texto ou um termo presente no aplicativo está em uma língua diferente da língua de configuração do dispositivo. Na versão Classic, há a opção de adicionar extensões criadas por usuários da comunidade Thinkable, por isso pode haver extensões que ajudem a acrescentar recursos que auxiliam na acessibilidade, porém essa opção ainda não está disponível para a versão X.

Em relação a sons, a plataforma Thinkable possui componentes para inserir sons e para entrada de voz. Ambos podem auxiliar diferentes diversidades funcionais. Sons auxiliam cegos, pois são uma pista das ações que ocorrem visualmente na tela, como no apertado de um botão. E a entrada de voz pode auxiliar na inserção de dados para pessoas com pouca mobilidade, porém, deve ser opcional já que não é acessível para todas as pessoas, por exemplo, surdos não-oralizados.

Na versão X, há como empregar nos blocos a vibração do dispositivo (*vibrate*) outra forma de oferecer alertas não-visuais aos usuários, como mostra a Figura 12, que apresenta um pedaço dos blocos do nosso projeto interligado ao bloco de vibração. Outros recursos multimídia são os vídeos, que podem ser acrescentados dentro do próprio banco de dados oferecido pela plataforma ou pelo YouTube. Ambas as opções são boas e oferecem ao desenvolvedor a opção de adequar o conteúdo de acordo com seu aplicativo.

Ademais, há alguns componentes que devem ser evitados, por exemplo, o recurso de animação. A recomendação 5.5 do eMAG (2014), explica que animações só devem ser utilizadas se também forem fornecidas formas de controle sobre elas, como botões de pausa e início. Por fim, não testamos o aplicativo com adaptações físicas ou de *hardware*, por isso não poderemos analisar a acessibilidade da plataforma em relação a dispositivos apontadores e teclados. Assim que, para facilitar a realização de tarefas para pessoas com pouca mobilidade, criamos botões com uma ampla área de clique.

Figura 12 – Bloco de vibração presente na plataforma Thinkable X



Fonte: Elaborado pela autora (captura de tela)

3.4 AVALIAÇÃO DO APLICATIVO COM USUÁRIOS

Logo após a etapa de implementação, executa-se a etapa de testes. O resultado dessa etapa pode vir a determinar que o produto passe por uma nova etapa para correção de erros antes da sua entrega final. Segundo Falbo (2014), o teste de um software verifica se seu comportamento está de acordo com o esperado pelo cliente. Devido ao enfoque desse trabalho, realizamos a avaliação do aplicativo com dois usuários com diversidade sensorial para examinar, principalmente, os requisitos estabelecidos na etapa de levantamento de requisitos, como as recomendações de acessibilidade do eMAG (2014) e WCAG 2.1 (2019), além da avaliação da interface com base nos princípios do *design* universal (CENTER OF UNIVERSAL DESIGN, 1997) e da experiência do usuário.

Ambos os usuários são membros do grupo de pesquisa e extensão Acesso Livre da Universidade de Brasília e fazem parte da equipe de acessibilidade do projeto Entreates, além de terem experiência em avaliações voltadas a mensuração da acessibilidade de ambientes, programas e afins. Optou-se pelo método de entrevista para permitir que as respostas fossem na primeira língua de cada participante, visto que um formulário *online* dificultaria respostas na língua de sinais.

Em um primeiro momento, foi explicado aos usuários, em português e Libras, qual o tema do trabalho, o que era o aplicativo e seu papel no evento Entreates, também foi deixado claro que o foco dessa avaliação é a funcionalidade do aplicativo em si, e não o conteúdo do aplicativo, como os vídeos nele presentes. Em um segundo momento, houve a experimentação com o *tablet* por parte dos avaliadores, onde especificamos que realizassem duas tarefas mínimas: a) selecionar as configurações de sua preferência (idioma e acessibilidade) e chegar ao menu; e b) entrar na página “artista”, escolher um artista e assistir o vídeo sobre ele. Por fim, observamos os usuários em uma exploração livre das demais páginas do aplicativo.

Em relação às perguntas, a avaliação do aplicativo pelos usuários acaba sendo mais voltada para a sua experiência em relação à interação com o aplicativo do que para suas especificidades técnicas. A entrevista, feita depois da exploração do aplicativo, tinha cinco perguntas base:

- a) A navegação pelo aplicativo é fácil e intuitiva?
- b) A interface do aplicativo é clara e organizada?
- c) Como você avalia a experiência que teve usando o aplicativo?
- d) O aplicativo a/o ajudaria a aproveitar melhor a exposição no dia do evento Entreates?
- e) Como você avalia a acessibilidade geral do aplicativo?

Para o teste, tivemos a disposição um *tablet* com sistema operacional iOS 10.2, que oferece como tecnologia assistiva o VoiceOver, o leitor de tela próprio do iOS. Apresentamos a seguir as respostas da primeira avaliadora, juntamente com comentários nossos feitos com base nas observações durante a exploração do *app* pela usuária.

Com o auxílio do VoiceOver, a avaliadora cega considerou a navegação pelo aplicativo mediana, pois teve dificuldades em perceber quando chegava ao final da página e a que os interruptores das páginas de configuração estavam interligados. É de se esperar que um aplicativo acessível retorne ao começo da página quando ela acabar, mas, como não há maneira de especificar o foco para o leitor de tela com a Thinkable, essa ação não se realiza. Além disso, os interruptores para escolher o

idioma e a acessibilidade se tornaram empecilhos, pois a avaliadora não conseguia saber qual interruptor acionava determinada opção. Isso seria evitado caso esses elementos estivessem agrupados, recurso não disponível na Thunkable.

A interface do *app* foi julgada como boa, devido à boa disposição dos menus, além de ser fácil de aprender a mexer, pois tem poucos botões, permitindo encontrar rapidamente o que se está buscando. Como fatores que influenciaram positivamente a experiência de utilização do aplicativo, foram citados a voz do leitor de tela do iOS e a supressão de informações desnecessárias, como propagandas. Contudo, afetou negativamente a experiência desta avaliadora a falta de escolha da velocidade da voz do leitor e a falta do botão para voltar. A velocidade da voz poderia ser controlada no próprio aparelho, saindo do aplicativo e indo para as configurações. Porém, o usuário pode não ter acesso a essas configurações em um aparelho público de um museu, portanto, uma possível solução para essa questão seria pedir aos mediadores responsáveis que ligassem e desligassem o leitor de tela no dia do evento e que deixassem em aberto aos usuários a escolha da velocidade da voz. O botão para voltar pode ser considerado um equívoco na prototipagem, pois, apesar de haver um botão de voltar e outro botão de ir ao menu em certas páginas, outras só contavam com o botão de ir ao menu. Isso se deve ao fato de que o botão de voltar nessas páginas fora considerado redundante, uma vez que levaria ao menu, e por isso foi suprimido. Porém, como percebeu-se com a avaliação, prefere-se que haja a coexistência dos dois botões mesmo nas páginas em que sejam redundantes.

Finalmente, a usuária avaliou a acessibilidade geral do aplicativo com uma nota 3 em uma escala de 1 a 5, sendo 1 para uma acessibilidade ruim e 5 para uma muito boa. E, sobre a quarta pergunta, ela respondeu que o *app* a ajudaria a aproveitar melhor a exposição, apesar dos obstáculos encontrados. Observamos, como desenvolvedores e pesquisadores, que a nota atribuída a acessibilidade não é considerada baixa. Especulamos que essa nota reflita as demais vivências da usuária, que, habituada a enfrentar cotidianamente sites sem nenhuma acessibilidade, considera razoável um *app* que apresenta diversos empecilhos, muitos dos quais já tinham sido presumidos durante a etapa de análise dos recursos da plataforma. Teorizamos, ainda, que a nota possa expressar emoções positivas da usuária para com um aplicativo que contou com uma preocupação sobre a área de acessibilidade em seu desenvolvimento.

A seguir, relatamos e comentamos a exploração do avaliador surdo, o qual não utilizou nenhuma tecnologia assistiva. Durante sua utilização, a navegação lhe pareceu simples, pois os textos e palavras eram curtos e descomplicados e tinha a opção de escolher entre legendas e janela de Libras para o conteúdo. A interface colorida idealizada na prototipagem agradou o usuário, pois, segundo ele, ajuda a despertar o interesse e a diferenciar os menus.

Outrossim, o usuário considerou sua experiência como primariamente positiva, mas sugeriu que houvessem a sinalização na língua de sinais dos títulos dos menus e uma página onde os usuários pudessem dar sugestões de melhorias ao aplicativo. Também considerou que o aplicativo seria útil no dia da exposição e frisou ser muito importante ter esse tipo de recurso disponível em todas as exposições e museus. Na última pergunta, o usuário conceituou o aplicativo como acessível e, com o desenvolvimento e mudanças sugeridas, ganharia uma nota 5, significando uma acessibilidade muito boa.

Nota-se uma discrepância entre a experiência dos dois usuários para com o aplicativo, o que gera também uma diferença na sua acessibilidade final. Consideramos que essa diferença é, principalmente, devida às questões técnicas de desenvolvimento com a Thunkable. O usuário surdo não utiliza o leitor de tela e se apoia no visual. Nesse quesito, a Thunkable atende o resultado esperado, pois o desenvolvedor consegue criar uma interface chamativa e organizada visualmente. O usuário cego, no entanto, não vivencia o mesmo cenário e sofre com a falta de precisão ao navegar com o leitor de tela. Devemos registrar que o usuário surdo tem domínio do português, o que, independentemente da proposta acessível, também auxilia na navegação.

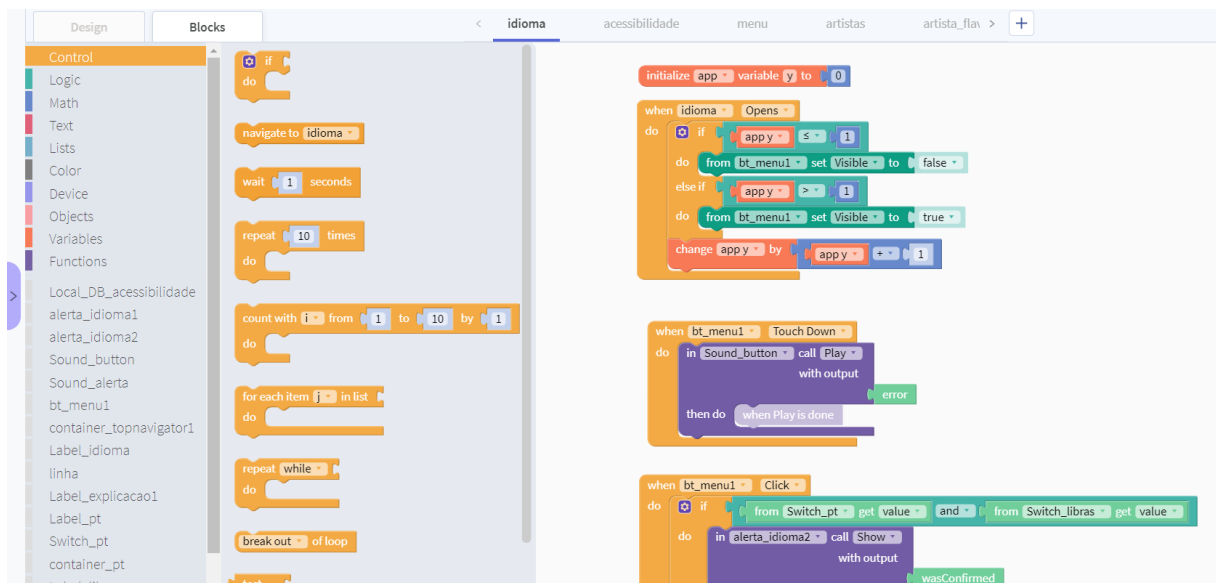
Na visão da desenvolvedora, essas dificuldades também ocorreram devido à ausência de certos recursos na plataforma, como a hierarquização de títulos e o agrupamento de conteúdo, que poderiam ser adicionados caso o aplicativo fosse programado com outros *frameworks*. Quanto à apresentação de etiquetas e funções dos componentes como os botões, os métodos personalizados supriram bem as necessidades do projeto, apesar de não serem soluções ideais a longo prazo, pois criam outras restrições, como no redimensionamento da fonte, quando mudada pela configuração de tamanho presente no dispositivo móvel.

3.5 AVALIAÇÃO DA THUNKABLE PELA DESENVOLVEDORA

Levando em consideração as características acessíveis esperadas para o aplicativo, a análise dos recursos da plataforma Thinkable X e o teste dos usuários, realizamos a avaliação final da plataforma quanto à sua capacidade de desenvolver aplicativos acessíveis. Separamos a avaliação em seis áreas compatíveis com as seções de recomendações do eMAG (2014): marcação, comportamento, conteúdo/informação, apresentação/*design*, multimídia e formulário.

A primeira seção do eMAG explora a marcação, ou seja, faz recomendações sobre o código HTML. E, apesar de a linguagem usada pela Thinkable X ser o JavaScript, consegue-se fazer alguns paralelos. Usando a plataforma, não se tem acesso ao código que se produz, somente aos blocos. Mostramos na Figura 13 a tela de montagem dos blocos, onde podemos escolher entre vários blocos pré-estabelecidos pela Thukable X para criar uma conexão lógica que funcionaria para ativar os elementos da tela de *design* da interface.

Figura 13 – Tela de montagem de blocos da plataforma Thinkable X



Fonte: Elaborado pela autora (captura de tela)

Por isso, não temos controle sobre a organização lógica e semântica do código, como espera a recomendação 1.2 do eMAG (2014). Ademais, a utilização de blocos lógicos facilita o desenvolvimento do aplicativo para pessoas sem o conhecimento de programação, porém, não possibilita a autonomia de acrescentar elementos que propiciam um conteúdo mais acessível, pois os blocos já são pré-determinados. Seria interessante disponibilizar, futuramente, blocos para suprir as necessidades encontradas nessa área de acessibilidade, como para níveis de

cabeçalho e agrupamento de conteúdo. Essa é a área em que se tem menos controle e, por isso, não conseguimos criar alternativas personalizadas para facilitar a acessibilidade do usuário.

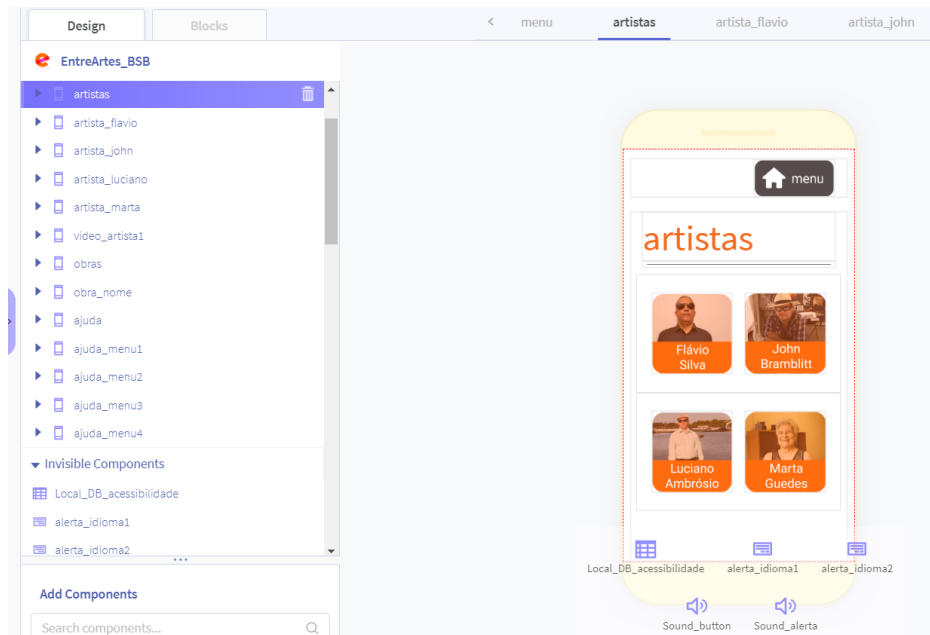
Quanto ao comportamento do aplicativo, o desenvolvedor tem controle sobre situações como limite de tempo e atualizações de páginas com o uso dos blocos na Thinkable X. Isso nos leva a crer que o comportamento seria acessível em casos que necessitassem o uso das recomendações 2.3 e 2.5 do eMAG (2014). Porém, não conseguimos testar o aplicativo com adaptações físicas e externas e não sabemos seu comportamento nessas circunstâncias.

Encontramos alguns empecilhos para tornar o conteúdo e as informações do aplicativo acessíveis com a Thinkable X. Não há como identificar o idioma principal do *app*, por exemplo. E, *a priori*, não existe um espaço próprio para textos alternativos com o intuito de descrever as imagens. Apesar disso, é possível criar elementos personalizados para a descrição de imagens com os recursos disponíveis na plataforma. Com relação à criação de textos curtos e com um vocabulário mais acessível, essas são recomendações que independem da plataforma e não encontram obstáculos ao serem implementadas na Thinkable X.

Apesar dos entraves encontrados em outras seções, é possível se adequar de forma satisfatória aos requisitos das seções de apresentação/*design* e multimídia do eMAG (2014) utilizando a plataforma Thinkable. Podemos ver na Figura 14 que pudemos criar uma interface semelhante à proposta no protótipo em relação às cores, estilos de fontes, adição de imagens e disposição de elementos.

O único contratempo na seção de apresentação/*design* seria a recomendação 4.3, sobre o redimensionamento de elementos. Há como redimensionar todos os elementos com a Thinkable X exceto as fontes. Isso pode vir a causar dificuldades de leitura em textos aumentados com a configuração do celular. As demais recomendações de multimídia sobre cores e alternativas de vídeo e áudio são possíveis de serem adotadas para aplicativos desenvolvidos na plataforma. Quanto à seção de formulários, percebe-se que a Thinkable não oferece meios de agrupamento de conteúdo e de apresentação de funções e etiquetas dos componentes. E, apesar de ser possível criar métodos personalizados para as funções e etiquetas, são alternativas que apresentam desvantagens.

Figura 14 – Tela de design da página de artistas do aplicativo Entreates na plataforma Thunkable X



Fonte: Elaborado pela autora (captura de tela)

Em suma, a plataforma Thunkable é uma boa iniciativa e possui qualidades na criação de interfaces organizadas e atrativas, além da inserção de elementos multimídia. Contudo, ainda necessita desenvolver aprimoramentos acessíveis para seus recursos e para o agrupamento desses elementos. Blocos para identificação de idioma e para redimensionamento de fontes também são acréscimos desejáveis. Atualmente, a plataforma não pode ser considerada capaz de suportar aplicativos acessíveis a todos os usuários, tendo mais opções para a acessibilidade para os surdos, com os recursos midiáticos, do que para cegos, por exemplo.

Destaca-se que houve indício de interesse por parte da equipe da plataforma de aprimorar a acessibilidade de seus recursos. A Thunkable oferece um espaço para discussões de dúvidas e sugestões de melhorias em sua página da comunidade, na qual fizemos questionamentos sobre a acessibilidade e obtivemos respostas proativas da própria equipe de desenvolvimento da plataforma. Por isso, espera-se que a plataforma acrescente mais recursos acessíveis à medida que ela se desenvolva e que haja uma maior procura por essa área pelos próprios desenvolvedores que utilizam a Thunkable X.

4 CONSIDERAÇÕES FINAIS

A expansão de *smartphones* traz consigo conveniências e facilidades na execução de atividades do cotidiano, pois podemos levá-los a qualquer lugar com muita facilidade. Não obstante, essa utilidade ainda não é desfrutada por todos os usuários, pois um grande número de aplicativos e *softwares* presentes nesses dispositivos não são acessíveis. Atualmente, algumas grandes empresas de *smartphones* começaram a inserir tecnologias assistivas em seus sistemas operacionais, como leitores de tela e lupas, para uma melhor experiência e usabilidade de usuários com diferentes preferências, usos e diversidades funcionais. Todavia, para utilizar esses recursos em todo seu potencial é necessário que os aplicativos nesses dispositivos possibilitem, por meio de seu código, que as TAs funcionem adequadamente.

Neste trabalho, exploramos os recursos da Thunkable, uma plataforma que propõe democratizar a criação de aplicativos por meio do uso de blocos lógicos que substituem a necessidade de programação. Investigamos se esses recursos conseguem auxiliar a acessibilidade e viabilizar o uso de tecnologias assistivas, tornando o aplicativo acessível a todos os usuários. Algumas limitações enfrentadas durante a execução do trabalho foram restrições de tempo e falta de acesso a tecnologias assistivas de natureza física, como adaptações de *hardware* para o celular. Assim, o *app* criado para avaliar a acessibilidade da Thunkable não pode ser testado por usuários que necessitam dessas adaptações. Almejamos poder abordar essas questões em futuros projetos, contanto com um aplicativo com mais línguas e um maior número de avaliadores.

A performance do aplicativo Entreates¹¹, criado para avaliar os recursos da Thunkable, foi examinada pela desenvolvedora e por dois usuários que são mediadores de acessibilidade do evento Entreates, sendo um surdo e outra cega. Averiguou-se que há uma diferença entre as experiências de ambos os usuários causada pelas diferentes necessidades de cada um. Os empecilhos enfrentados pela usuária cega foram relacionados ao uso do leitor de tela, o que fornece informações para concluir que não há recursos suficientes na plataforma Thunkable para que o usuário experimente uma acessibilidade total nesse quesito, ao passo

¹¹ O [projeto do aplicativo do Entreates](#), criado na plataforma Thunkable X, é público e está disponível para visualização de todos.

que o usuário surdo teve uma experiência com menos barreiras, pois os recursos de interface da plataforma oferecem meios de criar um *design* universal. Concluímos que a plataforma ainda não apresenta recursos suficientes para uma acessibilidade apropriada ao uso de todos os usuários, porém, é possível que esses recursos sejam adicionados no futuro visto que a Thinkable X é uma versão ainda nova e em expansão. É essencial que os próprios desenvolvedores que utilizam a Thinkable X para criar aplicativos se empenhem para abrir discussões sobre acessibilidade e reivindiquem esses novos recursos.

Sugerimos que a Thinkable X adicione alguns artifícios de acessibilidade para seus recursos e melhorias no código, como: código acessível que apresente as funções dos componentes como botões, imagens e alertas para leitores de tela; elementos de edição do componente de imagem para que o desenvolvedor possa adicionar uma descrição a cada imagem e, assim, auxiliar usuários cegos; avisos sobre as recomendações sobre contraste de cor da WCAG perto do recurso de escolha de cor para melhor informar os desenvolvedores; recursos para agrupamento dos componentes para facilitar a sequência lógica seguida pelo leitor de tela; blocos para marcar a hierarquia dos títulos também para sequenciamento lógico; blocos para identificação de idioma para uso do leitor de tela; blocos para redimensionamento de fontes para auxiliar a responsabilidade dos aplicativos; e adicionar o tópico de acessibilidade na documentação da plataforma para conscientização dos desenvolvedores. A acessibilidade digital é uma área que precisa ser difundida, pois torna possível a todas as pessoas o uso e o acesso a comunicação, informação e oportunidades trazidas pela tecnologia e internet. Esperamos que este trabalho colabore com a disseminação e a conscientização de mais pessoas para com o tema, aumentando tanto a realização de estudos com a temática, quanto a produção de *softwares* mais acessíveis.

REFERÊNCIAS

BERSCH, R. **Introdução à Tecnologia Assistiva**. Centro Especializado em Desenvolvimento Infantil. Porto Alegre. 2017.

BRASIL. Decreto nº 6.949, de 25 de agosto de 2009. Promulga a Convenção Internacional sobre os Direitos das Pessoas com Deficiência e seu Protocolo Facultativo, assinados em Nova York, em 30 de março de 2007. **Diário Oficial da União**, Brasília, DF, 26 ago. 2009a. Seção 1, p. 3.

_____. Lei nº 10.098, de 19 de dezembro de 2000. Estabelece normas gerais e critérios básicos para a promoção da acessibilidade das pessoas portadoras de deficiência ou com mobilidade reduzida, e dá outras providências. **Diário Oficial da União**, Brasília, DF, 20 dez. 2000. Seção 1, p. 2.

_____. Lei nº 13.146, de 6 de julho de 2015. Institui a Lei Brasileira de Inclusão da Pessoa com Deficiência (Estatuto da Pessoa com Deficiência). **Diário Oficial da União**, Brasília, DF, 7 jul. 2015. Seção 1, p. 2.

_____. Lei nº 7.853, 24 de outubro de 1989. Dispõe sobre o apoio às pessoas portadoras de deficiência, sua integração social, sobre a Coordenadoria Nacional para Integração da Pessoa Portadora de Deficiência - Corde, institui a tutela jurisdicional de interesses coletivos ou difusos dessas pessoas, disciplina a atuação do Ministério Público, define crimes, e dá outras providências. **Diário Oficial da União**, Brasília, DF, 25 out. 1989. Seção 1, p. 1.

_____. Portaria SEDH nº 2.344, de 3 de novembro de 2010. O Ministro de Estado Chefe da Secretaria de Direitos Humanos da Presidência da República, no uso de suas atribuições legais, faz publicar a Resolução nº 01, de 15 de outubro de 2010, do Conselho Nacional dos Direitos da Pessoa Portadora de Deficiência - CONADE, que altera dispositivos da Resolução nº 35, de 6 de julho de 2005. **Diário Oficial da União**, Brasília, DF, 5 nov. 2010. Seção 1, p. 4.

_____. Subsecretaria Nacional de Promoção dos Direitos da Pessoa com Deficiência. Comitê de Ajudas Técnicas. **Tecnologia Assistiva**. Brasília: CORDE, 2009b. 138 p. Disponível em: <https://www.pessoacomdeficiencia.gov.br/app/sites/default/files/publicacoes/livro-tecnologia-assistiva.pdf>. Acesso em: 5 abr. 2019.

BURGSTALLER, S. **Designing software that is accessible to individuals with disabilities**: and making it more usable by everyone. 2008. Disponível em: <https://www.washington.edu/doit/designing-software-accessible-individuals-disabilities>. Acesso em: 31 mar. 2019.

CARRERAS MONTOTO, O. **Apps nativas de Android accesibles**. 2018. Disponível em: <https://olgacarreras.blogspot.com/2018/02/apps-nativas-de-android-accesibles.html>. Acesso em: 28 abr. 2019.

CASTRO JÚNIOR, G. **Variação linguística em Língua de Sinais Brasileira**: foco no léxico. 2011. 123 p. Dissertação (Mestrado em Linguística) – Curso de Pós-graduação em Linguística, Universidade Federal de Brasília, Brasília.

CENTER FOR UNIVERSAL DESIGN. **The Principles of Universal Design**. North Carolina, United States, versão 2.0, 1997. Disponível em: https://projects.ncsu.edu/design/cud/about_ud/udprinciplestext.htm. Acesso em: 28 abr. 2019.

eMAG. **Recomendações de Acessibilidade para a Construção e Adaptação de Conteúdos do Governo Brasileiro na Internet.** e-MAG, Acessibilidade de Governo Eletrônico. Modelo de Acessibilidade. Departamento de Governo Eletrônico, Secretaria de Logística e Tecnologia da Informação, Ministério do Planejamento, Orçamento e Gestão. Doc. de Ref., versão 3.1, 2014.

ESPAÑA. Real Decreto Legislativo 1/2013, de 29 de noviembre, por el que se aprueba el Texto Refundido de la Ley General de derechos de las personas con discapacidad y de su inclusión social. **Boletín Oficial del Estado.** Espanha, n. 289, p. 95635-95673, 3 dez. 2013.

FALBO, R.A. **Engenharia de Software** - Notas de Aula, UFES, 2014.

FIGUEIRA, A. M. S. **Análise das Técnicas de Levantamento de Requisitos para Desenvolvimento de Software nas Empresas de Vitória da Conquista – Ba.** 2012. 74 p. Monografia (Graduação em Ciência da Computação) – Departamento de Ciências Exatas, Universidade Estadual do Sudoeste da Bahia, Vitória da Conquista.

FORLIZZI, J.; BATTARBEE, K. **Understanding experience in interactive systems.** Human-Computer Interaction Institute, v. 46, p. 260-286, 2004.

GABRILLI, M (org.). **Guia sobre a Lei Brasileira de Inclusão.** 2016. Disponível em: <https://www.maragabrilli.com.br/wp-content/uploads/2016/03/Guia-sobre-a-LBI-digital.pdf>. Acesso em: 22 abr. 2019.

GALVÃO FILHO, T. A. Abordagem multidimensional: acessibilidade tecnológica. *In: DÍAZ, F. et al. (orgs.). Educação inclusiva, deficiência e contexto social: questões contemporâneas* [online]. Salvador: EDUFBA, p. 191-202, 2009.

GIL, A. C. **Métodos e técnicas de pesquisa social.** 6. ed. São Paulo: Atlas, 2008.

GNU OPERATING SYSTEM. **GNU Accessibility Statement.** 2019. Disponível em: <https://www.gnu.org/accessibility/accessibility.en.html>. Acesso em: 16 abr. 2019.

GUIMARÃES, A. P. N.; TAVARES, T. A. Avaliação de interfaces de usuário voltada à acessibilidade em dispositivos móveis: boas práticas para experiências de usuário. *In: IX Workshop de Teses e Dissertações (WTD) do Simpósio Brasileiro de Sistemas Multimídia e Web: João Pessoa, 2014.*

HASSENZAHN, M.; TRACTINSKY, N. User experience – a research agenda. **Behaviour & Information Technology.** London, UK: Taylor & Francis, v. 25, n. 2, p. 91-97, 2006.

HEITKÖTTER, H.; HANSCHKE, S.; MAJCHRZAK, T. A. Evaluating crossplatform development approaches for mobile applications. *In: CORDEIRO, J.; KREMPELS, K. H. (eds.). International Conference on Web information systems and technologies, 8., 2012, Porto, Portugal. Anais [...]. Porto, Portugal: Springer, 2013. p. 120-138.*

IBGE. **Censo Demográfico 2010.** 2010. Disponível em: <https://www.ibge.gov.br/estatisticas/sociais/populacao/9662-censo-demografico-2010.html?edicao=9749&t=sobre>. Acesso em: 23 abr. 2019.

IIDA, I.; SILVA, T. B. P.; SARMET, M. M. Conexão Emocional no Design. *In: MORAES, D; KRUKEN, L (org.). Cadernos de Estudos Avançados em Design – Transversalidade.* Belo Horizonte, v. 1, caderno 2, p. 33-52, jul. 2008.

- KAVCIC, A. Software Accessibility: Recommendations and Guidelines. **EUROCON 2005**: The International Conference on “Computer as a Tool”. Belgrade, Serbia and Montenegro: IEEE, v. 2, p. 1024-1027, nov. 2005.
- LAZER, L.; DUDLEY-SPONAUGLE, A.; GREENIDGE, K. D. Improving web accessibility: a study of webmaster perceptions. *In*: EMURIAN, H. H. (ed.). **Computers in human behavior**: the compass of human-computer interaction. 2. ed. Amsterdam: Elsevier, 2004, v. 20, p. 269-288.
- MARANHO, B. C. A. O direito fundamental à acessibilidade. **Revista Jurídica da Escola Superior do Ministério Público de São Paulo**, São Paulo, v. 6, p. 143-158, 2014.
- NARASIMHAN, N. (ed.). **Web Accessibility Policy Making**: An International Perspective. 3. ed. India: Hemkunt, 2012.
- NAVES, S. B. *et al.* (eds.). **Guia para produções audiovisuais acessíveis**. Brasília: [s.n.], 2016.
- PALMIERI, M.; SINGH, I.; CICCHETTI, A. Comparison of Cross-Platform Mobile Development Tools. *In*: INTERNATIONAL CONFERENCE ON INTELLIGENCE IN NEXT GENERATION NETWORKS, 16., 2012, Berlin. **Anais [...]**. Berlin, Germany: IEEE, 2012, p. 179-186.
- PREZOTTO, E.; BONIATI, B. Estudo de Frameworks Multiplataforma Para Desenvolvimento de Aplicações Mobile Híbridas. *In*: ENCONTRO ANUAL DE TECNOLOGIA DA INFORMAÇÃO E SEMANA ACADÊMICA DE TECNOLOGIA DA INFORMAÇÃO, 4., n. 1, 2014, Frederico Westphalen. **Anais [...]**. Frederico Westphalen: UFSM, 2014. p. 72-79.
- PRNEWSWIRE. Após levantar US\$ 3,3 milhões de financiamento, a Thunkable lança o “Thunkable X”, um criador de aplicativos móveis do tipo faça você mesmo para não programadores. **Exame**, 05 jun. 2018. Disponível em: <https://exame.abril.com.br/negocios/releases/apos-levantar-us-33-milhoes-de-financiamento-a-thunkable-lanca-o-thunkable-x-um-criador-de-aplicativos-moveis-do-tipo-faca-voce-mesmo-para-nao-programadores/>. Acesso em: 18 maio 2019.
- ROMAÑACH, J; LOBATO, M. **Diversidad funcional, nuevo término para la lucha por la dignidad en la diversidad del ser humano**. Foro de Vida Independiente. 2005. Disponível em: http://forovidaindependiente.org/wp-content/uploads/diversidad_funcional.pdf. Acesso em: 16 maio 2019.
- RUIZ, B. *et al.* **Guías multimedia accesibles**: el museo para todos. Madrid: ICONO, 2008.
- SILVA, T. M. S.; LIMA, D. F. Direito fundamental à acessibilidade no Brasil: uma revisão narrativa sobre o tema. *In*: ENCONTRO DE PRODUÇÃO ACADÊMICO-CIENTÍFICO, 5. **Anais [...]**. Campina Grande, 2013.
- THATCHER, J. *et al.* **Web Accesibility**: Web Standards and Regulatory Compliance. New Work: Friends of – Springer-Verlag. 2006.
- THUNKABLE. **Thunkable Documentation**. [s.d.]. Disponível em: <https://docs.thunkable.com/>. Acesso em: 15 maio 2019.
- VANDERHEIDEN, G. C (comp.). **Application Software Design Guidelines**: Increasing the Accessibility of Application Software to People with Disabilities and

Older Users. Wisconsin: Trace R & D Center, v. 1.1, jun. 1994. Disponível em: <https://park.org/Guests/Trace/pavilion/softwar1.htm>. Acesso em: 8 maio 2019.

WCAG 2.1. **Web Content Accessibility Guidelines 2.1**. 2018. Disponível em: <https://www.w3.org/TR/WCAG21/> Acesso em: 15 abr. 2019.

WINN, Z. **Democratizing app development**. MIT News Office. 2019. Disponível em: <http://news.mit.edu/2019/thunkable-app-inventor-0116>. Acesso em: 17 maio 2019.

ZUMO NOTÍCIAS. No final de 2010, smartphone vendeu mais que PC. **Terra**, 08 fev. 2011. Disponível em: <https://www.terra.com.br/noticias/tecnologia/negocios-e-ti/no-final-de-2010-smartphone-vendeu-mais-que-pc,bd489256184ea310VgnCLD200000bbcceb0aRCRD.html>. Acesso em: 16 abr. 2019.