UNIVERSIDADE FEDERAL DE SANTA CATARINA
DEPARTAMENTO DE INFORMÁTICA E ESTATÍSTICA

Gustavo Dirschnabel

DESENVOLVIMENTO DE UM *CHECKLIST* DE AVALIAÇÃO DE HEURÍSTICAS DE EXPERIÊNCIA DE USUÁRIO DE APLICATIVOS COM INTELIGÊNCIA ARTIFICIAL

Florianópolis

2023

Gustavo Dirschnabel

DESENVOLVIMENTO DE UM CHECKLIST DE AVALIAÇÃO DE HEURÍSTICAS DE EXPERIÊNCIA DE USUÁRIO DE APLICATIVOS COM INTELIGÊNCIA ARTIFICIAL

Trabalho de Conclusão de Curso submetido ao Curso de Bacharelado em Ciências da Computação para a obtenção do Grau de Bacharel em Ciências da Computação. Orientadora: Prof. Dr. rer. nat. Christiane Gresse von Wangenheim, PMP

Florianópolis

2023

RESUMO

Com o desenvolvimento e a popularização de ferramentas para a criação de sistemas com inteligência artificial, o uso desta tecnologia tornou-se mais comum no mercado, especialmente em grandes empresas. Estes sistemas estão sendo desenvolvidos também como aplicativos para dispositivos móveis, com uma aplicação comum sendo a classificação de imagens. Para estes aplicativos, experiência de usuário também é um grande determinador do sucesso. Apesar de já existir pesquisa sobre UX em aplicativos tradicionais, ainda não existe muita informação sobre como adaptar as heurísticas de UX para o contexto de aplicativos inteligentes. Assim, no presente projeto, com base nos princípios de AIX, é definido um conjunto de heurísticas para a avaliação de interfaces de aplicativos móveis com inteligência artificial para classificação de imagens, operacionalizada por meio do desenvolvimento de um *checklist*. As heurísticas e o checklist são validados por meio de um estudo de caso com interfaces de apps Android. Visa-se como resultado contribuir para a melhoria da experiência de usuário de aplicativos móveis voltados a classificação de imagens.

LISTA DE FIGURAS

LISTA DE TABELAS

LISTA DE ABREVIATURAS E SIGLAS

ML - *Machine Learning*
DL - *Deep Learning*
UX - *User Experience*
AIX - *AI Experience Design*
GTM - *Generative Topography Map*
UI - Interface de usuário
GUI - Interface gráfica de usuário
ACM - *Association for Computing Machinery*
IEEE - Instituto de Engenheiros Eletricistas e Eletrônicos
IHC - Interação humano computador
TMIC - *Teachable Machine Image Classifier*
N/A - Não se aplica
KMO - Índice de Kaiser-Meyer-Olkin
HTML - *HyperText Markup Language*
CSS - *Cascading Style Sheets*
JS - Javascript
PDF - *Portable Document Format*
RF - Requisito funcional
RNF - Requisito não funcional

SUMÁRIO

1. INTRODUÇÃO

1.1 CONTEXTUALIZAÇÃO

Recentemente, um número cada vez maior de empresas, incluindo gigantes como Google, Meta e Netflix, estão incorporando *Machine Learning* (ML) nos sistemas de software (DAI et al., 2020). *Machine Learning* é uma tecnologia orientada a dados, que, de forma autônoma e a partir da análise de exemplos, é capaz de extrair informações, padrões relevantes nos dados e aprender com eles (DEISENROTH et al., 2020). *Machine Learning* é adotado para diversas tarefas, inclusive a classificação de imagens. A classificação de imagens sempre foi uma das principais tarefas de ML (WANG et al., 2021), com técnicas como redes neurais profundas sendo fortemente adotadas nos últimos anos (ZANG et. al., 2020). Em tempos recentes, ML está sendo incorporado também amplamente em aplicativos móveis. Nestes, ML pode ser utilizada com processamento completamente em nuvem, para grandes conjuntos de dados e modelos complexos, ou com processamento no dispositivo (DAI et al., 2020).

Implementações de ML em dispositivos móveis enfrentam uma série de desafios, como poder computacional, bateria, pouca memória, tempo de resposta e riscos de privacidade (DAI et al., 2020). Entre esses uma das principais questões se refere a *User Experience* (Experiência de Usuário - UX) desses aplicativos inteligentes. Mesmo já existindo uma vasta área de pesquisa voltado a UX de aplicativos móveis tradicionais, os aplicativos com IA apresentam características diferentes. Entre estas, existe a questão de apresentação de resultados de forma probabilística, e a adaptação/melhoria ao longo do uso, que representa um modelo mental diferente ao qual os usuários estão acostumados. Essas características, se não tratadas com cuidado, podem levar à tomada de decisões erradas, frustração e abandono do software (GOOGLE, 2022a).

Esses desafios estão levando à criação da área de *AIX - AI Experience Design* (SUBRAMONYAM et al., 2021), propondo diretrizes para aplicativos com IA como, por exemplo, para desenvolvimento de interfaces de usuário. O principal motivador para aplicação de diretrizes em projetos de ML é

desenvolver software com melhor Experiência de Usuário, e que tenha boa percepção do usuário nos quesitos de oferecer mais controle, confiabilidade e uma sensação de produtividade (LI et al., 2022).

Nesse contexto já estão sendo propostos alguns conjuntos de heurísticas de design de interface de usuário, inclusive de grandes empresas de tecnologia, como da Microsoft (LI et al., 2022), Apple (APPLE, 2022) e Google (GOOGLE, 2022a). Essas propostas apresentam um conjunto de heurísticas de usabilidade, incluindo, como p.ex. "*Crie expectativas para adaptação"* e "*Planeje para a calibração da confiança do usuário durante a experiência"* (GOOGLE, 2022a).

Porém, observando essa variedade de conjuntos existentes, surge a pergunta: Até que ponto esses conjuntos são semelhantes ou abordam heurísticas distintas? Atualmente também ainda não existe um *checklist,* feito a partir dessas diretrizes, para auxiliar a realização de uma avaliação heurística de interfaces de aplicativos móveis. Além disso, essas heurísticas propostas são mais voltadas a sistemas de recomendação e assim se observa-se a falta de heurísticas mais específicas voltadas à classificação de imagens. Em geral, existe pouca pesquisa sobre o assunto no âmbito acadêmico (LI et al., 2022).

Observa-se também a necessidade de ensinar estes conceitos como parte de cursos de ML na Educação Básica para assegurar a qualidade de artefatos sendo criados pelos estudantes como resultado de aprendizagem. Atualmente já existem cursos que ensinam neste estágio escolar o desenvolvimento de aplicativos móveis com App Inventor (MIT, 2022), implantando modelos de *Deep Learning* (DL) criados com a Google Teachable Machine (GOOGLE, 2022d) e utilizando a extensão TMIC (*Teachable Machine Image Classifier* (OLIVEIRA, 2022)).

Assim, o objetivo do presente trabalho é analisar e mapear as diretrizes existentes, identificando heurísticas que se referem a aplicativos móveis de classificação de imagens, e desenvolver um *checklist* para avaliar o atingimento dessas heurísticas.

1.2 OBJETIVO

Objetivo geral

O objetivo geral do presente projeto é desenvolver um *checklist* de heurísticas para avaliação de experiência de usuário de apps com inteligência artificial. Com base nos princípios de AIX, é definido um conjunto de heurísticas para a avaliação de interfaces de aplicativos móveis com inteligência artificial para classificação de imagens, operacionalizada por meio do desenvolvimento de um *checklist*. Visa-se também a customização destas heurísticas ao contexto educacional voltado a criação de aplicativos no App Inventor implantando modelos de DL criados com GTM. A confiabilidade e validade das heurísticas e do checklist são analisadas por meio de um estudo de caso com interfaces de apps Android. Visa-se também o desenvolvimento de uma ferramenta web para suportar a realização de avaliações heurísticas utilizando o *checklist*.

Objetivos Específicos

Para que o objetivo geral seja alcançado, faz-se necessário que os seguintes objetivos específicos sejam alcançados:

O1. Sintetizar a teoria da área de AI principalmente *machine learning* para classificação de imagens e os princípios do AIX.
O2. Levantar o estado da arte em relação à princípios/heurísticas voltadas à avaliação do AIX.
O3. Mapear e unificar as *heurísticas* existentes e derivar o *checklist* a partir das heurísticas com foco em aplicativos Android inteligentes para classificação de imagens.
O4. Avaliar a confiabilidade e validade do *checklist* desenvolvido.
O5. Desenvolvimento de uma ferramenta de suporte online.

1.3 MÉTODO DE PESQUISA

Do ponto de vista da natureza da pesquisa a ser empreendida, este trabalho se classifica como uma pesquisa aplicada (ou tecnológica), que tem

por objetivo gerar produtos e/ou processos inéditos, com finalidades imediatas, com base em conhecimentos prévios. Quanto aos objetivos da pesquisa, este trabalho se caracteriza como uma pesquisa exploratória, pois visa proporcionar maior familiaridade com o problema investigado a fim de torná-lo explícito. A metodologia de pesquisa deste projeto é dividida nas seguintes etapas:

Etapa 1. Fundamentação teórica: É realizada uma análise de literatura na área de IA e AIX.

Atividade 1.1: Analisar a área de IA, especialmente em relação ao ML/*deep learning* para classificação de imagens;

Atividade 1.2: Analisar os princípios de AIX para apps inteligentes.

Etapa 2. Levantamento do estado da arte: Nesta etapa é realizado um mapeamento sistemático de literatura seguindo o procedimento do Petersen et al. (2008) para identificar e estudar as diretrizes já propostas. No início do processo de mapeamento é feita a definição do escopo da pesquisa, e são definidos perguntas de pesquisa, palavras-chaves e critérios de inclusão e exclusão. O *search string* é calibrado por meio de buscas iniciais. Em seguida é realizada a busca e seleção de artigos relevantes. Destes artigos relevantes são extraídas informações que respondem à pergunta de pesquisa definida no protocolo da revisão. O mapeamento sistemático é finalizado com a análise das informações dos documentos restantes da etapa anterior. Desta forma, a subdivisão desta atividade se resume em:

Atividade 2.1: Definir o protocolo da revisão sistemática;

Atividade 2.2: Executar a busca;

Atividade 2.3: Analisar e interpretar as informações extraídas.

Etapa 3. Desenvolvimento de um conjunto de heurísticas, seguindo o procedimento proposto por Rusu et al. (2011). O primeiro passo

exploratório conforme essa metodologia já é realizada na etapa 2: Coletar bibliografia relacionada com os tópicos principais da pesquisa. Dessa forma enfoca-se nessa etapa na fase descritiva, para realçar e mapear as características mais importantes da informação previamente coletada. No próximo passo, no Estágio correlacional, são identificadas as características que as heurísticas de usabilidade para aplicações específicas devem ter, analisando a usabilidade de aplicativos com classificação de imagens. Por último, no estágio explicativo é especificado formalmente o conjunto de heurísticas proposto e um *checklist*, já testando as versões preliminares com aplicativos inteligentes.

Atividade 3.1: Mapear as características mais importantes;

Atividade 3.2: Identificar as características específicas das heurísticas;

Atividade 3.3: Especificar formalmente o conjunto de heurísticas e *checklist*.

Etapa 4. Avaliação do *checklist*: Seguindo Rusu et al. (2011) o conjunto de heurísticas e *checklist* desenvolvido é avaliado em termos de confiabilidade e validade meio de estudos de caso: (1) a sua avaliação de um app paralela por especialistas de design de interface e (2) por uma aplicação do *checklist* a um conjunto de apps para avaliar estatisticamente a confiabilidade e validade por meio de um estudo de caso em que apps Android do Google Play são avaliados utilizando o checklist.

Atividade 4.1: Avaliação do *checklist* por meio de especialistas

A4.1.1 Definir da avaliação do *checklist*

A4.1.2 Executar a avaliação e coleta de dados

A4.1.3 Analisar os dados obtidos

Atividade 4.2: Baseando-se no *feedback*, o conjunto de heurísticas e checklist é melhorado.

Atividade 4.3: Avaliação de confiabilidade e validade por meio de um estudo de caso

A4.3.1 Definir da avaliação do *checklist*

A4.3.2 Executar a avaliação e coleta de dados

A4.3.3 Analisar os dados obtidos

Etapa 5. Desenvolvimento de ferramenta de suporte: Para facilitar o uso do *checklist* é desenvolvido uma ferramenta web para possibilitar a realização de uma avaliação heurística e a apresentação dos resultados. Para o desenvolvimento se segue um processo iterativo incremental seguindo Larman (2004), composto das seguintes atividades:

Atividade 5.1: Análise de requisitos.

Atividade 5.2: Modelagem, implementação e testes.

1.4 ESTRUTURA DO DOCUMENTO

No capítulo 2 é apresentado uma fundamentação acerca dos conceitos essenciais para o desenvolvimento do trabalho: sistemas de IA para classificação de imagens, usabilidade e *User Experience*. No capítulo 3 é apresentado o resultado de uma revisão sistemática em relação aos conjuntos de heurísticas e *checklists* existentes para aplicativos inteligentes. No capítulo 4 é apresentado todo o processo de desenvolvimento de uma solução, mapeando os princípios de AIX obtidos anteriormente e especificando um *checklist* e conjunto de heurísticas que então validadas por especialistas, refinadas, avaliadas estatisticamente e por fim implementadas em um sistema de apoio para avaliação heurística. Os resultados do trabalho e conclusões são então discutidos no capítulo 5.

2. FUNDAMENTAÇÃO TEÓRICA

Neste capítulo são apresentados os conceitos básicos necessários ao entendimento do trabalho, como classificação de imagens e seu uso em aplicativos inteligentes, usabilidade, *user experience* e AIX.

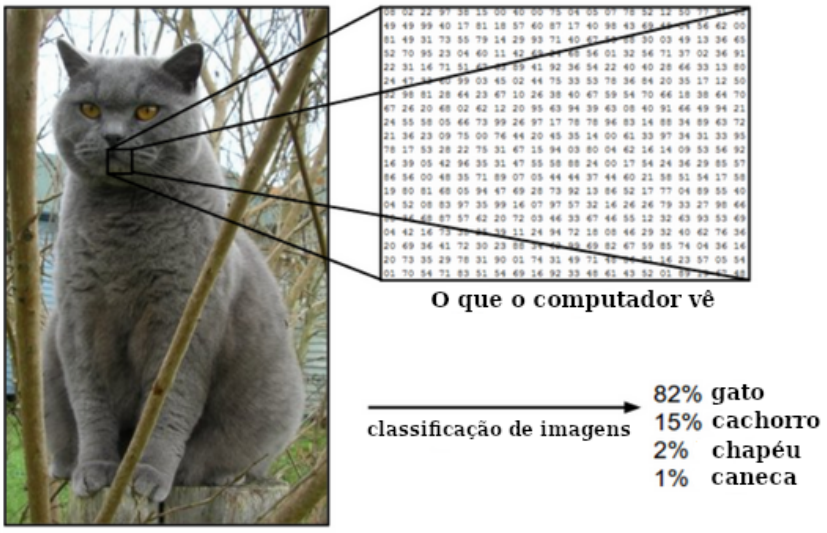
2.1 APLICATIVOS INTELIGENTES PARA CLASSIFICAÇÃO DE IMAGENS

2.1.1 Classificação de imagens

Atualmente, aplicativos para a detecção, reconhecimento e classificação de objetos vêm sendo desenvolvidos em algumas áreas. Na agricultura, existem aplicativos para o reconhecimento de doenças em plantas (VERMA, 2019); na eletroquímica há um aplicativo para a classificação do nível de contaminação bacteriana de uma amostra de água (GUNDA, 2019) e na área médica, especialmente a dermatologia, existe um considerável número de aplicativos para a detecção de doenças, inclusive de câncer de pele (DAI, 2019). Estes aplicativos se apoiam sobre a tecnologia de classificação de imagens.

Classificação de imagens pode ser definida como a tarefa de categorizar imagens em uma de várias classes pré-definidas (RAWAT, 2017). Em sistemas computacionais, imagens são representadas como um grande conjunto de *pixels*, que são por si conjuntos de números que representam uma configuração de cores (KARPATHY, 2016). O desafio da classificação de imagens é transformar esses números em uma rotulo, como “gato” (KARPATHY, 2016), conforme é apresentado na Figura 1.

Figura 1 - Classificação de uma imagem em “gato” com 82% de confiança



Fonte: KARPATHY (2016) (adaptado)

O processo de classificação de imagens envolve a definição de uma quantidade N de classes, como “gato”, “cachorro”, “chapéu” e “caneca” (Figura 1), e o treinamento de um modelo que possa inferir sobre os graus de pertinência de qualquer imagem a estas classes (GOOGLE, 2022b). Estes modelos devem ser capazes de inferir corretamente mesmo quando expostos a imagens que apresentam o objeto da classificação em diversas condições (KARPATHY, 2016):

- Variação de ponto de vista ao objeto;
- Diferentes níveis de iluminação;
- Oclusão parcial do objeto;
- Variação de escala do objeto;
- Deformação do objeto;
- Objeto confundido com o ambiente e fundos diferentes;
- A classe do objeto apresenta uma variação interna inerente.

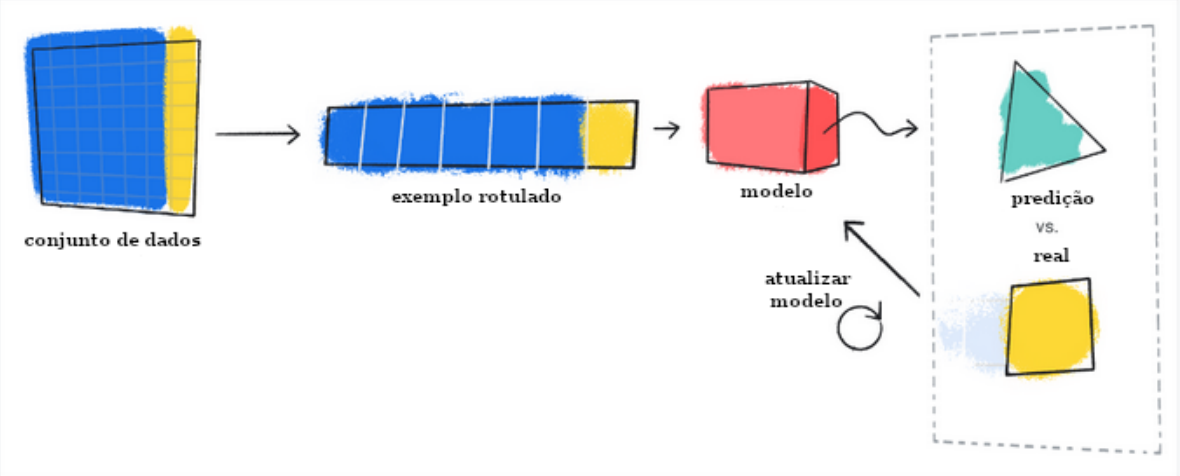
A criação de um algoritmo para o modelo de classificação que consiga operar com exatidão mesmo nas condições apresentadas anteriormente não é uma tarefa trivial. Por consequência, emprega-se uma técnica de inteligência artificial orientada a dados onde conjuntos de dados são utilizados para treinar um modelo de inferência baseado em ML (KARPATHY, 2016). A técnica de ML

mais utilizada atualmente para a classificação de imagens é o *deep learning* (MARTÍNEZ-FERNÁNDEZ, 2021).

A aplicação de *deep learning* usando redes neurais artificiais para a criação do modelo de classificação pode ser feita a partir de algoritmos de aprendizagem supervisionado que analisam as características de imagens pré-rotuladas para derivar um relacionamento matemático que procura explicar porque essas imagens pertencem às classes que lhes foram atribuídas, processo conhecido como treinamento do modelo (GOOGLE, 2022b).

A capacidade representativa da inferência realizada sobre o modelo treinado depende da qualidade do conjunto de dados utilizado para o treinamento, que é mensurada tipicamente pela acurácia, que consiste na fração de vezes em que a inferência acertou. Outras medidas existem em contextos mais específicos. *Precision* é definido como a proporção de classificações em um rótulo que estavam corretas, e é calculado como uma razão entre o número de classificações corretas de um rótulo sobre a quantidade total de classificações neste mesmo rótulo (GOOGLE, 2022b). *Recall* é definido como a proporção de vezes que objetos da mesma classe foram rotulados a ela, calculado como a razão entre o número de classificações corretas desta classe e o número total de objetos nessa classe (GOOGLE, 2022b). Para melhorar o desempenho do modelo em relação a essas métricas, o processo de treinamento pode ser realizado iterativamente ao longo do desenvolvimento com novos conjuntos de dados, assim como mostra a Figura 2 (GOOGLE, 2022b)

Figura 2 - Retreinamento do modelo com um novo conjunto de dados



Fonte: GOOGLE (2022b) (adaptado)

2.1.2 Sistemas com inteligência artificial

O uso de tecnologias de inteligência artificial em aplicativos levanta a questão de como isso pode afetar o usuário final do aplicativo. Para responder isso, uma fundamentação sobre IA em sistemas é necessária. Sistemas com inteligência artificial são sistemas que apresentam comportamento inteligente, que se manifesta a partir da análise do ambiente que estão inseridos, o aprendizado com ele e a tomada de medidas, e que é utilizado para atingir objetivos específicos com um determinado nível de autonomía (MARTÍNEZ-FERNÁNDEZ, 2021).

Por conta dessas características, sistemas com IA podem apresentar comportamentos imprevisíveis que podem confundir, atrapalhar, ofender e até mesmo pôr o usuário em perigo (AMERSHI et al., 2019). A capacidade do sistema de explicar por que tomou determinada decisão é essencial para a criação de confiança no sistema pelo usuário, especialmente para algumas áreas de aplicação, como a médica, financeira e legal (GUNNING, 2019). Pensando nisso, princípios para sistemas de IA confiáveis estão sendo levantados por órgãos como a União Européia (EUROPEAN COMMISSION, 2019) e organizações como a Google (2022c):

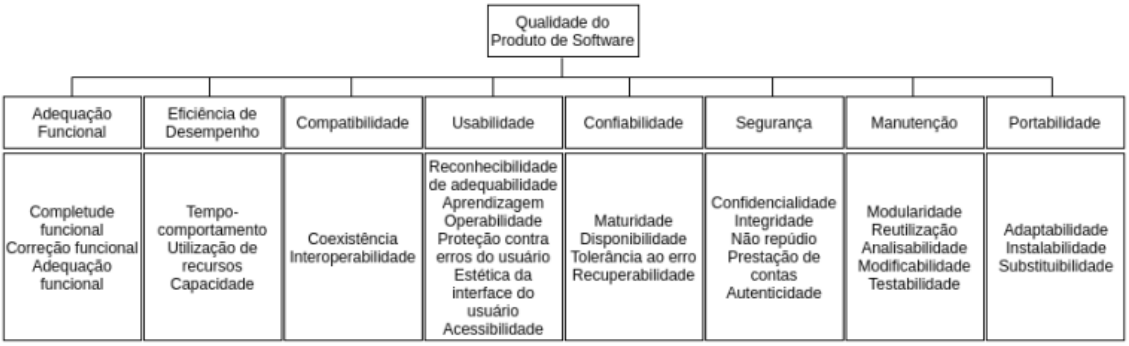
- **Trazer um benefício para a sociedade.** Sistemas de IA devem beneficiar todos os seres humanos, incluindo as gerações futuras.
- **Robustez e segurança técnica.** Ser construído e testado quanto à segurança para evitar resultados não intencionais que criam riscos de danos especialmente a pessoas.
- **Responsabilidade.** Devem ser implementados mecanismos para garantir responsabilidade e prestação de contas pelos sistemas de IA e seus resultados.
- **Diversidade, não discriminação e justiça.** Evitar criar ou reforçar preconceitos injustos que afetem as pessoas, principalmente aqueles relacionados a características sensíveis como raça, etnia, gênero, nacionalidade, renda, orientação sexual, habilidade e crença política ou religiosa.
- **Privacidade de dados.** Assegurar a privacidade de dados de usuários.
- **Transparência.** Os dados e os sistemas de IA devem ser transparentes e suas decisões devem ser explicáveis. Os humanos precisam estar cientes de que estão interagindo com um sistema de IA e devem ser informados sobre as capacidades e limitações do sistema.
- **Agência humana e supervisão.** Sistemas de IA devem capacitar os seres humanos, permitindo que tomem decisões informadas. Ao mesmo tempo, devem permitir controle e supervisão por humanos.

Durante a interação com qualquer sistema de software, o usuário se guia com o seu modelo mental, uma representação interna criada por ele de como o sistema funciona baseado em experiências anteriores (STAGGERS, 1993). O modelo mental do usuário deve ser adaptado em sistemas inteligentes devido às características de probabilidade e incerteza desses. É importante que o usuário seja informado sobre as capacidades do sistema de IA para que não construa um modelo mental que superestime o que o sistema pode fazer (GOOGLE, 2022a). O nível certo de explicação é fundamental para ajudar os usuários a entender como o sistema funciona. Uma vez que os usuários tenham modelos mentais claros dos recursos e limites do sistema, eles podem entender como e quando confiar nele para ajudar a atingir seus objetivos.

2.2 USABILIDADE E *USER EXPERIENCE*

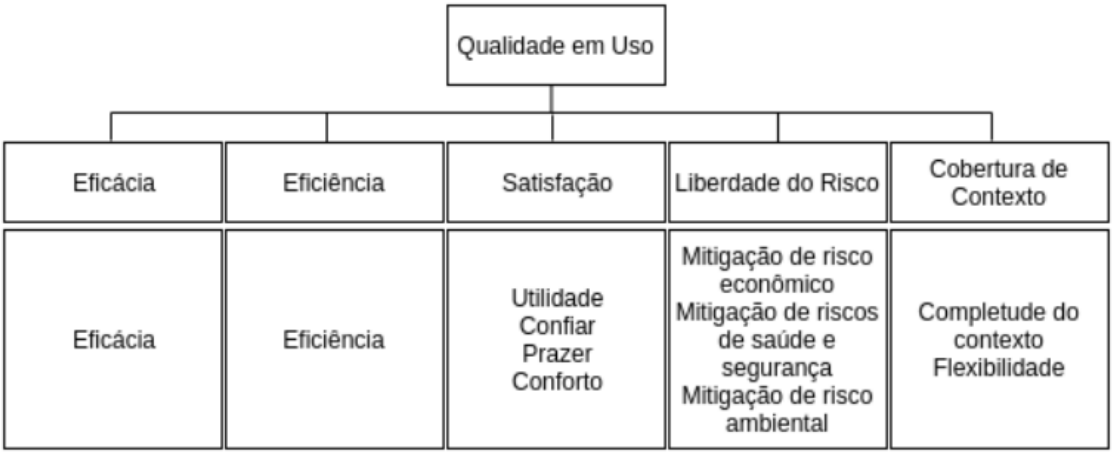
Interface de usuário (UI) é um componente de software que permite a interação, entendimento e controle de um software. A função principal de uma UI é capturar dados de entrada e apresentar os resultados do programa na tela (GALITZ, 2002). *Design* de interface, nesse contexto, é a construção de UIs que satisfaçam as necessidades, limitações e capacidades do usuário fornecendo uma boa usabilidade (JURISTO et al., 2007). Usabilidade, por sua vez, é definida como a capacidade de um usuário de completar seus objetivos com o software com eficácia, eficiência e satisfação (ISO/IEC, 2010). As normas SQuaRE (Requisitos e avaliação da qualidade de produto de software, série ISO 25050 a ISO 25099) posicionam a usabilidade como um entre os componentes que contribuem para a qualidade de um produto de software. Apesar disso, os fatores de usabilidade (eficácia, eficiência e satisfação) representam boa parte das características da qualidade de uso de software, no ponto de vista do usuário final (ISO/IEC, 2010; ISO/IEC, 2011) (Figura 4).

Figura 3 - Componentes de qualidade de um produto de software



Fonte: ISO20510, 2011. Adaptado por Silva, 2019

Figura 4 - Características da Qualidade em Uso



Fonte: ISO20510, 2011. Adaptado por Silva, 2019

Complementando o conceito de usabilidade, a Experiência de Usuário (UX), é definida como uma combinação de percepções e respostas do usuário, como emoções, crenças, preferências, conforto, comportamentos e realizações que advém de períodos anteriores, durante e após o uso do software (ISO/IEC, 2020).

Em aplicativos móveis, a busca pela usabilidade tornou a interface de usuário em uma das principais considerações no desenvolvimento, especialmente com a grande heterogeneidade de plataformas e requisitos de usuário atrelados a elas (RUIZ, 2021).

Interfaces de usuários para aplicativos móveis podem ser projetadas de uma forma similar a aplicações *web*, mas devem ser repensadas ao redor das características específicas desse domínio (WASSERMAN, 2010). Os tamanhos de tela menor e os diferentes estilos de interação dos usuários, estes últimos possibilitados por componentes de interface baseados em gestos, toques, movimentação do dispositivo e teclados virtuais, têm um grande impacto no projeto de interação do usuário com o aplicativo. As plataformas móveis incluem seus próprios *kits* de desenvolvimento de UI e diretrizes para utilizar os componentes de interface de modo a considerar essas características e padrões estabelecidos na plataforma (WASSERMAN, 2010).

Em aplicações que utilizam *ML* especificamente, o design de interface deve considerar alguns aspectos: a imprecisão do usuário, a incerteza entre as entradas e intenções dele, a evolução contínua do sistema e o

não-determinismo inerente à técnica, que leva à resultados probabilísticos (gerência de saída) (DUDLEY, 2018).

2.2.1 Avaliação heurística

Para a criação e avaliação da usabilidade de interfaces gráficas pode ser utilizado a avaliação heurística, que consiste em inspeções da interface considerando um conjunto de princípios de usabilidade, conhecidos como heurísticas (NIELSEN, 1994). Diversos autores propuseram conjuntos de heurísticas para diversos fins e em diferentes níveis de abstração (p.ex. MANDEL, 1997; NIELSEN, 1994; NORMAN, 1983; SHNEIDERMAN e PLAISANT, 2005). O conjunto de heurísticas de usabilidade mais comumente utilizadas para o desenvolvimento de aplicações *desktop* foram desenvolvidas por Nielsen (1994) e são apresentadas na Tabela 1.

Tabela 1 - As 10 Heurísticas de Nielsen

Nº	Heurística	Descrição
1	Visibilidade do status do sistema	O sistema deve sempre manter o usuário informado sobre o que está acontecendo. Utilizando *feedback* apropriado e em tempo razoável
2	Compatibilidade entre sistema e mundo real	O sistema deve utilizar uma linguagem familiar ao usuário e seguir convenções do mundo real
3	Controle e liberdade ao usuário	Deve existir uma maneira clara de escapar estados indesejados do sistema, de desfazer e refazer alterações
4	Consistência e padrões	Não deve haver ambiguidade nas palavras utilizadas na interface. Siga os padrões da plataforma.
5	Prevenção de erros	Projete para impedir erros de uso, remova condições que resultem em erros ou alerte os usuários sobre elas.
6	Reconhecimento em vez de lembrança	Minimize a carga mental do usuário deixando ações, opções, objetos e informações visíveis ou facilmente recuperáveis
7	Flexibilidade e eficiência de uso	Forneça mecanismos que acelerem o uso das funcionalidades mais utilizadas
8	Projeto minimalista e estético	Não apresente mais informação que o necessário para a tarefa atual
9	Auxiliar o usuário a reconhecer, diagnosticar e recuperar de erros	Mensagens de erro devem estar na linguagem do usuário, serem precisas e sugerirem soluções
10	Ajuda e documentação	Toda documentação deve ser concisa, de fácil acesso e concentrada nas tarefas do usuário

Fonte: adaptado de NIELSEN (2005)

Conjuntos de heurísticas como esse podem ser utilizados em contextos específicos desde que adaptados para as questões de usabilidade deste domínio (HERMAWATI e LAWSON, 2015). Existem exemplos de heurísticas adaptadas em diversos domínios como a educação on-line (DRINGUS e COHEN, 2005), interação humano-robô (TSUI et al., 2009), dispositivos médicos (KATRE et al., 2010), entre outros.

Na avaliação heurística, um grupo de avaliadores inspecionam os elementos da interface gráfica do software, buscando identificar pertinências e contradições aos princípios de um conjunto de heurísticas escolhido. Ao final do processo, cada avaliador apresenta uma lista de problemas de usabilidade encontrados, em que cada item possui referências a todas as heurísticas violadas. A agregação de todos os problemas levantados pode então ser levada para discussão com os desenvolvedores, com a finalidade de estipular mudanças que possam tratá-los. A avaliação heurística pode ser realizada em qualquer parte do ciclo de vida do software (NIELSEN, 1990).

Nielsen (1990), recomenda que cada avaliador execute suas inspeções de forma individual, apenas compartilhando seus resultados ao final do processo. Além disso, o autor sugere que sejam feitas pelo menos duas análises, a primeira de modo geral para entender o fluxo de interação com o software, e a segunda para dar um foco a elementos específicos da interface.

O número de avaliadores envolvidos no processo é um fator determinante de sua eficácia. Avaliadores individuais conseguem identificar apenas 20%-51% dos problemas de usabilidade (NIELSEN, 1990). O número ideal de avaliadores para o melhor custo benefício é de 3-5, nessas condições, em média 75% dos problemas de usabilidade são identificados (NIELSEN, 1990).

3 ESTADO DA ARTE

Este capítulo apresenta um mapeamento sistemático da literatura com o objetivo de levantar o estado da arte sobre heurísticas de avaliação de interface de usuário de aplicativos inteligentes. O mapeamento segue a metodologia proposta por Petersen, Vakkalanka e Kuzniarz (2015). O mapeamento responde à pergunta de pesquisa: Quais conjunto de heurísticas existem para a avaliação do design de interface de aplicativos inteligentes para *smartphones* Android. O foco principal do mapeamento está na avaliação de aplicativos que possuem funcionalidades de classificação de imagens. No entanto, são consideradas abordagens de avaliação de aplicativos inteligentes em geral, desde que contenham princípios que possam ser aplicados para classificação de imagens.

3.1 DEFINIÇÃO DO PROTOCOLO DE MAPEAMENTO

A pergunta de pesquisa é decomposta nas seguintes perguntas de análise:

PA1. Quais conjuntos de heurísticas de avaliação do design de interface existem e quais suas características?

PA2. Quais são as heurísticas destes modelos?

PA3. Qual o suporte existente para avaliação utilizando os conjuntos de heurísticas encontrados (como *checklist* ou ferramenta). Até que ponto essa avaliação foi automatizada?

PA4. Como o conjunto de heurísticas foi desenvolvido e avaliado?

As buscas foram realizadas nas principais bases de dados e bibliotecas digitais da área da computação: ACM Digital Library, IEEE Xplore Digital Library, Wiley e Scopus. Além dessas bases de dados, com o intuito de abranger uma maior gama de publicações, foram realizadas buscas no Google Scholar, que indexa um grande conjunto de dados de diversas fontes de produção científica (HADDAWAY et al., 2015). Buscas informais foram utilizadas para calibrar a string de busca, a qual foi definida com base nos termos relevantes da pergunta de pesquisa do mapeamento. Durante essas

buscas fora do protocolo do mapeamento, notou-se a existência de poucos artigos relevantes para o foco específico da pesquisa, e foram encontradas diretrizes propostas por corporações sem a publicação de um artigo correspondente. Essas diretrizes foram encontradas anexadas sob a página ML+DESIGN(2022), e este último foi incluído como base.

Portanto, decidiu-se ampliar o escopo das buscas, resultando nos termos de busca, sinônimos e termos similares apresentados na Tabela 2.

Tabela 2 - Termos de busca e sinônimos

Termo	**Sinônimos**
AIX	Human-AI, XAI, IML, “AI Experience”, “human in the loop”, “human centered”
Artificial Intelligence	AI, Deep learning, DL, ML, Machine Learning
Evaluation	assessment, identification
Heuristic	guidelines, principles, recommendations
“user interface”	GUI, UI, UX, “user experience”, usability

Fonte: elaborado pelo autor

Assim, definiu-se a seguinte *string* de busca genérica:

(heuristic OR guidelines OR principle* OR recommendation*) AND (GUI OR UI OR “user experience” OR UX OR “user interface” OR usability) AND (evaluat* OR assess* OR identif*) AND (“artificial intelligence” OR AI OR “machine learning” OR ML OR “deep learning” OR DL) AND (“human-ai” OR AIX OR XAI OR IML OR “AI Experience” OR “human in the loop” OR “Human-centered”).

A adaptação desta *string* de busca para a sintaxe de cada base pode ser vista na Tabela 3.

Tabela 3 - *Strings* de busca por base

Local de Busca	**String de Busca**
ACM	[[Abstract: heuristic] OR [Abstract: guidelines] OR [Abstract: principle*] OR [Abstract: recommendation*]] AND [[Abstract: "gui"] OR [Abstract: "ui"] OR [Abstract: "user experience"] OR [Abstract: ux] OR [Abstract: "user interface"] OR [Abstract: usability]] AND [[Abstract: evaluat*] OR [Abstract: assess*] OR [Abstract: identif*]] AND [[Abstract: "artificial intelligence"] OR [Abstract: "ai"] OR [Abstract: "machine learning"] OR [Abstract: "ml"] OR [Abstract: "deep learning"] OR [Abstract: "dl"]] AND [[Abstract: "human?ai"] OR [Abstract: aix] OR [Abstract: xai] OR [Abstract: iml] OR [Abstract: "ai experience"] OR [Abstract: "human?in?the?loop"] OR [Abstract: "human?centered"]] AND [Publication Date: (01/01/2017 TO *)]

IEEE	("All Metadata":heuristic OR "All Metadata":guidelines OR "All Metadata":principle* OR "All Metadata":recommendation*) AND ("All Metadata":GUI OR "All Metadata":UI OR "All Metadata":"user experience" OR "All Metadata":UX OR "All Metadata":"user interface" OR "All Metadata":usability) AND ("All Metadata":evaluat* OR "All Metadata":assess* OR "All Metadata":identif*) AND ("All Metadata":"artificial intelligence" OR "All Metadata":AI OR "All Metadata":"machine learning" OR "All Metadata":ML OR "All Metadata":"deep learning" OR "All Metadata":DL) AND ("All Metadata":"Human?AI" OR "All Metadata":AIX OR "All Metadata":XAI OR "All Metadata":IML OR "All Metadata":"AI Experience" OR "All Metadata":"Human?in?the?loop" OR "All Metadata":"Human?centered")
Google Scholar	guidelines “user interface” evaluation AI “AI Experience”
ML+Design	--
Scopus	TITLE-ABS-KEY (heuristic OR guidelines OR principle* OR recommendation*) AND TITLE-ABS-KEY (gui OR ui OR "user experience" OR ux OR "user interface" OR usability) AND TITLE-ABS-KEY (evaluat* OR assess* OR identif*) AND TITLE-ABS-KEY ("artificial intelligence" OR ai OR "machine learning" OR ml OR "deep learning" OR dl) AND TITLE-ABS-KEY (human?ai OR aix OR xai OR iml OR ai AND experience OR human?in?the?loop OR human?centered) AND PUBYEAR > 2017 AND SUBJAREA (comp)
Wiley	"heuristic OR guidelines OR principle* OR recommendation*" in Abstract and ""GUI" OR "UI" OR "user experience" OR UX OR "user interface" OR usability" in Abstract and "evaluat* OR assess* OR identif*" in Abstract and ""Artificial Intelligence" OR AI OR "Machine Learning" OR ML OR "Deep Learning" OR DL" in Abstract and "Human?AI OR AIX OR XAI OR IML OR "AI Experience" OR Human?in?the?loop OR Human?centered" in Abstract

Fonte: elaborado pelo autor

Os artigos analisados foram incluídos ou excluídos conforme os seguintes critérios:

- Apresenta um conjunto de heurísticas, *checklists* ou rubricas que avaliam a qualidade do design de interface de aplicativos móveis inteligentes. São consideradas também pesquisas voltadas de forma genérica a design de interface de qualquer tipo de sistema de software inteligente, porém como o foco deste trabalho é para sistemas de classificação de imagens, são excluídos conjuntos de heurísticas voltadas especificamente a outras tarefas de ML como *chatbots* ou sistemas de interação pela fala;
- Não é voltado à identificação de atividades ou processos no desenvolvimento de sistemas inteligentes, são apenas considerados trabalhos que contribuem para AIX;
- É específico à área da computação, ou seja não criar conjuntos de heurísticas aplicáveis somente em alguns domínios, como o médico;
- O idioma utilizado na escrita do artigo é inglês ou português;
- Pode ser acessado a partir do uso do Portal CAPES;
- Foi publicado nos últimos 5 anos no período de janeiro/2017 a dezembro/2022.

Foram incluídos apenas artigos que contenham heurísticas explícitas para avaliação do design visual, ou alternativamente artigos que reforcem a validade de um conjunto de heurísticas anterior.

3.2 EXECUÇÃO DA BUSCA

As buscas foram realizadas em outubro de 2022 em três etapas. Na primeira etapa foi aplicada a string de busca nas bases de dados. Na segunda etapa, foram aplicados os critérios de inclusão e exclusão sob os resumos dos resultados mais relevantes de cada base (limitando-se a 200 obras em cada), resultando em uma lista de artigos potencialmente relevantes. Na terceira etapa, todo o texto dos artigos potencialmente relevantes foi analisado, aplicando novamente os critérios de inclusão/exclusão e de qualidade. Como resultado foram identificadas 7 publicações relevantes (Tabela 4).

Tabela 4 - Resultado da busca

Base	**Resultados da busca**	**Artefatos Analisados**	**Quantidade de publicações potencialmente relevantes**	**Publicações relevantes**
ACM	39	39	5	4
IEEE	10	10	0	0
Google Scholar	77	77	1	0
ML+Design	8	8	4	3
Scopus	63	63	2	0
Wiley	4	4	0	0
Total (sem duplicatas)				7

Fonte: elaborado pelo autor

3.3 ANÁLISE DOS RESULTADOS

A partir dos artigos selecionados é feita uma análise em relação às perguntas anteriormente levantadas.

3.3.1 Quais conjuntos de heurísticas de avaliação do design de interface existem e quais suas características?

De forma geral foram encontrados poucos artigos científicos no contexto acadêmico, completados por trabalhos apresentados no contexto de organizações/empresas de TI. Dentre as publicações encontradas, Amershi et al. (2019), Li et al. (2022) e Microsoft (2019) se referem ao mesmo conjunto de heurísticas, e portanto foram agrupados nesse ponto em diante. Por conta disso, apenas 5 das 7 publicações indicam conjuntos diferentes de heurísticas.

Na Tabela 5, os conjuntos de heurísticas que foram encontrados são apresentados. Todos os trabalhos encontrados propõem heurísticas para uso geral. Referente às plataformas, incluem conjuntos de heurísticas tanto para sistemas web quanto aplicativos móveis. Analisando o foco das heurísticas em relação às tarefas de IA, observa-se que nenhum dos trabalhos apresenta um foco explícito sobre uma tarefa de IA, e portanto assume-se que suas heurísticas podem ser aplicadas a qualquer uma. Porém, implicitamente se observa que a maioria das heurísticas foram projetadas para sistemas de recomendação. Em termos de domínios de aplicação, Mohseni et al. (2021) são os únicos a proporem heurísticas voltadas a um domínio específico de AIX: a explicabilidade do sistema de IA ao usuário.

Tabela 5 - Conjuntos de heurísticas encontrados

Citação	**Plataforma**	**Tarefas de IA suportadas**	**Ambiente em que foram idealizadas**	**Domínio de aplicação**
(AMERSHI et al., 2019) (LI et al., 2022) (MICROSOFT, 2019)	Genérico	Genérico	Acadêmico	—
(APPLE, 2022)	Genérico	Genérico	Industrial	—
(DUDLEY, 2018)	Genérico	Genérico	Acadêmico	—
(GOOGLE, 2022a)	Genérico	Genérico	Industrial	—
(MOHSENI et al., 2021)	Genérico	Genérico	Acadêmico	Explicabilidade

Fonte: elaborado pelo autor

3.3.2. Quais são as heurísticas destes modelos?

As heurísticas formam uma base para a avaliação de AIX. Os trabalhos encontrados expressam suas heurísticas em forma de diretrizes, com a exceção do conjunto da Apple (2022), que as apresenta como conjuntos de um total de 60 princípios, agrupados por área de AIX.

Mohseni et al. (2021) propõem diretrizes para todo o processo de desenvolvimento de um sistema inteligente, porém levando em consideração o foco do presente mapeamento, só foram consideradas as diretrizes que se referem à interface de usuário.

As heurísticas propostas em cada trabalho são apresentadas na Tabela 6.

Tabela 6 - Conjuntos de heurísticas de AIX encontrados

Citação	**Heurísticas**
(AMERSHI et al., 2019) (LI et al., 2022) (MICROSOFT, 2019)	G1. Deixe claro o que o sistema pode fazer
	G2. Deixe claro o quão bem ele pode fazer
	G3. Baseie-se em contexto para agendar serviços
	G4. Mostre informações contextualmente importantes
	G5. Adeque-se às normas sociais relevantes
	G6. Mitigue os vieses sociais
	G7. Dê suporte à invocação eficiente
	G8. Dê suporte à dispensa eficiente
	G9. Dê suporte à correção eficiente
	G10. Limite o escopo dos serviços quando em dúvida
	G11. Deixe claro por que o sistema agiu dessa maneira
	G12. Lembre de interações recentes
	G13. Aprenda com o comportamento do usuário
	G14. Atualize e adapte com cuidado
	G15. Estimule o *feedback* granular
	G16. Comunique ao usuário as consequências de suas ações
	G17. Providencie controles globais
	G18. Notifique usuários sobre mudanças
(APPLE, 2022)	G1. Princípios sobre *feedback* explícito
	G2. Princípios sobre *feedback implícito*
	G3. Princípios sobre calibração
	G4. Princípios sobre correções
	G5. Princípios sobre erros
	G6. Princípios sobre múltiplas opções
	G7. Princípios sobre confiança
	G8. Princípios sobre atribuição
	G9. Princípios sobre limitações
(DUDLEY, 2018)	G1. Deixe objetivos e restrições da tarefa explícitos
	G2. Dê suporte ao entendimento do usuário sobre a incerteza e confiança do modelo
	G3. Capture a intenção do usuário e não a entrada
	G4. Providencie representações efetivas de dados
	G5. Tome vantagem da interatividade e promova interações ricas
	G6. Engaje o usuário
(GOOGLE, 2022)	G1. Determine se IA traz valor
	G2. Defina as expectativas certas
	G3. Explique o benefício, não a tecnologia

	G4. Seja responsável com os erros
	G5. Invista cedo em boas práticas com os dados
	G6. Balanceie precisão e *recall* com cuidado
	G7. Seja transparente sobre configurações de privacidade e dados
	G8. Faça-o seguro de explorar
	G9. Ancore-se no familiar
	G10. Adicione contexto a partir de fontes humanas
	G11. Determine como mostrar a confiança do modelo, e se deve mostrar
	G12. Explique com foco em entendimento, não completude
	G13. Vá além de explicações “no momento”
	G14. Automatize mais quando o risco é pequeno
	G15. Deixe usuários darem *feedback*
	G16. Permita usuários supervisionarem a automação
	G17. Automatize em fases
	G18. Devolva o controle ao usuário quando a automação falha
	G19. Projete para os seus classificadores de dados
	G20. Tenha manutenção ativa do dataset
	G21. Aprenda com desacordos de classificação
	G22. Aceite dados ruidosos
	G23. Receba conselho de especialistas de domínio na construção o dataset
(MOHSENI, 2021)	G4. Decida como explicar
	G5. Avalie a utilidade da explicação

Fonte: elaborado pelo autor

3.3.3. Qual o suporte existente para avaliação utilizando os conjuntos de heurísticas encontrados?

Nesta questão de análise, busca-se identificar o suporte à avaliação heurística, que pode ser operacionalizada por algum artefato ou ferramenta que possibilite a verificação do nível de conformidade do design de interface de um sistema de software a de um conjunto de heurísticas. Observou-se que nenhum dos trabalhos encontrados apresenta uma *checklist* ou ferramenta para apoiar a avaliação heurística. A maioria dos trabalhos utilizou somente exemplos para ilustrar a aplicação correta de cada heurística. A Tabela 7 apresenta um resumo sobre o suporte fornecido.

Tabela 7 - Suporte à avaliação

Citação	**Descrição com exemplos**	**Checklist**	**Suporte automatizado**
(AMERSHI et al., 2019) (LI et al., 2022) (MICROSOFT, 2022)	Exemplo de como uma aplicação poderia aplicar cada diretriz Exemplos e contra-exemplos da aplicação das diretrizes	não	não

(APPLE, 2022)	Exemplos de algumas diretrizes	não	não
(DUDLEY, 2018)	Sem exemplos	não	não
(GOOGLE, 2022)	Exemplo e contra-exemplo de aplicação, e caso de uso que implementa	não	não
(MOHSENI, 2021)	Exemplo feito sobre um estudo de caso dos autores	não	não

Fonte: elaborado pelo autor

3.3.4. Como o conjunto de heurísticas foi desenvolvido e avaliado?

Para assegurar a corretude de uma avaliação heurística é importante que o conjunto de heurísticas se refira às melhores práticas que resultam em boa UX, definindo as heurísticas de forma confiável e válida. Dessa maneira é importante que os conjuntos de heurísticas sejam desenvolvidos e avaliados sistematicamente (RUSU, 2011).

Entre os trabalhos encontrados, as heurísticas de Amershi et al. (2019, 2022), apresentaram a maior transparência sobre a forma como foram desenvolvidas, relatando também avaliações em duas instâncias. Conforme apresentado em por Amershi et al. (2019), as 18 diretrizes propostas foram obtidas a partir de uma síntese de princípios encontrados na literatura e em aplicações e guias de estilo da indústria, e são refinados a partir do uso das heurísticas em avaliações de produtos, primeiramente por diversos participantes, e depois por especialistas. Os fatores utilizados para o refinamento das heurísticas foram “relevância”, medida como aplicabilidade de cada heurística sobre os diferentes cenários de interação com IA, e “clareza”, que é a capacidade de uma heurística de ser interpretada corretamente.

Continuando a pesquisa (Li et al., 2022), foram executados 18 estudos fatoriais 2x2 que analisavam as percepções dos participantes sobre produtos que aplicam ou violam as diretrizes e efeitos de aplicar e violar sobre a UX do produto. Desses 18 estudos, 16 produziram um resultado tangível, e apontaram para a validade das diretrizes. Em cada um desses estudos, são manipuladas variáveis dependentes que representam o impacto de cada heurística sobre métricas de UX, utilizando a medida Eta Quadrado Generalizado (η^2_G) (OLEJNIK, 2003) aplicada a escala de tamanho de efeito de Cohen (2013). Um resumo dos valores obtidos para as métricas de UX pode ser visualizado na Tabela 9. Apesar dos resultados positivos, os autores alertam sobre a

importância de considerar o contexto da aplicação para a implementação de sistemas conforme as heurísticas.

Dudley e Kristensson (2018) e Mohseni et al. (2021) desenvolveram as heurísticas a partir de revisão sistemática, porém não apresentaram informações de avaliação de confiabilidade e validade. No ambiente industrial, Google (2022a) desenvolveu suas heurísticas a partir de revisão sistemática e estudos internos. Não foram encontradas informações referentes ao método de desenvolvimento adotado pela Apple (2022). Ambas as empresas também não forneceram informações sobre a avaliação das heurísticas propostas. Os métodos de desenvolvimento e validação são apresentados na Tabela 8.

Tabela 8 - Métodos de desenvolvimento e avaliação de heurísticas adotadas

Citação	**Método de desenvolvimento das heurísticas**	**Avaliação da confiabilidade e validade das heurísticas**
(AMERSHI et al., 2019) (LI et al., 2022) (MICROSOFT, 2019)	Compilação de 168 princípios relacionados a IA provindos da indústria e literatura acadêmica em 18 diretrizes	Avaliação heurística modificada com 11 membros da equipe avaliando 13 produtos com AI; Estudo de usuário com 49 praticantes de IHC, avaliando 20 produtos; Avaliação de especialista com 11 especialistas de UX/IHC revisando as heurísticas modificadas; 18 estudos independentes, um para cada uma das heurísticas, com 1043 participantes em forma de estudo fatorial 2x2
(APPLE, 2022)	Não informado	Não informado
(DUDLEY, 2018)	Revisão sistemática	Não informado
(GOOGLE, 2022a)	Revisão sistemática e estudos internos	Não informado
(MOHSENI, 2021)	Revisão sistemática	Não informado

Fonte: elaborado pelo autor

Tabela 9 - Variação do η^2_G nas variáveis dependentes

Métrica de UX	**Variação do η^2_G sobre os 16 estudos**
Sentindo mais em controle	0,3595
Sentindo menos inadequado	0,1487
Sentindo mais produtivo	0,2926
Sentindo mais seguro	0,2406
Sentindo menos incerto	0,2907
Confiança	0,3094
Confiabilidade	0,2168
Diminuição na suspeita	0,3769
Diminuição na expectativa de ferir	0,2472
Desempenho percebido	0,2584

Utilidade Percebida	0,2775
Net Promoter Score (REICHHELD, 2011)	0,3075
Intenção comportamental	0,3066

Fonte: elaborado pelo autor

3.3.5. Discussão

Como resultado dessa revisão sistemática foi observada a existência de poucas pesquisas voltadas ao desenvolvimento de heurísticas para avaliação de AIX. Entre as publicações encontradas, apesar de não indicarem um escopo específico, é possível perceber que grande parte concentra-se em sistemas de recomendação, por exemplo a proposta de Amershi et al. (2019) com quase a metade das 18 heurísticas voltadas a sistemas que adaptam suas funcionalidades baseados na ações do usuário. Não foi encontrado nenhum conjunto de heurísticas projetado especificamente para a tarefa de classificação de imagens.

Referente aos conjuntos de heurísticas encontradas, não é informado um tipo de dispositivo específico, assumindo-se que podem ser utilizados em qualquer um. Percebe-se que também nenhuma pesquisa concentra o escopo de aplicação das heurísticas para aplicativos exclusivamente.

Alguns dos conjuntos de heurísticas encontrados apresentaram formatos variados, como Mohseni et al. (2021), em que as heurísticas para avaliação de interfaces gráficas são um subconjunto de heurísticas de todo o sistema, incluindo também heurísticas voltadas ao estabelecimento de requisitos e objetivos para a explicabilidade no sistema, e heurísticas para o projeto de algoritmos interpretáveis. No conjunto da Apple (2022), os 60 princípios estão fracamente agrupados em o que se resume a 9 heurísticas extensas.

Observa-se também a falta de suporte à avaliação heurística nas pesquisas encontradas, basicamente com nenhuma apresentando *checklists* ou uma ferramenta de suporte para (semi-) automatizar a avaliação. Além da apresentação das heurísticas, a maioria das pesquisas limita-se ao método de comparação com um exemplo e contraexemplo de aplicação das heurísticas.

Em relação ao método de desenvolvido adotado para criar os conjuntos de heurísticas destaca-se a pesquisa apresentada por Amershi et al. (2019) sendo os únicos a refinar as diretrizes com uma avaliação heurística

modificada, um estudo de usuário e uma avaliação com especialistas, além de também serem os únicos a apresentar avaliações da validade e confiabilidade das heurísticas.

Os demais trabalhos que informam o método de desenvolvimento utilizado para desenvolver os conjuntos de heurísticas empregaram revisões sistemáticas sem informar mais nenhum método específico além dessa revisão.

Assim os resultados dessa revisão sistemática indicam a falta de um conjunto de heurísticas de AIX projetado especificamente para aplicativos de classificação de imagens, bem como o suporte à avaliação.

Ameaças a validade. Existem fatores que podem ameaçar a validade deste trabalho. Primeiramente, existe a não identificação de algum estudo relevante, que se manifesta neste trabalho na forma de não ter encontrado todos os conjuntos de heurísticas relevantes. As medidas tomadas para minimizar este risco foram a busca em diversas bases e a inclusão de sinônimos na *search string*.

Além disso, existem ameaças na seleção de publicações, manifestando na exclusão indevida de um conjunto de heurísticas relevantes. Por último, existem ameaças na extração de informações, em que o viés do autor e restrições de tempo prejudicam a qualidade das informações obtidas. Ambas essas ameaças foram tratadas a partir da definição explícita de critérios de inclusão e exclusão e por revisões de todo o processo pela orientadora.

4. DESENVOLVIMENTO DE HEURÍSTICAS E CHECKLIST PARA AIX E CLASSIFICAÇÃO DE IMAGENS

4.1 REQUISITOS/CONTEXTO DA SOLUÇÃO

No escopo do presente trabalho é definido um novo conjunto de heurísticas de usabilidade de IA, focando em aplicativos móveis de classificação de imagens. Inserido no contexto de pesquisa da iniciativa Computação na Escola/INCoD/INE/UFSC, visa-se também a customização destas heurísticas em um contexto educacional. Para apoiar a operacionalização da avaliação heurística é desenvolvido também um *checklist* a partir desse conjunto de heurísticas.

Enfocando em aplicativos de classificação de imagens, são considerados aplicativos que apresentam um fluxo de execução em que o usuário fornece uma imagem de um objeto e o sistema retorna a classificação deste objeto de acordo com o modelo de DL implantado, indicando como por exemplo, graus de confiança a cada uma das classes no modelo.

No contexto educacional visa-se focar em apps inteligentes criados com App Inventor (MIT, 2022) implantando modelos criados com Google Teachable Machine (GOOGLE, 2022d) usando a extensão TMIC (OLIVEIRA, 2022). *Teachable Machine Image Classifier* (TMIC) (OLIVEIRA, 2022) é uma ferramenta visual que permite a importação e o uso de modelos de classificação de imagens, projetada para dar suporte ao fluxo de trabalho do desenvolvimento dos modelos de ML.

Como um exemplo de aplicativo que se encaixa nesses requisitos, apresenta-se o aplicativo PodePão (GQS, 2023), projetado para identificar se um pão está livre de fungos e seguro para o consumo. Neste aplicativo, o usuário começa recebendo uma curta introdução às capacidades do sistema (Figura 5), é levado a uma tela onde ele pode tirar uma foto do pão (Figura 6) e após retirar a foto e um curto período de processamento da classificação é apresentado uma sugestão de como proceder com esse pão (Figura 7).

Observa-se que nesse contexto de criação de apps de classificação de imagens com GTM existe atualmente uma limitação, em que não é possível

realimentar automaticamente o modelo de classificação de imagens com o *feedback* do usuário para a melhoria contínua do desempenho do modelo.

Figura 5 - Apresentação inicial das capacidades do PodePão

Figura 6 - Tela para classificação de um pão

Figura 7 - Resultados da classificação do pão

Fonte: elaborado pelo autor

4.2 MAPEAMENTO DAS HEURÍSTICAS ENCONTRADAS

Para iniciar o processo de criação do novo conjunto de heurísticas, seguindo a metodologia de Rusu et al. (2011), foi feita uma análise dos conjuntos de heurísticas encontrados como resultado do levantamento do estado da arte para destacar as informações relevantes providas por cada um. Como resultado é proposto um mapeamento de todas as heurísticas, identificando a correlação entre os conjuntos encontrados e explicitando todas as diretrizes que compõem estas heurísticas.

Neste mapeamento, ainda não é considerado o contexto específico de classificação de imagens e o App Inventor. Entretanto, diretrizes que foram identificadas referentes ao desenvolvimento do modelo de ML de forma geral, sem nenhum impacto no projeto de UI, foram excluídas no mapeamento. Na unificação foram consideradas equivalentes as diretrizes que se referem a

princípios similares. Em casos em que um grupo de diretrizes de uma publicação possui um equivalente único em outra publicação que os agrupa, como as apresentadas por Apple (2022) em relação às outras publicações, optou-se pela diretriz única. Os resultados, apresentados na Tabela 10, mostram o mapeamento de todas as diretrizes em um conjunto de 33 itens. Já na Tabela 11 são apresentadas explicações de cada diretriz.

Tabela 10 - Mapeamento das diretrizes por conjuntos

Nº	Diretriz	AMERSHI	APPLE	DUDLEY	GOOGLE	MOHSENI
1	Deixe claro o que o sistema pode fazer	x	x	x	x	–
2	Deixe claro o quão bem ele pode fazer	x	x	x	x	–
3	Mostre informações contextualmente importantes	x	–	–	–	x
4	Adeque-se às normas sociais relevantes	x	–	–	–	–
5	Mitigue os viéses sociais	x	–	–	–	–
6	Dê suporte à invocação eficiente	x	–	–	–	–
7	Dê suporte à dispensa eficiente	x	x	–	x	–
8	Dê suporte à correção eficiente	x	x	x	x	–
9	Deixe claro por que o sistema agiu dessa maneira	x	–	–	–	–
10	Lembre de interações recentes	x	x	–	–	–
11	Aprenda com o comportamento do usuário	x	x	x	x	–
12	Atualize e adapte com cuidado	x	x	–	x	–
13	Estimule o *feedback* granular	x	x	x	x	–
14	Comunique ao usuário as consequências de suas ações	x	–	–	x	–
15	Providencie controles globais	x	x	–	x	–
16	Notifique usuários sobre mudanças	x	x	–	–	–
17	Explique o benefício, não a tecnologia	–	–	–	x	–
18	Seja transparente sobre configurações de privacidade e dados	–	x	–	x	–
19	Ancore-se no familiar	–	–	–	x	–
20	Adicione contexto a partir de fontes humanas	–	–	–	x	–
21	Determine como mostrar a confiança do modelo, e se deve mostrar	–	x	x	x	x
22	Explique com foco em entendimento, não completude	–	x	x	x	x
23	Vá além de explicações “no momento”	–	–	–	x	x
24	Solicite *feedback* explícito somente quando necessário	–	x	–	–	–
25	Faça com que o providenciamento de *feedback* explícito seja voluntário	–	x	–	–	–
26	Não peça por ambos *feedback* positivo e negativo	–	x	–	–	–
27	Evite pedir a participação na calibração mais de uma vez	–	x	–	–	–
28	Faça com que a calibração seja fácil e rápida	–	x	–	–	–
29	Quando possível, sugira correções guiadas ao invés de correções livres	–	x	–	–	–
30	Quando possível, trate erros sem complicar a UI	–	x	–	–	–
31	Princípios sobre múltiplas opções	–	x	–	–	–
32	Demonstre como conseguir os melhores resultados	–	x	–	–	–
33	Avalie a utilidade da explicação	–	–	–	–	x

Fonte: elaborado pelo autor

Observa-se pelo mapeamento que há, entre as publicações, uma concordância maior sobre a importância dos princípios que tratam de apresentação inicial das capacidades do sistema de IA, devolução de controle ao usuário em caso de erro, coleta de *feedback* para aperfeiçoamento contínuo do modelo de ML e formatação dos resultados para apresentação para o usuário.

Por outro lado, 17 diretrizes estão presentes em apenas uma das conjuntos analisadas, com diversas questões tendo importância atribuída em somente um destes. Como exemplo, somente Amershi et al. (2019) trata da questão de respeito a normas sociais e o conjunto da Google (2022a) é o único a ressaltar a importância de não confundir o usuário com uma UI diferente do padrão para o dispositivo, que prejudicaria ainda mais o modelo mental do usuário. Estas observações demonstram que ainda não há um consenso comum sobre as heurísticas de *user experience* para sistemas com IA.

4.3 PROPOSTA DE HEURÍSTICAS

Nesta etapa, são analisadas as características específicas de aplicativos de classificação de imagens desenvolvidos com a extensão TMIC no contexto da criação de apps App Inventor e modelos de DL criados com GTM para refinar e especializar as heurísticas obtidas pelo mapeamento.

Em preparação, foram analisados pelo autor todos os aplicativos da iniciativa computação na escola com classificação de imagens (COMPUTAÇÃO NA ESCOLA, 2023). Além desses, para comparações, 5 aplicativos comerciais foram analisados:

- Google Lens, um classificador de objetos em geral (GOOGLE PLAY, 2023a).
- Calorie Mama, um gerenciador de dieta com funcionalidade de reconhecimento de calorias a partir de uma imagem do alimento (GOOGLE PLAY, 2023b).
- Picture Insect, um classificador de insetos em geral (GOOGLE PLAY, 2023c).

- Dog identifier: Dog Scanner, um classificador de raças de cães (GOOGLE PLAY, 2023d).
- Gemius: Rock Identifier - Ston, um aplicativo para reconhecer rochas (GOOGLE PLAY, 2023e).

4.3.1 Refinamento do mapeamento

Começando com considerações gerais, nenhum aplicativo desenvolvido com TMIC tem capacidade de atualizar o seu modelo de classificação sem que ele seja manualmente re-treinado pelo *Teachable Machine* (OLIVEIRA, 2022). A consequência disso é a incapacidade de implementar a adaptação do modelo durante o uso. Por conta disso, os itens que se referem à atualização dinâmica do modelo e automatização (12, 13, 15, 16, na Tabela 12), são desconsiderados.

É relevante também o fato que não é possível extrair explicações sobre as decisões que o modelo tomou para a classificação com o *Teachable Machine*, a saída deste é apenas os graus de confiança (GOOGLE, 2022d). Por conta disso, o item 9, que se refere a explicar ao usuário os fatores envolvidos na decisão do modelo, é excluído.

Outra consideração importante é o fato de aplicativos de classificação de imagens não utilizarem os resultados anteriores de um usuário para personalizar o resultado para este usuário, então todas as heurísticas voltadas a isso podem ser removidas (10, 11, 14 na Tabela 12). Com isso, a calibração de uso sugerida por uma heurística da Apple (2022) torna-se desnecessária e também é removida (27 e 28 na Tabela 12).

Sobre questões mais específicas, a análise dos casos de uso mostrou que esses aplicativos geralmente apresentam um único e direto contexto de uso, que envolve simplesmente a tarefa de classificação de imagens. Porém existem também apps que a classificação de imagens é um funcionalidade secundária do aplicativo (GOOGLE PLAY, 2023b) embutida em sistemas de recomendação. Por conta desses aspectos, não há necessidade de verificar se as informações são relevantes ao contexto e o item 3 pode ser removido (Tabela 12).

Voltando aos casos em que a principal funcionalidade de IA era a classificação de imagens, como a tarefa do usuário necessariamente precisa do uso do modelo de ML para ser resolvida, não faz sentido incluir heurísticas para verificar se o modelo pode ser facilmente invocado ou dispensado quando necessário. Com isso, os itens 6 e 7 foram removidos (Tabela 12).

Sobre aspectos sociais, contextualização a partir de fontes humanas, da maneira que foi definida por GOOGLE (2022a) é uma heurística que aparenta não ser cabível à classificação de imagens, já que não é possível se apoiar sobre uma afirmação feita por uma instituição ou comunidade para decidir sobre a classe de um objeto em uma imagem. Porém, ainda é de interesse em casos em que uma classificação errada pode trazer risco ao usuário avaliar se é ofertada a verificação do usuário por especialistas humanos. Por conta disso, o item 20 foi mantido, mas com semântica alterada (Tabela 12). Adicionalmente, não foram observados casos em que um aplicativo pudesse ir contra alguma norma social. Em consequência, o item 4 foi descartado (Tabela 12).

Apesar disso, a classificação de imagens continua tendo desafios com viés, com pesquisas como a de Tong (2020) mostrando casos em que um conjunto de dados para jogadores de voleibol apresentava mais imagens de pessoas com tons de pele claro, e outro de basquete tons de pele escuro. Nestes experimentos, o modelo acabou por incorporar a cor de pele na classificação, que idealmente dependeria apenas dos uniformes e do ambiente onde a foto foi tirada. O *checklist* final ainda deve considerar esse problema, então o item 5 é mantido (Tabela 12).

Finalmente, como o checklist terá seu escopo limitado ao aplicativo em si, avaliar a presença de material externo de apoio não é interessante, excluindo o item 23. Além disso, o *checklist* não tem como objetivo especificar um estilo de UI considerado “mais correto”, portanto os itens 19 e 31 são removidos (Tabela 12).

O restante das heurísticas foram consideradas apropriadas para o contexto, e casos em que elas são seguidas ou violadas já puderam ser observadas nos casos de uso estudados (Picture Insect respeitou o item 1, ao passo que o Google Lens não), então estas serão mantidas para a próxima etapa.

Tabela 11 - Explicação das diretrizes mapeadas

Nº	Diretriz	Explicação
1	Deixe claro o que o sistema pode fazer	Informe o usuário sobre as capacidades do sistema
2	Deixe claro o quão bem ele pode fazer	Informe ao usuário as limitações do sistema, a gravidade dos erros e o quão frequentemente eles ocorrem.
3	Mostre informações contextualmente importantes	Leve o ambiente do usuário e sua tarefa atual em conta para mostrar informações que são relevantes a ele
4	Adeque-se às normas sociais relevantes	Certifique-se que a experiência de uso é de uma maneira esperada pelos usuários, dado o contexto sócio-cultural destes
5	Mitigue os vieses sociais	Garanta que a linguagem e comportamento do sistema não reforçam estereótipos e viés injustos ou indesejáveis
6	Dê suporte à invocação eficiente	Deixe fácil solicitar os serviços do sistema de IA quando necessário
7	Dê suporte à dispensa eficiente	Deixe fácil para dispensar ou ignorar os serviços de IA indesejáveis
8	Dê suporte à correção eficiente	Facilite a edição, refinamento e recuperação manual quando o sistema de IA está errado
9	Deixe claro por que o sistema agiu dessa maneira	Possibilite que o usuário acesse uma explicação por trás das ações da IA
10	Lembre de interações recentes	Mantenha uma memória de curto prazo sobre as interações e permita que o usuário referencie ela
11	Aprenda com o comportamento do usuário	Com o tempo, aprenda com as ações do usuário e personalize a experiência dele
12	Atualize e adapte com cuidado	Limite mudanças disruptivas quando for atualizar e adaptar os comportamentos da IA
13	Estimule o *feedback* granular	Permita que o usuário providencie *feedback* indicando suas preferências em intervalos regulares de uso
14	Comunique ao usuário as consequências de suas ações	Informe ao usuário como suas ações vão impactar no comportamento futuro do sistema de IA
15	Providencie controles globais	Permita que o usuário controle globalmente o que o sistema de IA monitora e como ele se comporta
16	Notifique usuários sobre mudanças	Informe ao usuário todas as adições e atualizações sobre as capacidades do sistema de IA
17	Explique o benefício, não a tecnologia	Ajude os seus usuários a entenderem as capacidades do sistema, em vez de como a tecnologia funciona atrás dos panos
18	Seja transparente sobre configurações de privacidade e dados	Desde o primeiro uso, comunique sobre as configurações de uso de dados e permissões necessárias ao sistema
19	Ancore-se no familiar	Use elementos de UI familiares ao usuário ao apresentar um sistema de IA
20	Adicione contexto a partir de fontes humanas	Use recomendações de fontes de terceiros para trazer confiança às recomendações do sistema
21	Determine como mostrar a confiança do modelo, e se deve mostrar	Se o sistema mostra níveis de confiança ao usuário, faça isso de uma forma que seja útil a este.
22	Explique com foco em entendimento, não completude	Foque em passar aos usuários apenas a informações que precisam no momento, não uma visão completa do funcionamento do sistema

23	Vá além de explicações “no momento”	Ajude os usuários a entenderem melhor como o sistema funciona, com explicações mais detalhadas, fora dos fluxos de uso imediatos dele
24	Solicite *feedback* explícito somente quando necessário	Evite solicitar ao usuário uma ação de preencher *feedback* quando este pode ser aprendido implicitamente com as suas ações
25	Faça com que o providenciamento de *feedback* explícito seja voluntário	Comunique como o providenciamento de *feedback* pode melhorar o sistema, sem fazer com que os usuários sintam que isso seja obrigatório para o uso dele
26	Não peça por ambos *feedback* positivo e negativo	Sugestões corretas não precisam de *feedback*, considere-os implicitamente positivos. Ao invés, dê ao usuário a possibilidade de dar *feedback* negativo em resultados que não os agradam
27	Evite pedir a participação na calibração mais de uma vez	Calibre o sistema ao usuário uma única vez, o mais cedo possível. Use o *feedback* para evoluir e adaptar o sistema às necessidades do usuário
28	Faça com que a calibração seja fácil e rápida	Na calibração, foque em obter apenas as informações que não podem ser inferidas, evite pedir por informações que o usuário deve procurar e não peça a ele ações que podem ser difíceis.
29	Quando possível, sugira correções guiadas ao invés de correções livres	Em comparação a correções livres, correções guidas apresentam alternativas próximas ao resultado e precisam de menos esforço por parte do usuário
30	Quando possível, trate erros sem complicar a UI	Balanceie os efeitos sobre a correção de erros de um padrão de UI em relação a sua capacidade de complicar a UI. Quando um padrão complexo ainda resulta em erro, o efeito é magnificado para o usuário.
31	Princípios sobre múltiplas opções	Apresente diversas opções de resultado ao usuário e permita que ele escolha a solução mais cabível. Foque em diversidade de opções, deixando-as fáceis de distinguir, mas não liste muitas opções, apresentando as mais prováveis primeiro
32	Demonstre como conseguir os melhores resultados	Ajude o usuário a construir um modelo mental melhor do sistema e usá-lo com mais eficácia usando elementos de UI que sutilmente o guiam e sugerem maneiras de atingir um resultado mais preciso
33	Avalie a utilidade da explicação	Certifique-se que toda explicação do sistema apresente algum benefício no entendimento do contexto, na satisfação do usuário com o sistema e com a melhoria do modelo mental do usuário.

Fonte: elaborado pelo autor

Tabela 12 - Heurísticas criadas a partir da seleção de diretrizes relevantes ao contexto de classificação de imagens

Heurística	**Nº**	**Diretriz**	**Inclusão para o contexto**	**Justificativa**
Deixar as expectativas e limitações explícitas	1	Deixe claro o que o sistema pode fazer	x	–
	2	Deixe claro o quão bem ele pode fazer	x	–
	17	Explique o benefício, não a tecnologia	x	–
Apoiar o uso efetivo	32	Demonstre como conseguir os melhores resultados	x	–
Apoiar a compreensão do usuário sobre incerteza e a confiança do modelo	21	Determine como mostrar a confiança do modelo, e se deve mostrar	x	–
	22	Explique com foco em entendimento, não completude	x	–
	33	Avalie a utilidade da explicação	x	–
Assegurar	18	Seja transparente sobre configurações de privacidade e	x	–

privacidade e segurança de dados		dados		
Falhar graciosamente e suportar a recuperação de erros	8	Dê suporte à correção eficiente	x	–
	20	Adicione contexto a partir de fontes humanas	x	–
	29	Quando possível, sugira correções guiadas ao invés de correções livres	x	–
	30	Quando possível, trate erros sem complicar a UI	x	–
Possibilitar coleta de feedback do usuário	24	Solicite *feedback* explícito somente quando necessário	x	–
	25	Faça com que o providenciamento de *feedback* explícito seja voluntário	x	–
	26	Não peça por ambos *feedback* positivo e negativo	x	–
Mitigar viés	5	Mitigue os viéses sociais	x	–
–	3	Mostre informações contextualmente importantes	–	Classificação de imagens apresenta um único contexto: a tarefa em si
–	4	Adeque-se às normas sociais relevantes	–	Pouca relevância para classificação de imagens
–	6	Dê suporte à invocação eficiente	–	Tarefa de Classificação de imagens assume o uso do modelo de ML e não é uma funcionalidade ativada/desativada
–	7	Dê suporte à dispensa eficiente	–	Tarefa de Classificação de imagens assume o uso do modelo de ML e não é uma funcionalidade ativada/desativada
–	9	Deixe claro por que o sistema agiu dessa maneira	–	*Teachable Machine/TMIC* não permite observar as *features* do modelo e/ou não fornece uma explicação
–	10	Lembre de interações recentes	–	Classificação de imagens não se beneficia de construir um perfil para o usuário
–	11	Aprenda com o comportamento do usuário	–	Classificação de imagens não se beneficia de construir um perfil para o usuário
–	12	Atualize e adapte com cuidado	–	*Teachable Machine/TMIC* não possui capacidade de atualizar o modelo de classificação

				dinamicamente
–	13	Estimule o *feedback* granular	–	*Teachable Machine/TMIC* não possui capacidade de atualizar o modelo de classificação dinamicamente
–	14	Comunique ao usuário as consequências de suas ações	–	Classificação de imagens não se beneficia de construir um perfil para o usuário
–	15	Providencie controles globais	–	Não há interação com IA personalizável em classificação de imagens
–	16	Notifique usuários sobre mudanças	–	*Teachable Machine/TMIC* não possui capacidade de atualizar o modelo de classificação dinamicamente
–	19	Ancore-se no familiar	–	As heurísticas não devem limitar estilos
–	23	Vá além de explicações “no momento”	–	As heurísticas dizem respeito ao aplicativo em si
–	27	Evite pedir a participação na calibração mais de uma vez	–	Classificação de imagens não apresenta calibração da IA
–	28	Faça com que a calibração seja fácil e rápida	–	Classificação de imagens não apresenta calibração da IA
–	31	Princípios sobre múltiplas opções	–	As heurísticas não devem limitar estilos

Fonte: elaborado pelo autor

4.3.2 Especificação das heurísticas

O conjunto de diretrizes selecionado na etapa anterior é formalizado nesta seção, utilizando o modelo padrão proposto por Rusu et al. (2011), que define para cada heurística:

- Identificador, nome e definição
- Explicação
- Exemplos de aplicação

- Benefícios
- Problemas com má-interpretação (se existirem)

Diretrizes muito próximas são combinadas em heurísticas nesta etapa, conforme explicado na Tabela 12. Em adição a estas, alguns novos aspectos não encontrados no mapeamento foram incorporados às heurísticas, como por exemplo, a questão de considerar riscos ao usuário quando o aplicativo classifica objetos perigosos.

(1) Deixar as expectativas e limitações explícitas: Ajude o usuário a entender as capacidades e limitações do sistema, ajustando suas expectativas a fim de não vender a “magia da IA”. Evite fazer uso de linguajar técnico para isso.

Apresente as capacidades do sistema, de um modo que fique claro e bem explícito que em que situações ou problemas ele pode auxiliar. Concentre-se nos benefícios que o sistema pode trazer, não mencionando detalhes técnicos de como ela pode providenciá-los. Também faça com que o usuário saiba das limitações do sistema, para que não crie uma confiança no modelo de classificação.

Exemplo: Um aplicativo classifica cogumelos para determinar se são seguros para consumo. Em uma tela de apresentação do aplicativo é deixado claro que “Classificador de cogumelos te ajuda a identificar macrofungos, dando uma indicação inicial se eles são seguros para o consumo adulto”.

Contra-exemplo: O mesmo classificador de cogumelos acima, mas extrapolando suas capacidades e vendendo a “magia da IA”. “Classificador de cogumelos é um especialista em micologia que cabe no seu bolso. Treinado com técnicas modernas de *deep learning* e utilizando um vasto e diverso *dataset.* Use ele para identificar qualquer fungo”.

Deixar as expectativas e limitações explícitas traz a vantagem de temperar o modelo mental do usuário em relação ao sistema, amenizando frustrações quando o sistema errar na classificação.

Essa heurística não impede o uso de linguajar técnico em aplicativos, apenas sugere que isso não seja feito na apresentação das capacidades dele.

(2) Apoiar o uso efetivo: Ajude o usuário a interagir efetivamente com o sistema indicando boas práticas para o uso.

O resultado de uma classificação depende da qualidade da imagem de entrada fornecida pelo usuário. Apesar de que o modelo pode ser treinado levando em conta dados “ruidosos”, o usuário deve saber o que é uma imagem com qualidade boa para a classificação. Evite apresentar todas as boas práticas de uma vez só, sobrecarregando o usuário. Ao invés disso apresente-as à medida que são relevantes (pedir que o usuário leve em conta a iluminação após inserir uma imagem muito escura).

Exemplo: PodePão classifica mofo em pães para saber se estão próprios para consumo. Em sua tela inicial, informa o usuário a alinhar a câmera do celular de um modo que o pão ocupe toda a tela, tratando apenas o problema mais comum nesse contexto: escala do objeto.

Essa heurística traz o benefício de permitir o usuário a ajudar o sistema na classificação, resultando em classificações com mais acurácia.

Não deve-se criar um “manual de uso” que detalha condições para a imagem que devem ser seguidas para que o sistema funcione.

(3) Apoiar a compreensão do usuário sobre incerteza e a confiança do modelo: Possibilite a compreensão dos resultados incertos que serão produzidos pelo modelo e permita com que usuários interessados aprendam como o sistema funciona.

Faça com que o usuário consiga perceber a incerteza inerente do sistema, e consiga levar isso em conta mesmo quando os resultados da classificação de imagens são utilizados em outras funcionalidades do aplicativo.

Além disso, considere a utilidade de apresentar os níveis de confiança crus obtidos pelo modelo ao usuário, pois existem casos onde ele não será capaz de correlacionar um valor numérico a uma conclusão que pode ser tirada deste valor. Ao invés disso, determine interpretações textuais a partir de limiares de confiança e apresente-as aos usuários.

Exemplo 1: Um aplicativo ajuda a identificar a raça de um cachorro dentre as classes dachshund, pitbull, poodle e rottweiler. Após a classificação, o usuário é informado que o seu cachorro se parece 30%, 55%, 5% e 10% com

cada uma das classes, respectivamente. Isso é o suficiente para que ele perceba que o seu cachorro é mais próximo de um dachshund e um pitbull do que os outros.

Exemplo 2: Um outro aplicativo utiliza imagens de alimentos para determinar se eles ainda estão frescos, com a intenção de ser usado após as datas de validade destes expirarem. Depois de tirar a foto de uma fatia de presunto, é apresentado ao usuário que é “É provável que esteja fresco”.

Contra-exemplo: O mesmo aplicativo acima, mas a classificação é apresentada como “83,6% fresco!”, levando o usuário a questionar o perigo relacionado aos 16,4% de chance de estar estragado.

Seguir estes princípios ajudará o usuário a tomar decisões a partir dos resultados com modelo com entendimento de quão corretos eles estão.
Essa heurística não determina casos onde se deve usar o intervalo de confiança e onde deve-se usar uma interpretação dele. Avalie o resultado como um usuário para escolher qual formato usar em seu caso de uso.

(4) Assegurar privacidade e segurança de dados: Explique como os dados do usuário estão sendo usados no sistema, e o motivo por trás de toda permissão solicitada.

Comunique ao usuário sobre como suas imagens estão sendo usadas no modelo de ML. Caso a plataforma alvo precise de permissões para acessar recursos como câmera e armazenamento, explique por que precisa solicitar estas permissões.

Exemplo: Um aplicativo que reconhece linguagem de sinais usando fotos tiradas pelo usuário explica o motivo de estar solicitando permissão para acessar a câmera. Além disso, este possui, em um menu de informações, uma seção que deixa claro que as fotos tiradas são descartadas após o processo de classificação.

Assegurar privacidade e segurança de dados traz o benefício de deixar transparente ao usuário que os recursos de seu telefone celular e os dados que compartilhou com o aplicativo não estão sendo usados de modo malicioso.

Esta heurística não deve ser levada a um extremo onde cada explicação sobre privacidade é extensa, pois pode levar o usuário a desistir de usar o sistema.

(5) Falhar graciosamente e suportar a recuperação de erros: Informe ao usuário sobre os erros de classificação e ofereça alternativas para corrigi-los.

Filtre resultados propensos ao erro, como aqueles onde os níveis de confiança não atinjam um patamar mínimo, informando um erro ao usuário. Em linhas gerais, quando um erro ocorrer, possibilite que o usuário possa rapidamente reiniciar o processo de classificação, verificar o resultado com especialistas humanos, ou ofereça outro método de correção que seja eficiente.

Exemplo: Um classificador de rochas obtém internamente um nível de confiança mais alto de 70%, então informa ao usuário um erro de classificação e navega ao começo da classificação, sugerindo dicas de como tirar uma foto melhor.

Contra-Exemplo: O classificador acima, mas falhando apenas em casos em que há um erro interno, e mesmo nesses casos o usuário é enviado à tela principal do aplicativo antes de poder tentar novamente.

Essa heurística, se seguida, ajuda o usuário a manter confiança nas capacidades do sistema, ao filtrar casos onde o modelo pode estar potencialmente errado. O suporte eficiente à recuperação de erros serve como lembrete ao usuário das limitações do modelo e permite com que ele consiga chegar mais facilmente a um resultado útil.

(6) Possibilitar coleta de feedback do usuário: Permita que usuários com conhecimento suficiente do domínio possam contribuir com a evolução do modelo de classificação**.**

Dê a usuários especialistas de domínio a capacidade de sinalizar erros de classificação, mas não possibilite que um usuário genérico seja capaz de influenciar no retreinamento do modelo. Quando for coletar *feedback*, deixe bem claro o motivo de estar o solicitando e como as informações coletadas podem impactar no desempenho do modelo.

Exemplo: Um reconhecedor de pessoas famosas possui uma opção que permite que o usuário sinalize um erro, apresentando a possibilidade de ir a uma seção onde é explicado como ele pode contribuir para melhorar o sistema.

Contra-Exemplo: Um reconhecedor de cobras peçonhentas foi feito com a mesma base do aplicativo acima, e permite erroneamente que qualquer usuário possa influenciar nos resultados do sistema.

A coleta controlada do *feedback* traz os benefícios de fazer com que o modelo possa evoluir a partir de dados rotulados unicamente por usuários especialistas, conscientes de como isso afeta o funcionamento do sistema.

Como essa heurística diz respeito a impedir que qualquer usuário possa enviar *feedback* e de simultaneamente possibilitar os usuários especialistas de enviar, e é difícil implementar essa divisão dentro de um aplicativo, os desenvolvedores devem levar em conta qual desses grupos de usuário é uma maioria para decidir sobre a coleta de *feedback*.

(7) Mitigar viés: Garanta que nenhum grupo de usuários possa se sentir excluído ou mal representado pela aplicação.

Em casos de uso em que a classificação envolve fatores humanos, é importante que elementos de interface e material extra como textos, imagens e vídeos não excluam algum grupo social ou reforcem preconceitos injustos associados a eles. Assim como o modelo de ML deve evitar atribuições de cor, gênero e cultura na classificação, as interfaces devem reforçar a inclusividade do sistema.

Exemplo: Um aplicativo identifica marcas de roupas e foi treinado para apenas considerar a roupa em si, não o indivíduo a vestindo. Suas interfaces são então projetadas para mostrar diversos grupos sociais interagindo com o aplicativo e não apresentam grupos étnicos a partir de seus vestimentos estereotípicos.

Esta heurística ajuda o aplicativo a alcançar um número maior de usuários e a estabelecer uma relação de respeito com todos eles.

(8) Considerar os riscos ao usuário: Certifique-se que o usuário esteja informado dos possíveis riscos na utilização do sistema.

Garanta que o usuário está ciente dos riscos que corre ao classificar objetos perigosos. Ressalte a incerteza do modelo se um erro de classificação pode trazer danos.

Exemplo: Um classificador de insetos avisa o usuário para manter distância e usar o zoom da câmera ao selecionar a opção de tirar uma foto para classificação.

Exemplo 2: Um aplicativo que classifica cobras peçonhentas alerta ao usuário para que tire uma foto da cobra enquanto acompanhado por uma pessoa alerta, e que não fique no caminho do animal.

Considerar os riscos ao usuário reforça a percepção deste sobre a competência e responsabilidade do aplicativo de fornecer um serviço seguro e confiável, mesmo em situações de risco.

Note que essa heurística se refere inteiramente a sistemas em que existe risco ao usuário.

4.3.3 Especificação do *checklist*

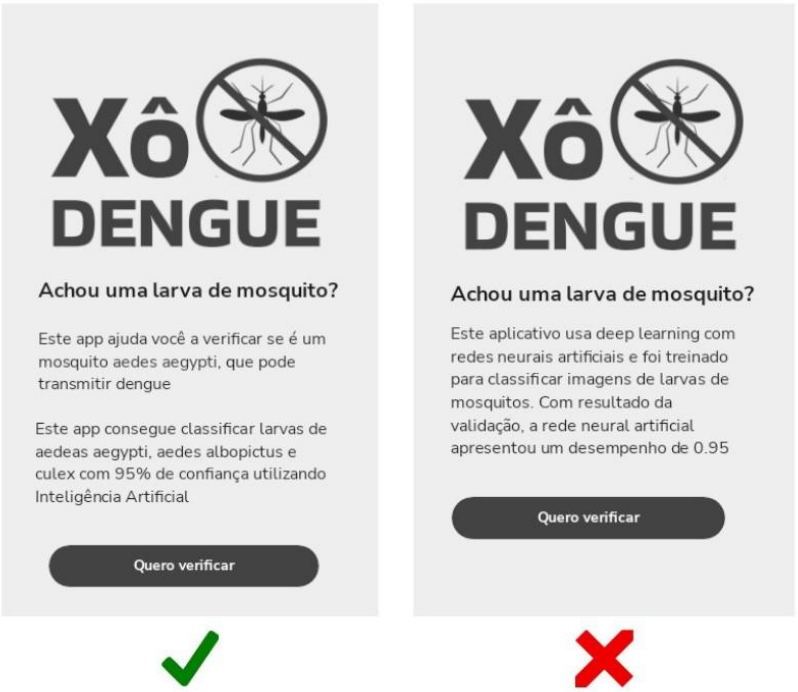
Com as heurísticas definidas, itens de *checklist* são definidos com o objetivo de detalhar os conceitos apresentados anteriormente em perguntas que podem ser utilizadas para avaliar o AIX de interfaces de aplicativos de classificação de imagens.

Cada item de *checklist* é agrupado sob a heurística que avalia, e apresenta:

- Nome.
- Descrição.
- Figuras indicando a aplicação correta e, quando possível, incorreta do item.
- Escala de resposta.

Esta versão inicial do *checklist* desenvolvido pode ser vista na Tabela 13, com todas as figuras de exemplo da versão final disponíveis no Apêndice A. Abaixo são apresentados três exemplos destas figuras, a primeira representando o item "*4. O app faz as explicações de forma compreensível?"* (Figura 8), a segunda o item "*5. O app mostra dicas de como obter imagens com qualidade adequada?"* (Figura 9), e a última o item "*12. O app mostra um aviso quando o sistema não é capaz de classificar com confiança suficiente?"* (Figura 10).

Figura 8 - Exemplo e contraexemplo do item 4 do *checklist*

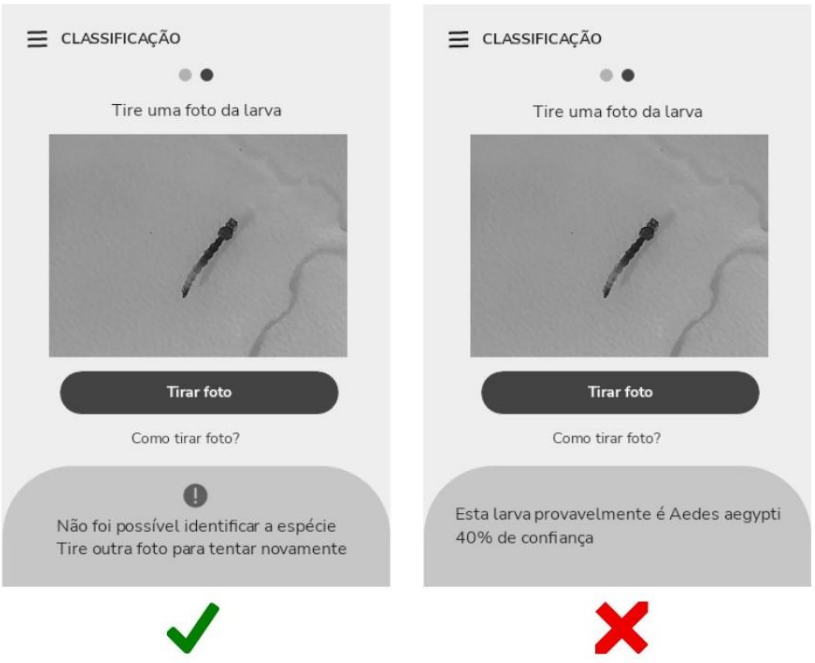


Fonte: elaborado pelo autor

Figura 9 - Exemplo do item 5 do *checklist*

Fonte: elaborado pelo autor

Figura 10 - Exemplo e contraexemplo do item 12 do *checklist*



Fonte: elaborado pelo autor

Tabela 13 - Resumo do *checklist* desenvolvido v0.1

Heurística	Item de *checklist*	Explicação	Escala de resposta
Deixar as expectativas e limitações explícitas	1. O app deixa claro o que pode fazer?	O app apresenta as categorias que é capaz de distinguir antes do usuário poder classificar uma imagem.	Sim, Não
	2. O app informa o grau de desempenho de suas classificações?	O app apresenta ao usuário o grau do seu desempenho (p.ex. acurácia) antes do usuário poder classificar uma imagem.	Sim, Não
	3. O app explicita suas limitações?	O app apresenta o que não é capaz de distinguir antes do usuário poder classificar uma imagem.	Sim, Não
	4. O app faz as explicações de forma compreensível?	O app utiliza uma terminologia compreensível pelo público alvo, evitando jargão técnico, ao apresentar as expectativas e limitações?	Sim, Não
Apoiar o uso efetivo	5. O app mostra dicas de como obter imagens com qualidade adequada?	O app apresenta instruções/dicas para guiar o usuário a obter imagens de qualidade adequada para a classificação.	Sim, Não, N/A
Apoiar a compreensão do usuário sobre incerteza e a confiança do modelo	6. O app demonstra os resultados de uma forma compreensível pelo público alvo?	O resultado da classificação está em um formato compreensível para o usuário do público-alvo. É evitada a apresentação de somente percentuais de confiança.	Sim, Não
	7. O app demonstra os resultados de uma forma útil?	O resultado da classificação está sendo apresentado de uma forma que ajude o usuário a tomar uma decisão de acordo com o caso de uso.	Sim, Não
	8. O app deixa claro que existe incerteza quando utiliza o resultado da classificação em outra funcionalidade?	O sistema apresenta a incerteza de alguma forma no resultado da classificação quando depende deste em outra funcionalidade (p.ex. Marcar locais em um mapa que oferecem o objeto).	Sim, Não, N/A
	9. O app fornece informações sobre como o modelo de ML foi desenvolvido??	O aplicativo mostra informações sobre o desenvolvimento do modelo de ML, incluindo p.ex. informações sobre a quantidade de imagens usadas no treinamento, rotulação feita por quem, que tipo de modelo de ML está sendo usado e qual o desempenho deste.	Sim, Não
Assegurar privacidade e segurança de dados	10. O app disponibiliza informações sobre o uso das imagens do usuário usadas na classificação?	Existe informação sobre como as imagens do usuário a serem classificadas são utilizadas e se elas são (ou não) armazenadas persistentemente. Caso sejam armazenadas, as condições deste armazenamento e o acesso a estas imagens é especificado.	Sim, Não, N/A
Falhar graciosamente e suportar a recuperação de erros	11. O app ajuda o usuário a se recuperar de possíveis erros?	O aplicativo permite que o usuário possa facilmente refazer a classificação quando um erro é identificado, explicando como ele deve fazer.	Sim, Não
	12. O app mostra um aviso quando o sistema não é capaz de classificar com confiança suficiente?	Em casos em que não é possível classificar com confiança suficiente (p.ex < 70%), o app alerta ao usuário informando que não foi capaz de classificar esta imagem (ao invés de mostrar resultados mesmo com grau de confiança baixa).	Sim, Não
	13. O app permite a verificação do resultado por especialistas humanos?	O aplicativo oferece um meio de contato com especialistas do domínio para verificar o resultado.	Sim, Não, N/A
Possibilitar coleta de *feedback* do usuário	14. O app permite que usuários com conhecimento no domínio de aplicação possam enviar *feedback* referente ao resultado da classificação?	O aplicativo possibilita que um usuário com conhecimento de domínio possa indicar se um resultado de classificação está correto ou não.	Sim, Não, N/A
	15. O app não permite que usuários sem conhecimento no domínio de aplicação enviem *feedback* referente ao resultado da classificação?	O aplicativo não oferece a opção de feedback referente a corretude de resultado de classificação a usuários que não poderiam fornecê-lo corretamente.	Sim, Não, N/A

	16. O app deixa claro o propósito de enviar *feedback*?	O aplicativo faz com que o usuário entenda como seu feedback pode afetar no funcionamento, e é guiado a fornecê-lo com cuidado.	Sim, Não, N/A
Mitigar viés	17. O app está livre de viés?	Não há nenhum reforço de viés social, preconceitos ou uso de terminologia inapropriada na interface de usuário.	Sim, Não, N/A
Considerar os riscos ao usuário	18. O app indica os devidos cuidados para a obtenção de imagens?	No caso em que a captura de imagens pode acarretar em risco (p.ex. tentando tirar foto de uma cobra peçonhenta) diretrizes são apresentadas para fazê-lo de um modo seguro e alertar ao usuário do perigo.	Sim, Não, N/A
	19. O app destaca os riscos envolvidos com um possível erro de classificação?	O aplicativo indica as consequências de um possível erro de classificação, especificamente em casos em que isso pode resultar em danos humanos.	Sim, Não, N/A
	20. O app mostra elementos de alerta caso o objeto classificado possa causar danos físicos a humanos?	O aplicativo mostra de forma visual p.ex. usando cores e/ou ícones para alertar o usuário sobre o perigo pelo objeto classificado.	Sim, Não, N/A

Fonte: elaborado pelo autor

4.4 AVALIAÇÃO DAS HEURÍSTICAS

Para verificar a qualidade do conjunto de heurísticas e do *checklist* desenvolvido, seguindo Rusu et al. (2011), é feita uma avaliação com um painel de especialistas.

Entretanto, enquanto Rusu et al. (2011) sugere uma comparação entre as heurísticas de Nielsen (1994) com as desenvolvidas, neste trabalho optou-se por solicitar que os especialistas julguem aspectos como a completude, corretude, consistência e ambiguidade das heurísticas e o *checklist* a partir da aplicação dele em uma avaliação heurística, seguindo a metodologia descrita por Lawshe (1975).

4.4.1 Execução do painel de especialistas

A avaliação por painel de especialistas foi executada em abril de 2023 por seis pesquisadores das áreas de ciência da computação e design. A avaliação foi feita em dois passos: primeiramente o avaliador aplicava o checklist para avaliar um app com classificação de imagens e em seguida avaliava os fatores de qualidade do checklist. A cada especialista foi associado um aplicativo inteligente de classificação de imagens diferente. As instruções, o checklist e o questionário de avaliação foram enviados em forma de um formulário *online* em que as heurísticas e itens do *checklist* eram apresentados e imediatamente utilizados para avaliar este aplicativo inteligente. Após essa tarefa, cada especialista respondeu uma sequência de perguntas para relatar os problemas que identificaram durante o processo:

- Existe alguma heurística ou item de *checklist* que não representa corretamente experiência de usuário para aplicativos inteligentes de classificação de imagens?
- Existe alguma heurística ou item de *checklist* que é irrelevante para o contexto de apps de classificação de imagens no ensino de computação?
- Falta alguma heurística ou item de *checklist*?
- Existe alguma heurística ou item de *checklist* que não está escrito claramente?

- O modelo está apropriadamente decomposto em heurísticas e itens de *checklist*?
- Este *checklist* é fácil de usar para avaliar a experiência de usuário de aplicativos inteligentes de classificação de imagens?
- Você considera este *checklist* aplicável para avaliação de desempenho no ensino de computação na Educação Básica?
- Mais algum comentário ou sugestão?

A avaliação foi respondida por todos os especialistas convidados. Entre esses, no quesito de nível de conhecimento em design de interface de usuário, 5 haviam feito pelo menos uma disciplina na área, e 1 era formado. Quatro dos especialistas já tinham desenvolvido um aplicativo de classificação de imagens com *Deep Learning*.

4.4.2 Análise das respostas

A maioria dos especialistas consideraram o *checklist* útil e aplicável no contexto de avaliação de desempenho no ensino de computação, e o modelo corretamente decomposto em heurísticas e *checklist.*

Sobre itens ou heurísticas que não representam corretamente AIX, apenas um especialista expressou considerações, defendendo a adição de “Não se aplica” aos itens 1 e 3, e a remoção do item 8, por considerá-lo implícito ao contexto.

Em relação a itens irrelevantes para o contexto, um especialista percebeu uma redundância entre os itens 6 e 7, e entre os itens 14 e 15.

Sobre a ausência alguma heurística ou item importante, foi sugerido por um participante um item para informar o usuário que suas imagens estão sendo armazenadas pelo aplicativo, dentro da heurística de *Assegurar privacidade e segurança de dados.*

No quesito de itens ou heurísticas mal redigidas, dois especialistas sugeriram que o item 15 fosse reescrito de uma negação para uma afirmação, a fim de padronizar com os outros 19 itens.

Um comentário geral da maioria dos especialistas é a dependência da compreensão dos itens do *checklist* nas suas imagens de exemplo. Isto

enfatiza que a avaliação heurística não é possível sem a consulta das imagens.

4.5 REFINAMENTO DAS HEURÍSTICAS E CHECKLIST

Utilizando os resultados e comentários do painel de especialistas, um processo de refinamento das heurísticas e *checklist* foi realizado, consistindo na etapa final da metodologia cíclica de desenvolvimento de heurísticas proposta por Rusu et al (2011).

Para começar, foram levantados requisitos de mudança para tratar os problemas destacados anteriormente. Considerando que as heurísticas em si foram julgadas suficientes e bem descritas, apenas o *checklist* foi levado em conta. As mudanças necessárias eram:

- Melhorar a descrição e nomes para diminuir a dependência do *checklist* nos exemplos.
- Reconsiderar os itens opcionais (“Não se aplica”).
- Reescrever o item 15 de uma negação para uma afirmação.

Com os requisitos levantados, uma nova versão do *checklist* foi criada, que pode ser visualizada na Tabela 14. Em comparação à anterior, a versão 0.2 do *checklist* passou a contar, adicionalmente ao requisitos anteriores, com exemplos de condições para escolher a opção “Não se aplica” (N/A) em todo item que a possuir, remoção do antigo item 3 (absorvido pelos itens 1 e 2) e adição de 4 novos itens - marcados em negrito na Tabela 14.

Tabela 14 - *Checklist* v0.2 ajustado depois da avaliação do painel de especialistas

Heurística	**Item de *checklist***	**Explicação do item**	**Escala de resposta**
Deixar as expectativas e limitações explícitas	1. O app deixa claro quais classes ele pode classificar?	O app apresenta as classes que é capaz de distinguir antes do usuário poder classificar uma imagem. (p.ex. na tela home).	Sim, Não
	2. **O aplicativo deixa claro o quão bem ele pode fazer a classificação de imagens?**	O app apresenta ao usuário o grau do seu desempenho (p.ex. acurácia) antes do usuário poder classificar uma imagem. p.ex. na tela home).	Sim, Não
	3. O aplicativo fornece explicações compreensíveis?	O app utiliza apenas uma terminologia compreensível pelo público alvo, evitando jargão técnico, ao apresentar as expectativas e limitações.	Sim, Não
Apoiar o uso efetivo	4. O app mostra dicas de como tirar fotos com qualidade adequada?	O app apresenta instruções/dicas para guiar o usuário a tirar fotos com qualidade adequada para a classificação.	Sim, Não
	5. O app visualiza o status durante o processamento da classificação?	**O app apresenta elementos para visualizar o status do progresso durante o processamento da classificação.**	**Sim, Não**
Apoiar a compreensão do usuário sobre incerteza e a confiança do modelo	**6. O app deixa claro que existe incerteza em relação ao resultado da classificação?**	**O resultado da classificação é apresentado indicando que existe uma incerteza em relação a este resultado.**	**Sim, Não**
	7. O app indica a incerteza de forma compreensível pelo público alvo?	O resultado da classificação é apresentado de forma compreensível para o usuário alvo, p. Ex. usando valores categóricos como alto/médio/baixo ou muito provável/provável/pouco provável ou apresentando as n-melhores alternativas de resposta. É evitada a apresentação de apenas percentuais de confiança. **N/A: caso o app não indica incerteza com o resultado da classificação**	Sim, Não, **N/A**
	8. O app demonstra os resultados de forma útil?	O resultado da classificação está sendo apresentado de forma que ajude o usuário a tomar uma decisão de acordo com o caso de uso (p.ex. na classificação de aranhas peçonhentas a indicação de que tipo de assistência médica deve ser procurada). **N/A: caso o único objetivo é a classificação sem nenhuma outra finalidade**	Sim, Não, **N/A**
	9. O app deixa claro que existe incerteza quando utiliza o resultado da classificação em outra funcionalidade?	O app apresenta a incerteza do resultado de classificação de alguma forma, mesmo quando é usado diretamente como parte de outra funcionalidade (p.ex. marcar locais em um mapa que oferecem o objeto classificado). **N/A: A classificação de imagens é a única função do app**	Sim, Não, N/A
	10. O app fornece informações sobre como o modelo de ML foi desenvolvido?	O app mostra informações sobre o desenvolvimento do modelo de ML, incluindo p.ex. informações sobre a quantidade de imagens usadas no treinamento, por quem as imagens foram rotuladas, que tipo de modelo de ML está sendo usado e qual o desempenho.	Sim, Não
Assegurar privacidade e segurança de dados	11. O app disponibiliza informações sobre o uso das fotos do usuário usadas na classificação?	Existe informação sobre como as fotos do usuário a serem classificadas são utilizadas e se elas são (ou não) armazenadas persistentemente. Caso sejam armazenadas, as condições deste armazenamento e o acesso a estas imagens é especificado.	Sim, Não
Falhar graciosamente e suportar a recuperação de erros	**12. O app permite a recuperação de erros?**	**O app permite que o usuário possa facilmente refazer a classificação quando ocorre um erro de classificação (p.ex. manter o botão de tirar foto).**	**Sim, Não**
	13. O app ajuda o usuário a se recuperar de possíveis erros?	**O app explica o que fazer quando ocorre um erro de classificação (p.ex. Solicitando tirar outra foto).**	Sim, Não

	14. O app indica quando se trata de objetos fora do seu escopo de classificação?	**Caso a imagem seja de um objeto não sendo classificado pelo app (p.ex. um copo num app de classificação de cachorros), o app apresenta como resultado a informação que se trata de um objeto fora do escopo de classificação deste app.** **N/A: caso o app visa classificar qualquer objeto**	**Sim, Não, N/A**
	15. O app mostra um aviso quando o sistema não é capaz de classificar uma foto com confiança suficiente?	Em casos em que não é possível classificar uma foto com confiança suficiente (p.ex < 70%), o app alerta ao usuário informando que não foi capaz de classificar esta foto (ao invés de mostrar resultados mesmo com grau de confiança baixa).	Sim, Não
	16. O app permite que o usuário solicite a verificação do resultado por especialistas humanos?	O app fornece um meio de entrar em contato com especialistas de domínio para verificar o resultado da classificação. **N/A: Caso o app não precise da segurança da validação humana.**	Sim, Não, N/A
Possibilitar coleta de *feedback* do usuário	17. O app permite que usuários com conhecimento no domínio do aplicativo possam enviar *feedback* referente ao resultado da classificação?	O app possibilita que usuários com conhecimento de domínio possam indicar se um resultado de classificação está correto ou não. **N/A: caso o app se direciona a um público alvo sem conhecimento de domínio**	Sim, Não, N/A
	18. O app proíbe usuários sem conhecimento no domínio do aplicativo enviem *feedback* referente ao resultado da classificação?	O app está sem a opção de *feedback* de corretude de resultado de classificação no caso em que seus usuários não poderiam fornecê-lo corretamente. **N/A: caso o app se direciona a um público alvo com conhecimento de domínio**	Sim, Não, N/A
	19. O app deixa claro o propósito de enviar *feedback*?	Caso o app forneça a possibilidade de *feedback*, o app permite que o usuário entenda como seu *feedback* pode afetar o funcionamento, sendo orientado a fornecê-lo com cuidado. **N/A: caso que o app não fornece a possibilidade de *feedback***	Sim, Não, N/A
Mitigar viés	20. O app está livre de viés?	**A interface do usuário está livre de reforço de viés social, preconceito ou uso de terminologia inapropriada.**	Sim, Não
Considerar os riscos ao usuário	21. O app indica os devidos cuidados para tirar fotos?	No caso em que a captura de fotos pode ser arriscada para o usuário (p.ex. tentando tirar foto de uma cobra peçonhenta) são apresentadas diretrizes para fazê-lo de um modo seguro e alertar o usuário sobre o perigo. **N/A: caso a captura de fotos pode ser feito sem perigo**	Sim, Não, N/A
	22. O app destaca os riscos envolvidos com um possível erro de classificação?	O app indica as consequências de um possível erro de classificação, especificamente em casos em que isso pode resultar em danos físicos a humanos. **N/A: caso que erros de classificação não trazem muitos riscos**	Sim, Não, N/A
	23. O app mostra elementos visuais de alerta caso o objeto classificado possa causar danos físicos a humanos?	O app mostra de forma visual (p.ex. usando cor e/ou ícone) para alertar o usuário sobre o perigo pelo objeto classificado. **N/A: caso o escopo do app não contêm objetos perigosos**	Sim, Não, N/A

Fonte: elaborado pelo autor

4.6 ANÁLISE ESTATÍSTICA

Com o objetivo de iniciar a avaliação de confiabilidade e validade da estrutura das heurísticas e *checklist* foi realizado uma análise estatística de caráter exploratório sobre a versão v0.2 deste. Seguindo a metodologia *Goal/Question/Metric* (CALDIERA, 1994), foram definidas as seguintes perguntas de análise:

Confiabilidade:

P1. Existe evidência de consistência interna do *checklist*?

Validade da estrutura:

P2. Existe evidência de validade convergente do *checklist*?

P3. Como fatores subjacentes influenciam as respostas dos itens do *checklist?*

Coleta de dados. Os dados utilizados nesta análise foram pontuações atribuídas a aplicativos inteligentes de classificação de imagens a partir de uma avaliação heurística utilizando a versão 0.2 do *checklist*. A amostra consistiu em 101 avaliações heurísticas em aplicativos da plataforma Google Play (2023) com classificação de imagens e disponíveis gratuitamente, selecionados de modo aleatório. As avaliações foram realizadas no mês de maio/2023 pelo autor e a orientadora do trabalho.

Uma característica percebida nos dados coletados foi a predominância de aplicativos com características similares, que se encaixam em sua grande maioria em diversos critérios na categoria de resposta “Não se aplica”. Em 8 itens nos dados coletados 70% ou mais das respostas eram “Não se aplica”. Esses aspectos e o tamanho da amostra ainda relativamente pequeno contribuem para que as análises executadas sejam em grande parte, apenas uma indicação dos resultados esperados com dados mais representativos e em maior número.

Preparação dos dados para análise. Para a execução das análises, os dados devem ser codificados numericamente. Como respostas do tipo “Não se aplica” representam a ausência daquele item, seria recomendável a remoção total dos “N/A”s, associando números apenas para “Sim” e “Não”. Porém, os métodos de análise falharam com essa codificação, retornando valores de erro.

Para contornar esse problema, foram executadas as análises com duas configurações alternativas:

- **Alternativa 1:** Codificação apenas de “Sim” e “Não”**,** com exclusão de “N/A”. Itens com mais de 90% de respostas “N/A” foram excluídos da análise, resultando em apenas os itens da Tabela 15.
- **Alternativa 2:** Codificação de todas as respostas.

Tabela 15 - Heurísticas e itens restantes na alternativa 1 de análise

Heurística	**Itens do *checklist***
Deixar as expectativas e limitações explícitas	c1, c2, c3
Apoiar o uso efetivo	c4, c5
Apoiar a compreensão do usuário sobre incerteza e a confiança do modelo	c6, c10
Assegurar privacidade e segurança de dados	c11
Falhar graciosamente e suportar a recuperação de erros	c12, c13, c14, c15
Mitigar viés	c20

Fonte: elaborado pelo autor

4.6.1 Metodologia da análise

Análise de consistência interna do *checklist*

A consistência interna do *checklist* foi medida a partir do coeficiente alfa de Cronbach (1951) e o coeficiente ômega de Mcdonald (2013).

O alfa de Cronbach mede o quanto um conjunto de itens influencia indiretamente em um fator de qualidade (DEVELLIS, 2021), que no caso do *checklist* proposto é a avaliação de AIX. Esse método possui algumas suposições fortes sobre os dados, esperando que haja variância igual entre eles, que tenham distribuição normal, e explicam de forma igual um único fator que influencia nas respostas (MCNEISH, 2018).

Como alternativa ao alfa de Cronbach, optou-se pelo uso de ômega. A principal diferença entre os métodos é que o coeficiente ômega realiza uma análise fatorial, não assumindo que existe um único fator e que todos os itens o explicam igualmente (MCNEISH, 2018). Existem algumas variações do ômega, neste trabalho foi escolhido o ômega total, que será equivalente ao alfa de Cronbach caso os dados respeitem todas as suposições (MCNEISH, 2018).

O limite considerado aceitável para se dizer que existe consistência interna com alfa e ômega é de 0,7 a 0,95 (DEVELLIS, 2021; MCNEISH, 2018).

Análise de validade convergente do *checklist*

Para determinar se as dimensões (heurísticas) estão bem definidas no *checklist, análises* de correlação foram realizadas analisando a correlação entre itens (correlação policórica) e correlação item total. Com a correlação policórica, é possível verificar a validade convergente observando se itens que pertencem à mesma dimensão possuem correlação alta entre si (WOHLIN et al., 2012), o que neste tipo de análise pode ser definida como maior que 0,29 (COHEN, 2013). Já com a correlação item total, pode-se determinar a consistência entre heurísticas. Os itens com correlações baixas afetam negativamente a validade do *checklist* e espera-se nessa análise que cada item tenha uma correlação média ou alta com todos os outros (DEVELLIS, 2021). O critério utilizado como limite para se considerar boa correlação é o mesmo da última análise, o valor de 0,29 sugerido por Cohen (2013). Essa métrica é apresentada junto a uma análise do alfa de Cronbach com a remoção do item, em que se espera que não exista aumento significativo do valor obtido (WOHLIN et al., 2012).

Como fatores subjacentes influenciam as respostas dos itens do *checklist?*

Uma análise fatorial foi executada para determinar o número de fatores que influenciam nas respostas dos itens. Idealmente, o valor encontrado deveria ser igual ao número de heurísticas, que acredita-se ser a quantidade de diferentes aspectos de AIX avaliados pelo *checklist*.

Inicialmente, a possibilidade de executar uma análise fatorial deve ser verificada. Isso é feito a partir do índice Kaiser-Meyer-Olkin, ou índice KMO (BROWN, 2015). A adequação da amostra é medida entre 0 e 1, com valores maiores que 0,5 sendo considerados suficientes para justificar uma análise fatorial (BROWN, 2015).

O número de fatores utilizados é determinado por uma análise de fatores (GLORFELD, 1995), e para isso optou-se pela análise paralela, onde o número

de fatores é definido a partir do cálculo que valores eigen, que podem ser significantes somente se maiores que 1.

Para analisar a carga de cada item nos fatores, é utilizado o método de rotação Oblimin (JACKSON, 2014). Nesse é definido que a carga de cada item, para ser considerado aceitável naquele fator, deve ser maior que 0,55 (COMREY e LEE, 2013). Foram executadas duas análises fatoriais em cada alternativa de análise, uma considerando o número de fatores indicado pela análise paralela, e outra com apenas um fator, que apresenta semântica de quanto AIX em geral influencia nas respostas.

4.6.2 Resultados da Alternativa de Análise 1

Existe evidência de consistência interna do *checklist*?

O valor de alfa de cronbach foi 0,63 e o de ômega total foi 0,72. Indicando que existe uma consistência interna mínima considerando que os dados explicam fatores diferentes de AIX.

Existe evidência de validade convergente do *checklist*?

A correlação policórica obtida pode ser visualizada na Tabela 16, onde correlações entre itens da mesma heurística estão destacados. Em geral, itens da mesma heurística apresentaram boa intercorrelação, indicando possível validade convergente, com apenas os pares c1-c2 e c12-c14 não atingindo o limite de Cohen (2013) de 0,29.

Entretanto, a correlação item total (Tabela 17) revelou que talvez não exista muita consistência entre os itens, já que apenas 5 itens atingiram o valor esperado (> 0,29). Os dois itens que apresentaram pior correlação também resultam em um aumento da consistência interna medida pelo alfa de Cronbach se removidos.

Tabela 16 - Correlação policórica da Análise 1

Heurística	Item	c1	c2	c3	c4	c5	c6	c10	c11	c12	c13	c14	c15	c20
Deixar as expectativas e limitações explícitas	c1	1												
	c2	0,183	1											
	c3	0,376	0,618	1										
Apoiar o uso efetivo	c4	0,225	0,147	0,305	1									
	c5	0,393	-0,017	0,253	0,683	1								
Apoiar a compreensão do usuário sobre incerteza e a confiança do modelo	c6	0,071	-0,071	-0,033	0,408	0,198	1							
	c10	0,17	0,623	0,548	0,367	0,047	0,639	1						
Assegurar privacidade e segurança de dados	c11	0,203	-0,025	-0,003	0,194	0,31	0,235	0,195	1					
Falhar graciosamente e suportar a recuperação de erros	c12	-0,166	0,209	-0,144	0,051	-0,149	-0,06	-0,005	-0,095	1				
	c13	0,052	0,302	0,102	0,609	0,696	0,262	0,232	0,16	0,505	1			
	c14	0,385	0,56	0,732	0,447	0,493	0,099	0,367	0,168	0,243	0,541	1		
	c15	0,148	0,432	0,056	0,3	0,406	0,077	0,038	0,199	0,316	0,63	0,642	1	
Mitigar viés	c20	0,17	0,651	0,209	0,21	-0,051	0,271	0,672	0,12	-0,282	0,023	-0,019	0,112	1

Fonte: elaborado pelo autor

Tabela 17 - Correlação item total na Análise 1

Item	Item total	Alfa se removido
c1	0.23373999	0.6307088
c2	0.21943769	0.6328931
c3	0.15758682	0.6399023
c4	0.48364787	0.5780064
c5	0.36784418	0.6054222
c6	0.19727261	0.6387944
c10	0.21943769	0.6328931
c11	0.19120903	0.6374405
c12	0.05975197	0.6666996
c13	0.51567460	0.5795284
c14	0.51804996	0.5767847
c15	0.40957496	0.5971202
c20	0.04940678	0.6470609

Fonte: elaborado pelo autor

Como fatores subjacentes influenciam as respostas dos itens do *checklist?*

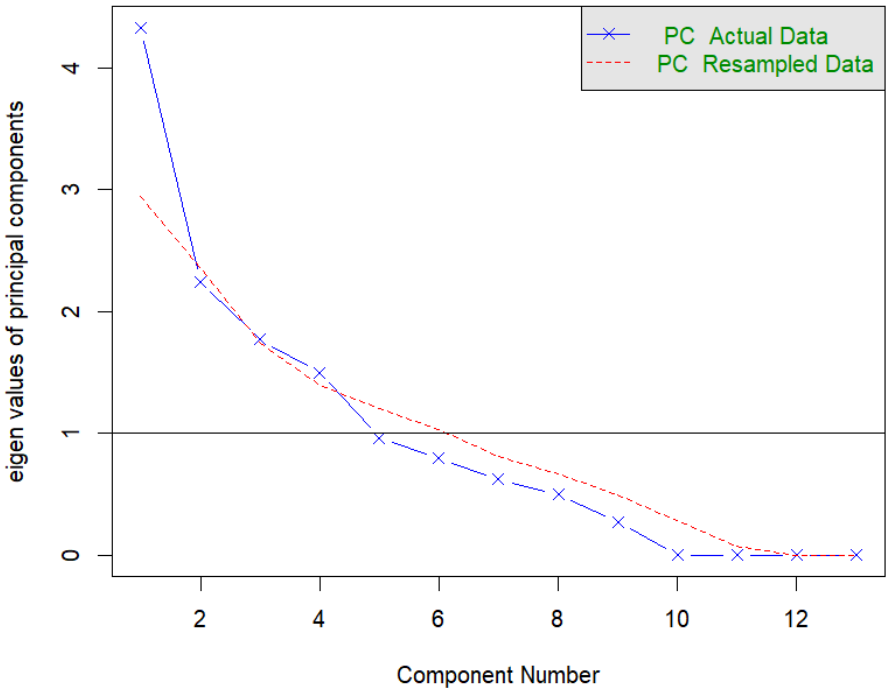
O número de dimensões encontradas com a análise de matrizes paralelas aleatórias é determinado a partir da quantidade de Xs acima da linha pontilhada vermelha do gráfico na Figura 11. A análise sugere apenas 3 dimensões, ao passo que se esperavam 6 nesta alternativa, uma para cada heurística.

Sobre a análise fatorial com esse número de fatores, o valor de KMO na alternativa 1 foi observado em 0,63, permitindo a análise fatorial. A carga de cada item sobre os 3 fatores está na Tabela 18, com destaque dado a maior carga de cada item. Alguns itens não apresentaram carga suficiente em nenhum fator, porém entre os que tiveram em geral, percebe-se que a maioria foi agrupada junto a outros itens de suas heurísticas, com exceção aos itens c10 e c14. A distribuição dos itens sugere que os fatores que influenciam no nas respostas do *checklist* são:

- F1: Apoiar o uso efetivo
- F2: Falhar graciosamente e suportar a recuperação de erros
- F3: Explicar o sistema de modo compreensível.

A carga dos itens considerando que o único fator que influencia nas respostas é AIX pode ser vista na Tabela 19. Diversos itens não alcançaram o limite de 0,55 definido por Comrey e Lee (2013), potencialmente representando nesta alternativa que o *checklist* não é efetivo para avaliação de AIX.

Figura 11 - Dimensões encontradas com matrizes paralelas aleatórias na Análise 1



Fonte: elaborado pelo autor

Tabela 18 - Análise fatorial com 3 dimensões da Alternativa de análise 1

Heurística	Item	F1	F2	F3
Deixar as expectativas e	c1	0.2350	-0.1967	0.35623

limitações explícitas	c2	-0.1261	0.1719	1.00675
	c3	0.0162	-0.3079	0.97154
Apoiar o uso efetivo	c4	0.7936	0.0764	0.01346
	c5	0.9784	-0.1870	0.08854
Apoiar a compreensão do usuário sobre incerteza e a confiança do modelo	c6	0.4464	-0.0155	-0.21750
	c10	0.1146	-0.0527	0.69652
Assegurar privacidade e segurança de dados	c11	0.3522	-0.1032	-0.07886
Falhar graciosamente e suportar a recuperação de erros	c12	-0.2756	0.9744	0.00139
	c13	0.5869	0.7449	-0.06053
	c14	0.3764	0.3733	0.68674
	c15	0.3269	0.4825	0.33364
Mitigar viés	c20	0.1249	0.0201	0.08282

Fonte: elaborado pelo autor

Tabela 19 - Análise fatorial com 1 dimensão da Alternativa de análise 1

Item	F1
c1	0.327
c2	0.605
c3	0.373
c4	0.728
c5	0.840
c6	0.301
c10	0.525
c11	0.273
c12	0.161
c13	0.878
c14	0.783
c15	0.725
c20	0.131

Fonte: elaborado pelo autor

4.6.5 Resultados da Alternativa de Análise 2

Existe evidência de consistência interna do *checklist*?

Em comparação à Alternativa 1, o valor de alfa obtido foi menor, com apenas 0,57 e o ômega total obtido na Alternativa 2 foi maior, com 0,77. Os valores obtidos sugerem uma consistência interna mínima, e assim como encontrado na alternativa 1, que depende de fatores além de simplesmente AIX.

Existe evidência de validade convergente do *checklist*?

A correlação policórica (Tabela 20) apresentou diversas intercorrelações abaixo do limite de 0,29, e valores negativos, que sugerem que um item não é pontuado caso o par seja. A maior intercorrelação negativa observada foi entre o par c17-c18, o que é parcialmente explicado pelo caráter complementar entre

eles, uma explicação que não pode ser estendida às outras ocorrências. A heurística *Considerar os riscos ao usuário* foi a única nesta análise (considerando as com mais de um item) com todos os seus itens intercorrelacionados.

Os valores resultantes da correlação item total estão disponíveis na Tabela 21. De modo similar à Alternativa 1, é visível a falta de itens cuja correlação item total atinge o limite. Algumas correlações item total negativas foram observadas nesse caso, e todas resultam em um aumento no alfa de Cronbach quando removidas.

Tabela 20 - Correlação policórica da Análise 2

Heurística	Item	c1	c2	c3	c4	c5	c6	c7	c8	c9	c10	c11	c12	c13	c14	c15	c16	c17	c18	c19	c20	c21	c22	c23
Deixar as expectativas e limitações explícitas	c1	1																						
	c2	0,187	1																					
	c3	0,428	0,122	1																				
Apoiar o uso efetivo	c4	0,223	0,131	0,383	1																			
	c5	0,398	-0,018	0,356	0,676	1																		
Apoiar a compreensão do usuário sobre incerteza e a confiança do modelo	c6	0,085	-0,105	-0,135	0,396	0,205	1																	
	c7	0,095	-0,171	-0,125	0,346	0,233	0,896	1																
	c8	-0,253	0,244	-0,137	0,056	-0,12	-0,019	0,01	1															
	c9	-0,098	0,205	-0,112	0,074	-0,06	-0,492	-0,423	0,585	1														
	c10	0,167	0,683	0,123	0,342	0,018	0,316	0,082	0,408	0,211	1													
Assegurar privacidade e segurança de dados	c11	0,207	-0,023	-0,042	0,21	0,31	0,211	0,202	0,057	0,247	0,231	1												
Falhar graciosamente e suportar a recuperação de erros	c12	-0,167	0,292	-0,253	0,061	-0,156	-0,101	-0,065	0,197	0,015	0,075	-0,116	1											
	c13	0,056	0,354	0,03	0,628	0,471	0,27	0,237	0,314	0,101	0,306	0,166	0,528	1										
	c14	0,202	0,31	0,445	0,423	0,409	0,104	0,061	-0,238	-0,367	0,097	0,022	0,27	0,486	1									
	c15	0,155	0,467	0,016	0,284	0,34	0,112	0,169	0,219	0,062	0,065	0,188	0,267	0,668	0,524	1								
	c16	-0,381	-0,199	-0,17	-0,031	-0,008	0,097	0,091	0,312	-0,167	-0,025	-0,031	0,102	-0,123	-0,186	-0,209	1							
Possibilitar coleta de feedback do usuário	c17	-0,146	0,118	-0,378	-0,021	-0,092	0,182	0,22	0,547	0,406	0,105	0,237	0,172	0,361	-0,284	0,482	-0,098	1						
	c18	0,155	0,145	0,438	-0,049	-0,088	-0,181	-0,141	-0,24	-0,28	0,037	-0,396	-0,195	-0,424	0,117	-0,329	0,134	-0,795	1					
	c19	-0,016	0,092	0,193	0,475	0,354	0,203	0,183	0,222	0,137	0,284	0,272	0,039	0,503	0,243	0,127	0,027	0,29	-0,496	1				
Mitigar viés	c20	0,167	-0,178	0,067	0,244	0,018	0,218	0,214	-0,324	-0,216	-0,144	0,106	-0,338	-0,052	0,129	0,035	0,03	-0,354	0,295	-0,071	1			
Considerar os riscos ao usuário	c21	-0,268	0,062	-0,043	0,134	0,176	-0,093	-0,071	0,238	0,073	0,09	-0,071	0,134	-0,115	-0,194	-0,195	0,711	0,007	0,085	0,159	-0,368	1		
	c22	-0,246	-0,093	-0,003	0,028	0,172	0,135	0,177	0,224	-0,099	0,072	0,112	-0,048	-0,1	-0,18	-0,181	0,804	0,005	-0,014	0,196	-0,154	0,651	1	
	c23	-0,13	-0,012	-0,098	0,018	0,114	0,048	0,009	0,384	-0,006	0,208	-0,079	0,127	-0,069	-0,154	-0,247	0,816	-0,017	0,047	0,17	-0,259	0,729	0,789	1

Fonte: elaborado pelo autor

Tabela 21 - Correlação item total na Análise 2

Item	Item total	Alfa se removido
c1	0.05565379	0.5747479
c2	0.05263879	0.5694380
c3	0.11631509	0.5651957
c4	0.44632796	0.5167661
c5	0.34768056	0.5362787
c6	0.37460117	0.5283063
c7	0.31135295	0.5318412
c8	0.15454090	0.5609893
c9	-0.14597273	0.5805173
c10	0.17721849	0.5623549
c11	0.12040791	0.5654487
c12	0.05437534	0.5759331
c13	0.37375570	0.5343351
c14	0.27752223	0.5436790
c15	0.25068617	0.5486008
c16	0.31949157	0.5390443
c17	-0.03221939	0.5846081
c18	-0.21149062	0.6434696
c19	0.23340125	0.5507009
c20	0.03359204	0.5707737
c21	0.22530108	0.5514885
c22	0.28844915	0.5386478
c23	0.31912148	0.5364747

Fonte: elaborado pelo autor

Como fatores subjacentes influenciam as respostas dos itens do *checklist?*

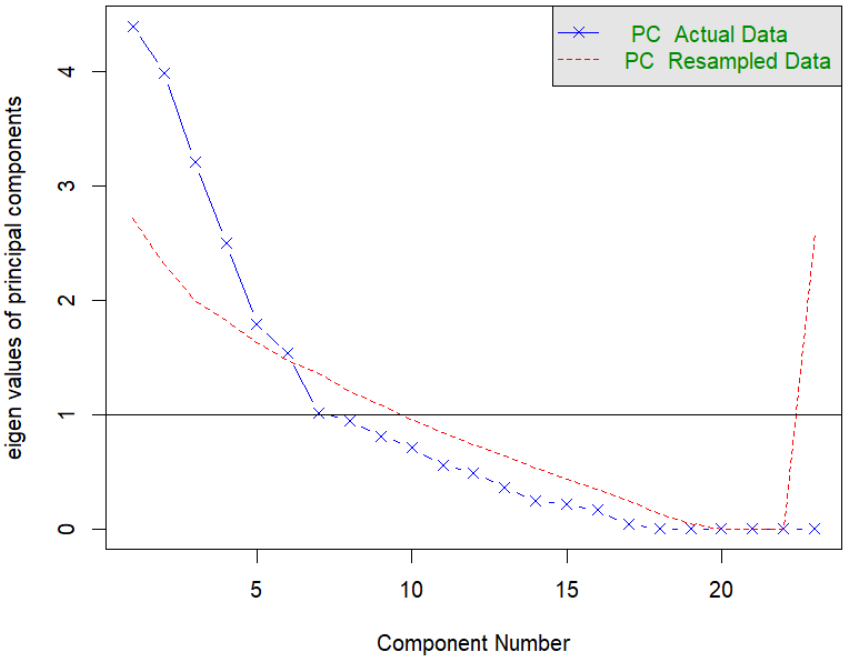
A quantidade de dimensões sugeridas para a análise fatorial (Figura 12) na Alternativa 2 foi 6. Assim como no caso anterior, um valor menor do que as 8 heurísticas esperadas.

Para a análise fatorial, O KMO obtido foi de 0,58, permitindo que ela fosse realizada. Observou-se nesta alternativa de análise uma quantidade considerável de itens que não se encaixaram em nenhum dos 6 fatores, com heurísticas inteiras não representadas. Além disso, em contrapartida à alternativa 1, não é possível visualizar explicações para os fatores a partir da distribuição dos itens.

Considerando apenas 1 fator, pode-se observar (Tabela 23) que, analogamente ao resultado da Alternativa 1, existe uma possibilidade do *checklist* não ser efetivo para avaliar AIX. A presença de itens com cargas

negativas no único fator pode indicar a necessidade de considerar a redução do *checklist*.

Figura 12 - Dimensões encontradas com matrizes paralelas aleatórias na Análise 2



Fonte: elaborado pelo autor

Tabela 22 - Análise fatorial com 6 dimensões da Alternativa de análise 2

Heurística	Item	F1	F2	F3	F4	F5	F6
Deixar as expectativas e limitações explícitas	c1	0.17543	0.02942	-0.74270	0.0415	0.00316	0.07257
	c2	0.15817	-0.07688	0.01130	-0.1062	-0.07794	0.74173
	c3	0.24882	-0.65712	-0.46329	0.1091	0.17077	-0.15480
Apoiar o uso efetivo	c4	0.18603	0.62861	-0.10736	-0.1791	0.14181	-0.10431
	c5	0.10086	0.00467	0.11059	-0.0284	0.31105	-0.55700
Apoiar a compreensão do usuário sobre incerteza e a confiança do modelo	c6	-0.02640	-0.07933	-0.07001	0.8181	0.02222	0.04593
	c7	0.00902	-0.12780	-0.09123	0.7732	0.01132	-0.05938
	c8	-0.08911	-0.09120	0.63327	-0.0669	0.42313	-0.04942
	c9	-0.17674	0.27251	0.10493	-0.5358	0.34055	-0.00323
	c10	-0.04039	0.04036	-0.02701	0.0693	0.17373	0.84851
Assegurar privacidade e segurança de dados	c11	0.04338	0.70813	-0.10450	-0.0371	0.24049	-0.09788
Falhar graciosamente e suportar a recuperação de erros	c12	0.12355	-0.10483	0.85289	-0.1358	-0.07200	-0.00411
	c13	-0.02091	-0.11906	0.72592	-0.1245	0.10836	-0.14025
	c14	0.61672	0.10924	-0.10333	0.0861	-0.09917	-0.04184
	c15	0.35708	0.16949	0.49771	0.2008	0.21950	-0.25528
	c16	-0.09820	-0.70826	0.11948	0.1388	0.01449	0.12189
Possibilitar coleta de feedback do usuário	c17	0.33249	0.27712	0.00867	0.3517	0.19849	-0.21415
	c18	-0.18332	-0.52232	-0.23465	-0.1283	-0.14738	0.30914
	c19	0.29654	0.07908	-0.66648	0.1024	0.02238	0.01953
Mitigar viés	c20	-0.02351	0.77009	-0.19113	-0.1010	-0.06388	-0.01011
Considerar os riscos ao usuário	c21	-0.13557	-0.73572	0.14377	0.1844	0.02597	-0.03975
	c22	-0.29630	-0.27814	0.20398	-0.1739	-0.07536	0.02983

	c23	-0.29729	-0.20184	0.13441	-0.2291	-0.10763	0.06816

Tabela 23 - Análise fatorial com 1 dimensão da Alternativa de análise 2

Item	F1
c1	0.2672
c2	0.7081
c3	0.2270
c4	0.3162
c5	0.1889
c6	0.0622
c7	0.0504
c8	-0.3609
c9	0.1596
c10	0.4109
c11	0.1698
c12	0.0932
c13	0.5566
c14	0.5098
c15	0.5335
c16	-0.9920
c17	0.1423
c18	-0.1172
c19	0.2306
c20	0.2800
c21	-0.9114
c22	-0.9133
c23	-0.9697

Fonte: elaborado pelo autor

4.6.4 Discussão

Os resultados obtidos em ambas as análises tendem a sugerir que o *checklist* possui uma consistência interna suficiente, considerando os coeficientes de ômega total obtidos.

Na questão de validade convergente, a Alternativa 1 sugere que os itens estão bem agrupados dentro de suas heurísticas. Entretanto, quando todos os itens são preservados na Alternativa 2, a correlação entre os itens enfraquece significativamente. Em ambos os casos foi identificada uma grande ausência de correlação entre todos os itens, indicando a necessidade de revisar ou até mesmo remover itens do *checklist*.

A análise dos fatores subjacentes em geral é inconclusiva. Em nenhum dos casos o número de fatores se igualou a número de heurísticas, e apesar de que era possível ver semântica nos fatores identificados na Alternativa 1, o mesmo não pode ser dito na Alternativa 2. Também não é possível afirmar que o conjunto de todos os itens é influenciado em igual por um fator único de AIX.

Como as análises realizadas foram prejudicadas pelo tamanho da amostra e a quantidade grande de itens com “N/A”, sugere-se ainda não propor modificações na versão 0.2 do checklist.

Ameaças à Validade. Para reduzir o impacto de ameaças relacionadas a esta análise, foi utilizado uma metodologia sistemática (CALDIERA, 1994). A qualidade dos dados em termos de padronização das respostas é um risco que foi mitigado pela aplicação cuidadosa do *checklist*, definindo uma resposta padrão em casos ambíguos. Outro risco se refere ao agrupamento de dados de diferentes contextos, que não foi considerado problemático neste trabalho pois o *checklist* deveria ser aplicável em um contexto genérico de classificação de imagens. Uma outra ameaça à validade externa consta no tamanho e diversidade de amostra, que não foi mitigada neste trabalho devido a uma falta generalizada de aplicativos de classificação de imagens na plataforma de distribuição Google Play, algo que foi considerado na execução e discussão das análises. Um outro risco é o quanto os dados dependem de pesquisadores específicos, que foi mitigado especificando uma metodologia sistemática, que define o propósito do estudo, o método de coleta de dados e os métodos estatísticos da análise. Para mitigar o risco de escolha correta de métodos estatísticos, foi seguido a proposta de DeVellis (2021) de construir escalas de medição, que são alinhadas com procedimentos de avaliação de consistência interna e validade da estrutura dos instrumentos de medição (TROCHIM e DONNELLY, 2001).

Porém observa-se que o tamanho da amostra e a quantidade grande de respostas “N/A” impactaram significativamente as análises, impedindo uma análise significativa da validade do *checklist*. Desta forma os resultados apresentados podem ser considerados somente como uma análise bem inicial e devem ser repetidas com maior amostra e/ou mais aplicativos que de forma mais completa atendem às heurísticas e/ou completados por outros tipos de estudos para avaliar a validade.

4.7 FERRAMENTA ONLINE DE SUPORTE DE AVALIAÇÃO HEURÍSTICA

Com o objetivo de oferecer um suporte para executar uma avaliação heurística utilizando o *checklist* proposto, uma ferramenta *web* foi desenvolvida. O sistema foi implementado utilizando tecnologias *web* (HTML, CSS e Javascript), e idealizado como um “quiz”, em que cada item do *checklist* é respondido como uma pergunta e ao final o resultado é apresentado.

Como parte do resultado é apresentado uma lista das respostas e o percentual dos itens satisfeitos, calculada a partir da seguinte fórmula, onde n é a quantidade de respostas “Sim” e m a quantidade de respostas “Não”:

$$n * 100 / (n + m) \quad (1)$$

4.7.1 Requisitos

Os requisitos da ferramenta são divididos em requisitos funcionais e não funcionais:

Requisitos Funcionais:

- **RF1 - Responder o checklist:** O sistema deve apresentar cada item do checklist (incluindo: nome, explicação e imagem de exemplo) e a respectiva escala de resposta, permitindo responder cada item. As respostas devem ser armazenadas pelo sistema.
- **RF2 - Calcular o percentual dos itens satisfeitos**: Ao submeter a resposta para o último item do checklist, deve ser calculado o percentual de respostas “Sim” em relação à quantidade total de respostas “Sim” e “Não”.
- **RF3 - Visualizar resultado**: Ao submeter a resposta para o último item do *checklist*, deve ser apresentado uma lista dos itens, indicando de forma visual a resposta (Sim-verde, Não - vermelho, NA-cinza junto com ícones indicativos) e o percentual calculada.
- **RF4 - Exportar PDF:** A apresentação dos resultados deve ser exportável para o formato PDF.

Requisitos Não Funcionais:

- **RNF1 - Idiomas:** O sistema deve estar disponível em inglês e português brasileiro.
- **RNF2 -** O sistema deve ter uma interface estreita para possibilitar a sua visualização em tela dividida com um outra aba do navegador aberta no App Inventor.
- **RNF3 - Responsividade *Mobile*:** O sistema deve poder ser utilizável nos navegadores de telefones celulares sem a necessidade de *scroll* horizontal. Botões devem ser grandes o suficiente para serem utilizáveis por toque.

4.7.2 Implementação

O sistema foi implementado como uma única página *web*, em que a navegação entre as diferentes telas se dá pela mudança de qual componente está atualmente visível, com um *script* principal orquestrando a visibilidade dos componentes e mantendo o estado das variáveis usadas na avaliação.

A estrutura das telas de pergunta foi especificada em um único ponto, contendo todos os elementos gráficos comuns destas (Figura 14), o que permite que perguntas sejam instanciadas dinamicamente a partir de um modelo e inseridas na página web após alterar seus textos e imagens. A especificação de uma pergunta se dá pela instanciação da classe *ItemChecklist,* que preenche os campos desse modelo (Figura 15 e FIgura 16) a partir de uma estrutura de dados e disponibiliza um fragmento HTML (MOZILLA, 2023) que pode ser inserido na página.

A navegação entre perguntas e o armazenamento de respostas foi feito a partir do padrão *Observer* (GAMMA et al., 1995), em que o *script* principal é um receptor de eventos de submissão de respostas de cada *ItemChecklist*, e reage armazenando a resposta e incrementando um índice que indica qual pergunta é visível. Ao projetar desse modo, cada *ItemChecklist* opera apenas sobre o fragmento HTML que representa, sendo independente de qualquer estrutura que o utiliza.

Quando o último resultado é armazenado, o *script* navega a uma tela de resultados (Figura 17) que realiza o RF2. Nesta, a resposta de cada pergunta é apresentada com um esquema de cores (Verde - “Sim”, Vermelho - “Não” e

Cinza - “N/A” e ícones indicativos) e é apresentado o resultado da avaliação, indicando o percentual dos itens satisfeitos, de acordo com o pseudocódigo apresentado no Quadro 1.

As informações de resultado podem ser exportadas para PDF (Figura 18) com um botão na tela de resultados. A implementação dessa funcionalidade foi feita a partir do uso da biblioteca jsPDF (PARALLAX, 2023).

Quadro 1 - Algoritmo para cálculo da nota

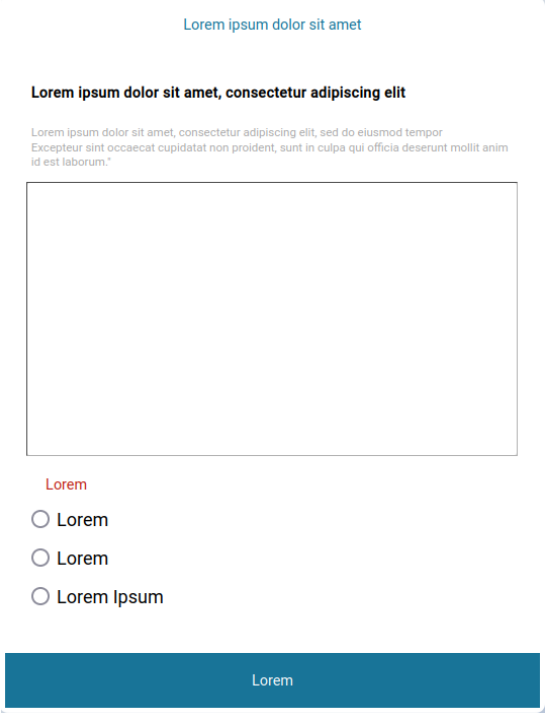
```
INÍCIO

VARIÁVEIS
soma,respondidas,nota:Real; respostas: String[]

PARA i DE 0 ATÉ TAMANHO DE respostas FAÇA:
  SE respostas ACESSADO EM i IGUAL A “Sim” ENTÃO
    soma <- soma + 1
    respondidas <- respondidas + 1
  SENÃO SE respostas ACESSADO EM i IGUAL A “Não” ENTÃO
    respondidas <- respondidas + 1
  FIM SE
FIM PARA
nota <- (soma / respondidas) x 100

FIM
```

Fonte: elaborado pelo autor

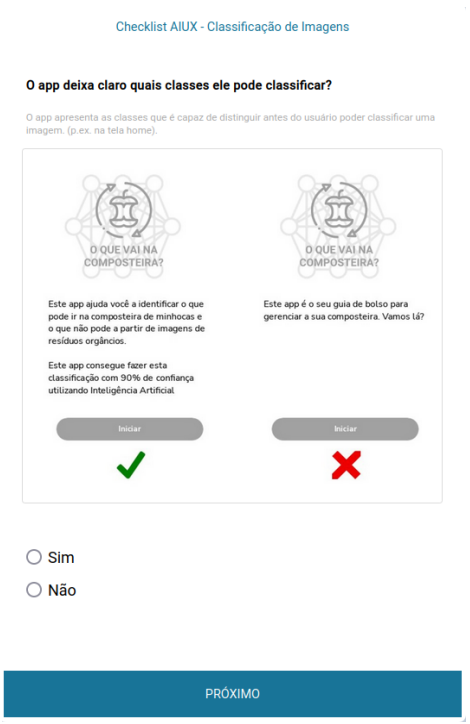
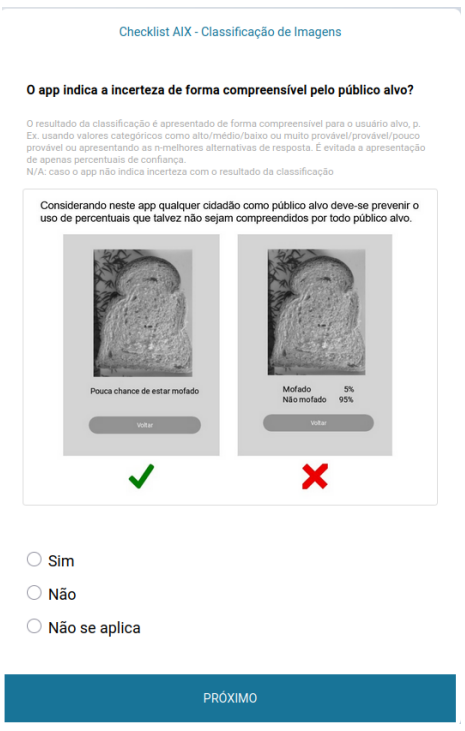
Figura 13 - Tela inicial da ferramenta JS

Fonte: elaborado pelo autor

Figura 14 - Estrutura de uma tela de pergunta

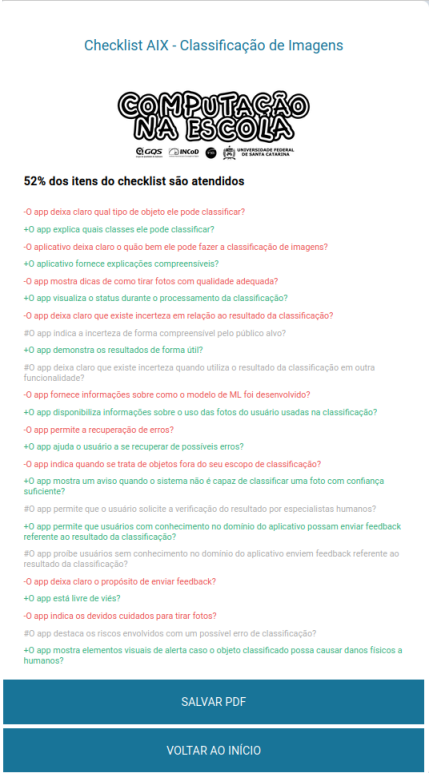


Fonte: elaborado pelo autor

Figura 15 - Tela do item 1 do *checklist,* preenchida a partir do modelo	Figura 16 - Tela do item 7 do *checklist,* preenchida a partir do modelo
	

Fonte: elaborado pelo autor

Figura 17 - Tela de respostas



Fonte: elaborado pelo autor

Figura 18 - PDF das respostas

Fonte: Elaborado pelo autor

Para dar suporte a mais de um idioma na ferramenta, optou-se por utilizar uma grande estrutura de dados onde a cada texto é associado um identificador (como TITLE para o texto do título) e existe uma replicação de todo texto para os idiomas suportados, com funções auxiliares para selecionar um texto a partir de seu identificador. Apesar da existência de bibliotecas de tradução (WANG et al., 2022), a tradução manual foi necessária neste caso para preservar a semântica dos itens do *checklist*.

As imagens de exemplo desenvolvidas para o *checklist* (em ambas as línguas para respeitar o RNF1) foram introduzidas na estrutura de tradução sob a forma de um *link* indexado pelo número do item de *checklist*, e uma função de acesso foi criada para facilitar a seleção da imagem correta de acordo com o idioma.

O design de interface foi projetado de forma estreita e com um design visual limpo. Para prover responsividade *mobile* é esperado que o sistema tenha que ser visualizado em telas de tamanhos diferentes. De acordo com Google (2023a), a largura das telas de celulares Android geralmente varia de 600dp a 920dp. A partir dessa informação, a estrutura das telas foi projetada para redimensionar dentro dessa faixa. No quesito de botões, foi garantido que a área de toque destes tem no mínimo 48px por 48px, assim como sugerido pelo Material Design 2 (GOOGLE, 2023b).

A ferramenta é disponibilizada de forma gratuita pelo site da iniciativa Computação na Escola/INCOD/INE/UFSC:

http://apps.computacaonaescola.ufsc.br/aix/

5. CONCLUSÃO

Neste trabalho, foi definido um conjunto de heurísticas e um checklist para avaliação de UX de apps com classificação de imagens. O trabalho foi embasado na teoria de classificação de imagens com *ML*, considerando tanto aspectos técnicos, como éticos. Após isso, foi sintetizada a teoria de usabilidade e de avaliações heurísticas, explicando assim os aspectos que compõem AIX (OE1). Para levantar o estado da arte de AIX, foi realizada uma revisão sistemática da literatura em que descobriu-se que existem poucas obras que definem princípios de usabilidade para IA, com nenhuma referindo-se especificamente ao contexto de classificação de imagens. Nenhum suporte a avaliação heurística para AIX foi encontrado (OE2).

Com base nos princípios de usabilidade encontrados, foi realizado um mapeamento em um conjunto de heurísticas voltadas ao contexto de aplicativos inteligentes de classificação de imagens. A partir destes, um *checklist* v0.1 composto de 20 itens foi especificado (OE3). Esta versão do checklist foi avaliada por um painel de especialistas. Como resultado identificou-se que alguns itens poderiam ser escritos de uma forma mais clara, alguns itens opcionais reconsiderados, e de que em geral o *checklist* precisava de explicações melhores para reduzir a dependência em imagens de exemplo. Essas mudanças foram realizadas e uma nova versão do *checklist* (v0.2) foi criada. Uma análise estatística foi realizada utilizando a nova versão, buscando analisar a confiabilidade e validade desta. Porém, pelo tamanho pequeno da amostra e a quantidade grande de respostas “N/A” as análises foram prejudicadas e consequentemente não foi possível inferir alterações para melhorar a qualidade do *checklist (OE4)*.

Para facilitar a utilização do checklist em uma avaliação heurística, foi desenvolvido também uma ferramenta *web*. O sistema guia a avaliação, apresentando cada um dos itens e ao final mostrando os resultados. O sistema também apresenta responsividade *mobile*, está disponível em português brasileiro e inglês e permite exportar um PDF com o resultado da avaliação heurística (OE5).

Desta forma a principal contribuição deste trabalho está na elaboração de um conjunto de heurísticas e *checklist* para a avaliação heurística de

aplicativos inteligentes de classificação de imagens. Mesmo já existindo outras propostas para AIX, a maioria atualmente é mais voltado para sistemas de recomendação e assim esta customização das heurísticas para classificação de imagens pode ser considerada inédita. Espera-se que a definição destas heurísticas possa ajudar tanto no design de aplicativos deste tipo quanto na sua avaliação, contribuindo desta maneira a melhoria da UX e assim a qualidade dos sistemas.

Como trabalhos futuros, sugere-se a realização de uma análise estatística com um conjunto de dados maior e mais variado, seguido do aperfeiçoamento das heurísticas e *checklist*. Alternativamente, recomenda-se o estudo de uma possível automação da avaliação utilizando o *checklist*.

REFERÊNCIAS

BIANCA, C. S A. **Desenvolvimento de um Curso Ensinando a Criação de Apps Inteligentes para a Classificação de Imagens com Machine Learning e Design Thinking**. 2022. Trabalho de Conclusão de Curso. (Graduação em Sistemas de Informação) – Universidade Federal de Santa Catarina.

AMERSHI, S. et al. **Guidelines for human-AI interaction**. Proceedings of the CHI Conference on Human Factors in Computing Systems. Glasgow, Scotland Uk, 2019. p. 1-13.

APPLE. **Human Interface Guidelines**. 2022. Disponível em <https://developer.apple.com/design/human-interface-guidelines/technologies/machine-learning/introduction/> Acesso em: 03/07/2022

BROWN, A. **Confirmatory factor analysis for applied research**. Guilford publications, 2015.

CALDIERA, V. R; BASILI, G; ROMBACH, H. D. **The goal question metric approach.** Encyclopedia of software engineering, p. 528-532, 1994.

COHEN, J. **Statistical power analysis for the behavioral sciences**. Routledge, 2013.

COMPUTAÇÃO NA ESCOLA. **Apps desenvolvidos por jovens no programa PodeCrer do Instituto Pe. Vilson Groh**. Disponível em <https://computacaonaescola.ufsc.br/appsivg2022/> Acesso em: 14/04/2023

COMREY, A L.; LEE, H. B. **A first course in factor analysis**. Psychology press, 2013.

CRONBACH, L. J. **Coefficient alpha and the internal structure of tests**. psychometrika, v. 16, n. 3, p. 297-334, 1951.

DAI, X; SPASIĆ, I; CHAPMAN, S; MEYER, B. **The state of the art in implementing machine learning for mobile apps: A survey.** 2020 SoutheastCon, p. 1-8, 2020.

DAI, X. et al. Machine learning on mobile: **An on-device inference app for skin cancer detection**. 2019 Fourth International Conference on Fog and Mobile Edge Computing . IEEE, 2019. p. 301-305.

DEISENROTH, M P; FAISAL, A. A; ONG, C. S. **Mathematics for machine learning**. Cambridge University Press, 2020.

DEVELLIS, R F.; THORPE, C T. **Scale development: Theory and applications**. Sage publications, 2021.

DUDLEY, J. J.; KRISTENSSON, P. O. **A review of user interface design for interactive machine learning.** ACM Transactions on Interactive Intelligent Systems, v. 8, n. 2, p. 1-37, 2018.

EUROPEAN COMMISION. **Ethics guidelines for trustworthy.** 2019. Disponível em: <https://digital-strategy.ec.europa.eu/en/library/ethics-guidelines-trustworthy-ai>. Acesso em: 26/11/2022

GALITZ, W. O. **Essential Guide to User Interface Design: An Introduction to GUI Design Principles and Techniques**. 2002.

GAMMA, E. et al. **Design patterns: elements of reusable object-oriented software**. Pearson Deutschland GmbH, 1995.

GLORFELD, L. W. **An improvement on Horn's parallel analysis methodology for selecting the correct number of factors to retain.** Educational and psychological measurement, v. 55, n. 3, p. 377-393, 1995.

GOOGLE. **People + AI Guidebook.** 2022a. Disponível em <https://pair.withgoogle.com/guidebook> Acesso em: 03/07/2022

GOOGLE. **Machine Learning - Google Developers**. 2022b. Disponível em <https://developers.google.com/machine-learning>. Acesso em 16/09/2022

GOOGLE. **Our Principles – Google AI.** 2022c. Disponível em: <https://ai.google/principles/>. Acesso em: 26/11/2022

GOOGLE. **Teachable Machine - Google.** 2022d. Disponível em: <https://teachablemachine.withgoogle.com/>. Acesso em: 03/12/2022

GOOGLE. **Suporte a tamanhos de tela diferentes.** 2023a. Disponível em: <https://developer.android.com/guide/topics/large-screens/support-different-screen-sizes?hl=pt-br>. Acesso em: 21/05/2023

GOOGLE. **Touch Target.** 2023b. Disponível em: <https://m2.material.io/develop/web/supporting/touch-target>. Acesso em: 21/05/2023

GOOGLE PLAY. **Google Lens**. 2023a. Disponível em: <https://play.google.com/store/apps/details?id=com.google.ar.lens&hl=pt_BR&gl=US>. Acesso em: 14/04/2023

GOOGLE PLAY. **Calorie Mama AI: Meal Planner**. 2023b. Disponível em: <https://play.google.com/store/apps/details?id=com.azumio.android.caloriesbuddy&hl=pt_BR&gl=US>. Acesso em: 14/04/2023

GOOGLE PLAY. **Picture Insect - Insetos ID**. 2023c. Disponível em: <https://play.google.com/store/apps/details?id=com.glority.pictureinsect&hl=pt_BR&gl=US>. Acesso em: 14/04/2023

GOOGLE PLAY. **Dog Identifier: Dog Scanner**. 2023d. Disponível em: <https://play.google.com/store/apps/details?id=com.differenz.dogidentifier&hl=pt_BR&gl=US>. Acesso em: 14/04/2023

GOOGLE PLAY. **Gemius: Rock Identifier - Ston**. 2023e. Disponível em: <https://play.google.com/store/apps/details?id=com.codeway.rockidentifier&hl=pt_BR&gl=US>. Acesso em: 14/04/2023

GUNDA, N. S. K; GAUTAM, S. H.; MITRA, S. K. **Artificial intelligence based mobile application for water quality monitoring**. Journal of The Electrochemical Society, v. 166, n. 9, p. B3031, 2019.

GUNNING, D. et al. **XAI—Explainable artificial intelligence**. Science robotics, v. 4, n. 37, p. eaay7120, 2019

GQS. **Design de aplicativos na disciplina INE5624-08208 Engenharia de Usabilidade (2022-2)**. 2023. Disponível em: <https://www.gqs.ufsc.br/design-de-aplicativos-na-disciplina-ine5624-08208-engenharia-de-usabilidade-2022-2/>. Acesso em: 01/07/2023.

HADDAWAY, N. R. et al. **The role of Google Scholar in evidence reviews and its applicability to grey literature searching**. PloS one, v. 10, n. 9, p. e0138237, 2015.

ISO/IEC. **ISO 9241-210:2010(en) Ergonomics of human-system interaction — Part 210: Human-centred design for interactive systems**. 2010. Disponível em: <https://www.iso.org/obp/ui/#iso:std:iso:9241:-210:ed-1:v1:en>. Acesso em: 24/09/2022

ISO/IEC. **ISO 9241-110:2020(en) Ergonomics of human-system interaction — Part 110: Interaction principles**. 2020. Disponível em: <https://www.iso.org/obp/ui/#iso:std:iso:9241:-210:ed-1:v1:en>. Acesso em: 25/09/2022

ISO/IEC. **ISO/IEC 25010:2011 Systems and software engineering — Systems and software Quality Requirements and Evaluation (SQuaRE) — System and software quality models.** 2011. Disponível em: <https://iso25000.com/index.php/en/iso-25000-standards/iso-25010>. Acesso em: 24/09/2022

JACKSON, J. E. **Oblimin Rotation.** Wiley StatsRef: Statistics Reference Online, 2014.

JURISTO, N; MORENO, A. M.; SANCHEZ-SEGURA, M. I.. **Analysing the impact of usability on software design**. Journal of Systems and Software, v. 80, n. 9, p. 1506-1516, 2007.

Karpathy, A.. **CS231n Convolutional neural networks for visual recognition**. 2022. Disponível em: <http://cs231n.github.io/classification/>. Acesso em: 09/09/2022

LARMAN, C. **Agile and iterative development: a manager's guide**. Addison-Wesley Professional, 2004.

LAWSHE, C. H. et al. **A quantitative approach to content validity**. Personnel psychology, v. 28, n. 4, p. 563-575, 1975.

LI, T; VORVOREANU, M; DEBELLIS, D. AMERSHI, S; **Assessing Human-AI Interaction Early through Factorial Surveys: A Study on the Guidelines for Human-AI Interaction**. ACM Transactions on Computer-Human Interaction, 2022.

MARTÍNEZ-FERNÁNDEZ, S; CASTANYER, R. C.; FRANCH, X. **Integration of convolutional neural networks in mobile applications**. Proceedings. of the IEEE/ACM 1st Workshop on AI Engineering-Software Engineering for AI. IEEE, 2021. p. 27-34.

MCDONALD, R. P. **Test theory: A unified treatment**. psychology press, 2013.

MCNEISH, D. **Thanks coefficient alpha, we'll take it from here.** Psychological methods, v. 23, n. 3, p. 412, 2018.

MICROSOFT. **Guidelines for Human-AI Interaction.** 2019; Disponível em: <https://www.microsoft.com/en-us/research/project/guidelines-for-human-ai-interaction/>. Acesso em: 02/10/2022

MIT. **MIT App Inventor**. 2022. Disponível em: <https://appinventor.mit.edu/>. Acesso em: 03/12/2022

ML+Design. **MACHINE LEARNING + DESIGN.** 2022. Disponível em: <https://machinelearning.design/>. Acesso em: 08/10/2022

MOHSENI, S; ZAREI, N; RAGAN, E. D. **A multidisciplinary survey and framework for design and evaluation of explainable AI systems**. ACM Transactions on Interactive Intelligent Systems, v. 11, n. 3-4, p. 1-45, 2021.

MOZILLA. **DocumentFragment**. 2023. Disponível em: <https://developer.mozilla.org/pt-BR/docs/Web/API/DocumentFragment>. Acesso em: 21/05/2023

NIELSEN, J. **Enhancing the explanatory power of usability heuristics**. Proceedings of the SIGCHI conference on Human Factors in Computing Systems. Morristown: New Jersey, 1994. p. 152-158.

NIELSEN, J. **Ten usability heuristics**. 2005. Disponível em: <https://pdfs.semanticscholar.org/5f03/b251093aee730ab9772db2e1a8a7eb8522cb.pdf.>. Acesso em 09/09/2022.

NIELSEN, J; MOLICH, R. **Heuristic evaluation of user interfaces**. Proceedings of the SIGCHI conference on Human factors in computing systems. 1990. p. 249-256.

OLEJNIK, S; ALGINA, J. **Generalized eta and omega squared statistics: measures of effect size for some common research designs**. Psychological methods, v. 8, n. 4, p. 434, 2003.

OLIVEIRA, F. P; VON WANGENHEIM, C. G; HAUCK, J. C. R. **TMIC: App Inventor Extension for the Deployment of Image Classification Models Exported from Teachable Machine**. arXiv preprint arXiv:2208.12637, 2022.

PARALLAX. **jsPDF**. 2023. Disponível em: <https://parall.ax/products/jspdf>. Acesso em: 21/05/2023

PETERSEN, K; VAKKALANKA, Sairam; KUZNIARZ, Ludwik. **Guidelines for conducting systematic mapping studies in software engineering: An update**. Information and Software Technology, v. 64, p. 1-18, 2015.

RAWAT, W; WANG, Z.. **Deep convolutional neural networks for image classification: A comprehensive review**. Neural computation, v. 29, n. 9, p. 2352-2449, 2017.

REICHHELD, F. **The ultimate question 2.0 (revised and expanded edition): How net promoter companies thrive in a customer-driven world**. Harvard Business Review Press, 2011.

RUIZ, J; SERRAL, E; SNOECK, M. **Unifying functional user interface design principles**. International Journal of Human–Computer Interaction, v. 37, n. 1, p. 47-67, 2021.

RUSU, C. et al. **A Methodology to establish usability heuristics**. Research Gate, 2011. Disponível em <https://www.researchgate.net/publication/229040164_A_Methodology_to_establish_usability_heuristics> . Acesso em: 18/11/2022.

SILVA, E G. **Catálogos de características de qualidade de software: um mapeamento sistemático**. 2019. Disponível em <https://repositorio.ufc.br/bitstream/riufc/44567/1/2019_tcc_egsilva.pdf>. Acesso em: 12/02/2023

STAGGERS, N; NORCIO, A. F. **Mental models: concepts for human-computer interaction research**. International Journal of Man-machine Studies, v. 38, n. 4, p. 587-605, 1993.

SUBRAMONYAM, H; SEIFERT, C; ADAR, E. **Towards a process model for co-creating AI experiences**. Designing Interactive Systems Conference. 2021. p. 1529-1543.

TONG, S. **Detecting bias in image classification using model explanations**. 2020. Tese de Doutorado. Massachusetts Institute of Technology.

TROCHIM, W. M. K; DONNELLY, J. P. **Research methods knowledge base**. Macmillan Publishing Company, New York: Atomic Dog Pub., 2001.

VERMA, S. et al. **Deep learning-based mobile application for plant disease diagnosis: A proof of concept with a case study on tomato plant**. Applications of image processing and soft computing systems in agriculture. IGI global, 2019. p. 242-271.

WANG, P; FAN, E; WANG, P. **Comparative analysis of image classification algorithms based on traditional machine learning and deep learning**. Pattern Recognition Letters, v. 141, p. 61-67, 2021.

WANG, P; YOON, H. J. S; CHUNG, S. **A Comparison of Internationalization and Localization Solutions for Web and Mobile Applications**. Journal of Information Systems Applied Research, v.15, p. 39-46, 2022.

WASSERMAN, A. I. **Software engineering issues for mobile application development.** Proceedings of the FSE/SDP workshop on Future of software engineering research. ACM,, 397–400, 2010.

WOHLIN, C. et al. **Experimentation in software engineering**. Springer Science & Business Media, 2012.

ZHANG, C; CAI, Y; LIN, G; SHEN, C. **Deepemd: Few-shot image classification with differentiable earth mover's distance and structured classifiers.** Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2020. p. 12203-12213.

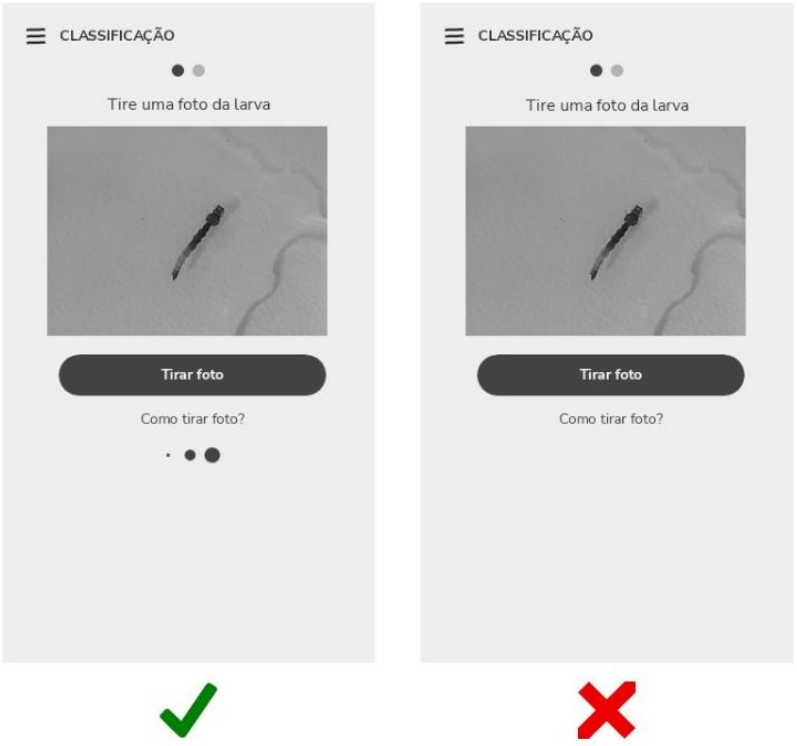
APÊNDICE A - Figuras de exemplo e contraexemplo de aplicação dos itens do *checklist* v0.2

Exemplo e contraexemplo dos itens *1. O app deixa claro quais classes ele pode classificar?* e *2. O aplicativo deixa claro o quão bem ele pode fazer a classificação de imagens?*

Exemplo e contraexemplo do item *3. O aplicativo fornece explicações compreensíveis?*

Exemplo do item *4. O app mostra dicas de como tirar fotos com qualidade adequada?*

Exemplo e contraexemplo do item *5. O app visualiza o status durante o processamento da classificação?*

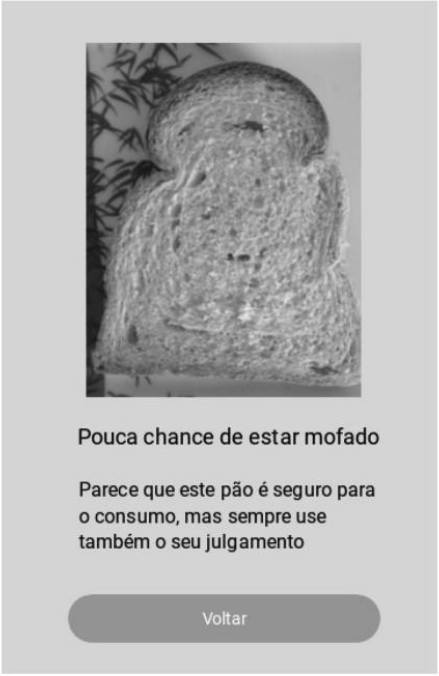


Exemplo e contraexemplo do item *6. O app deixa claro que existe incerteza em relação ao resultado da classificação?*

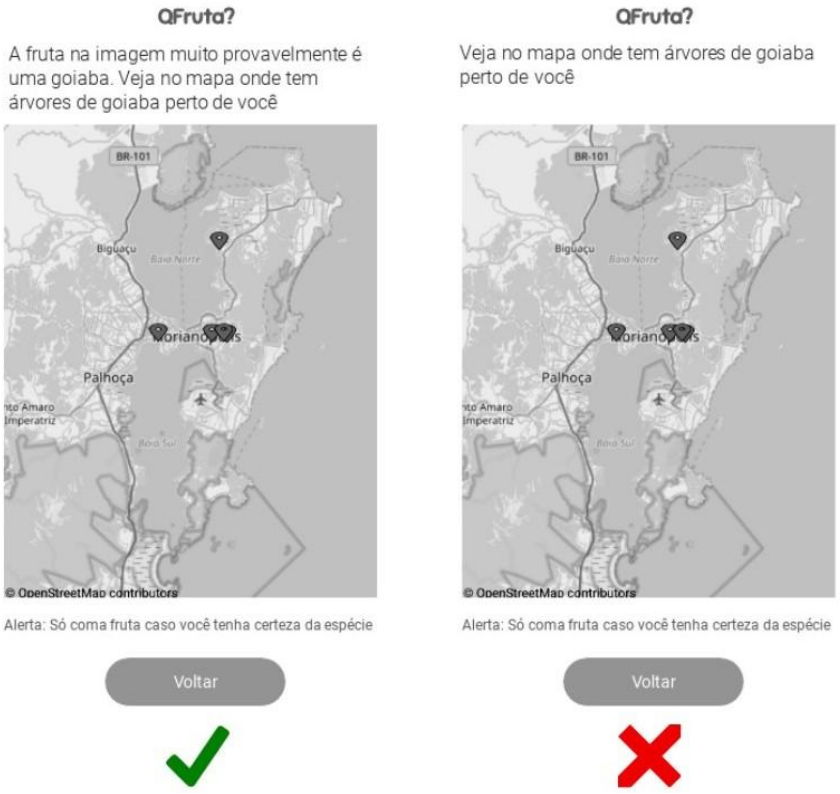
Exemplo e contraexemplo do item *7. O app indica a incerteza de forma compreensível pelo público alvo?*

Considerando neste app qualquer cidadão como público alvo deve-se prevenir o uso de percentuais que talvez não sejam compreendidos por todo público alvo.

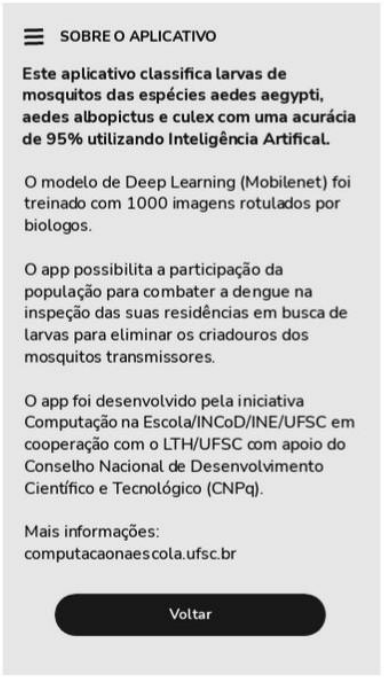
Exemplo do item *8. O app demonstra os resultados de forma útil?*



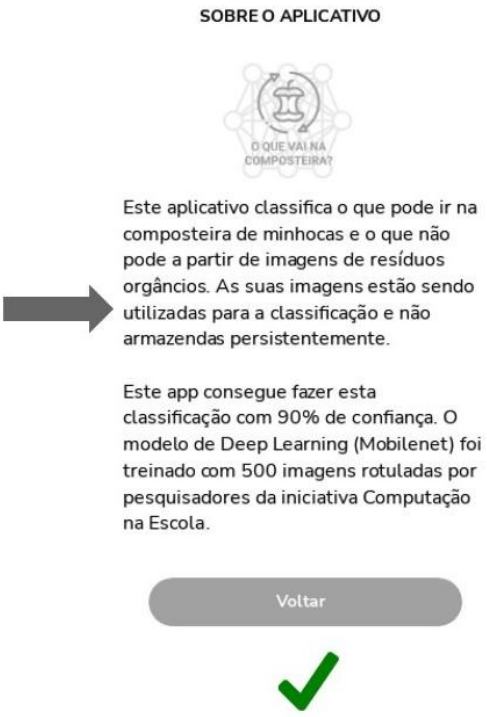
Exemplo e contraexemplo do item *9. O app deixa claro que existe incerteza quando utiliza o resultado da classificação em outra funcionalidade?*



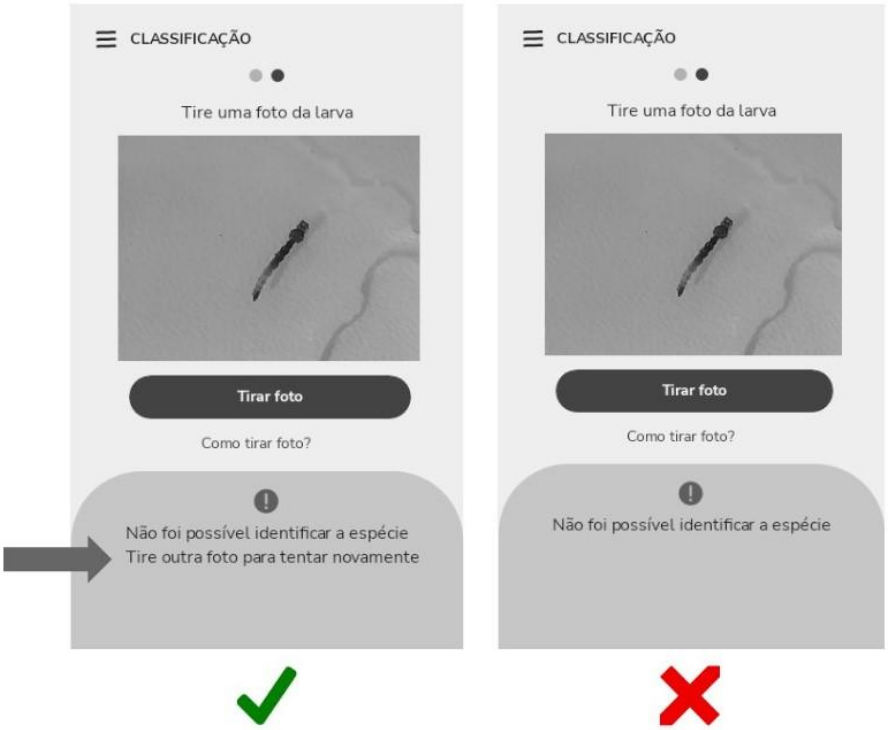
Exemplo do item *10. O app fornece informações sobre como o modelo de ML foi desenvolvido?*



Exemplo do item *11. O app disponibiliza informações sobre o uso das fotos do usuário usadas na classificação?*

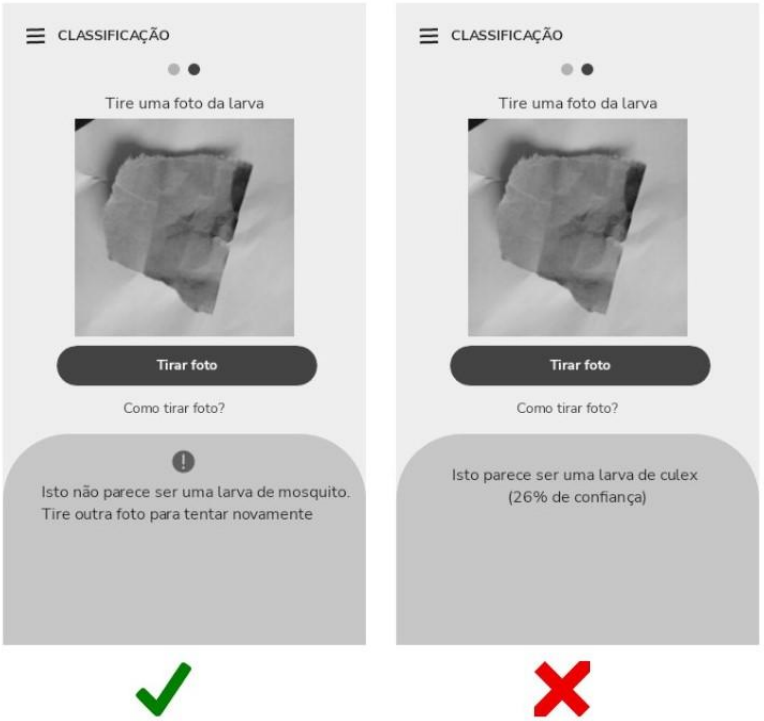


Exemplo e contraexemplo do item *12. O app permite a recuperação de erros?*

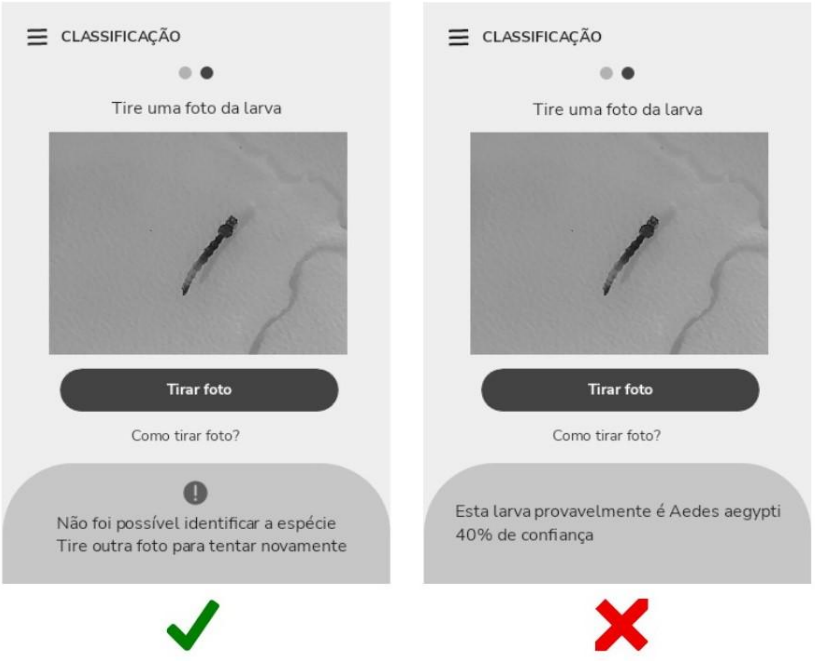


Exemplo do item *13. O app ajuda o usuário a se recuperar de possíveis erros?*

Exemplo e contraexemplo do item 14. *O app indica quando se trata de objetos fora do seu escopo de classificação?*



Exemplo e contraexemplo do item *15. O app mostra um aviso quando o sistema não é capaz de classificar uma foto com confiança suficiente?*



Exemplo do item *16. O app permite que o usuário solicite a verificação do resultado por especialistas humanos?*

Exemplo do item *17. O app permite que usuários com conhecimento no domínio do aplicativo possam enviar feedback referente ao resultado da classificação?*

App para classificação de fungos destinado a biólogos na área de micologia

Exemplo do item *18. O app proíbe usuários sem conhecimento no domínio do aplicativo enviem feedback referente ao resultado da classificação?*

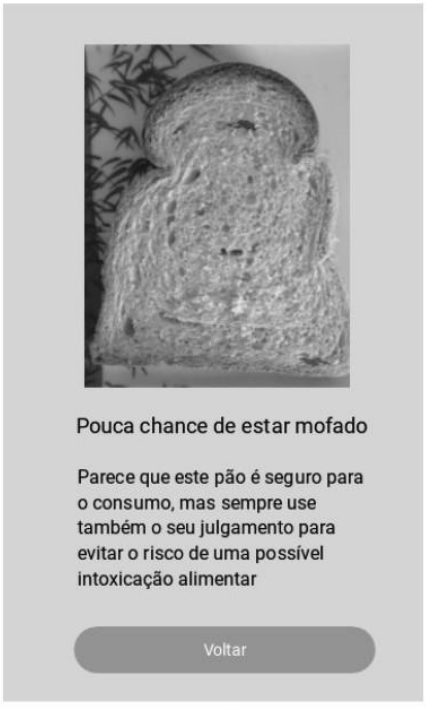
App para classificação de larvas de mosquitos destinado ao público geral sem conhecimento específico sobre o assunto

Sem possibilidade de mandar feedback

Exemplo do item *19. O app deixa claro o propósito de enviar feedback?*

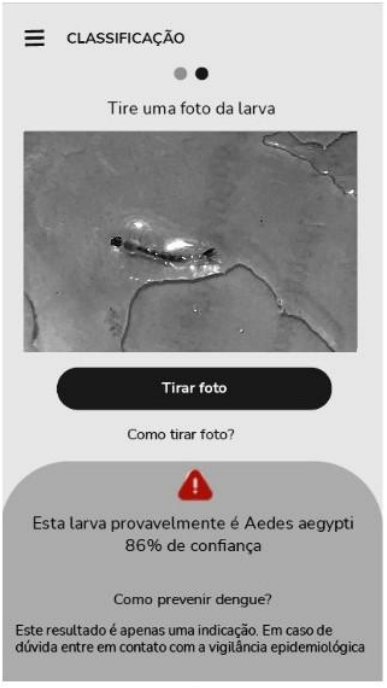
Exemplo e contraexemplo do item *20. O app está livre de viés?*

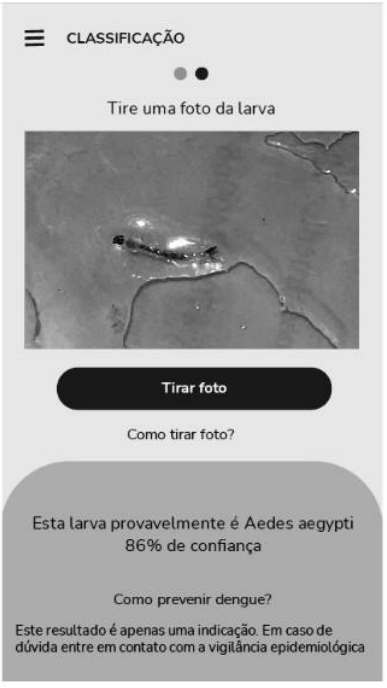
Exemplo do item *21. O app indica os devidos cuidados para tirar fotos?*



Exemplo do item *22. O app destaca os riscos envolvidos com um possível erro de classificação?*

Exemplo e contraexemplo do item *23. O app mostra elementos visuais de alerta caso o objeto classificado possa causar danos físicos a humanos?*





APÊNDICE B - Código fonte da ferramenta de suporte para avaliação heurística

Código fonte disponível em:
https://codigos.ufsc.br/gqs/aiux-checklist/-/tree/main

B1. src

B1.1. index.html

```
<head>
        <script
src="https://cdnjs.cloudflare.com/ajax/libs/html2canvas/1.4.1/html2canvas.min.js"
integrity="sha512-BNaRQnYJYiPSqHHDb58B0yaPfCu+Wgds8Gp/gU33kqBtgNS4tSPHuGibyoeqMV/TJlSKda6FXzoE
yYGjTe+vXA==" crossorigin="anonymous" referrerpolicy="no-referrer"></script>
        <script src="https://cdnjs.cloudflare.com/ajax/libs/jspdf/2.5.1/jspdf.umd.min.js"
integrity="sha512-qZvrmS2ekKPF2mSznTQsxqPgnpkI4DNTlrdUmTzrDgektczlKNRRhy5X5AAOnx5S09ydFYWWNSfc
EqDTTHgtNA==" crossorigin="anonymous" referrerpolicy="no-referrer"></script>
        <script type="module" src="scripts/aix_checklist.js"></script>
        <link rel="stylesheet" href="aix_checklist.css">
</head>
<body>
        <div class="quiz-container" id="home">
        <select class="vodiapicker">
        <option value="pt_br" data-thumbnail="images/pt-br/br.png"></option>
        <option value="en_us" data-thumbnail="images/en-us/us.png"></option>
        </select>

        <div class="lang-select">
        <button class="btn-select" value=""></button>
        <div class="option-container">
                <ul id="language-option"></ul>
        </div>
        </div>
        <div class="quiz-header">
        <p class="title" id="title1"></p>
        <p id="description"></p>
        <div style="text-align: center;">
        <img src="images/cne_logo.png" style="width:300px; border: none">
        </div>
        </div>
        <button id="start"></button>
        </div>

        <div id="quizzes">
        </div>

        <div class="quiz-container" id="results" style="display:none">
        <div class="quiz-header">
        <p class="title" id="title2"></p>
        <div style="text-align: center;">
                <img src="images/cne_logo.png" style="width:300px; border: none">
        </div>
        <div id="show-result">
        </div>
        </div>
```

```
        <button id="save-pdf"></button>
        <button id="back-to-home"></button>
        </div>
</body>
```

B1.2 aix_checklist.css

```
@import
url('https://fonts.googleapis.com/css2?family=Roboto:wght@200;300;400;500&display=swap');
*{
  box-sizing: border-box;
}
.language-selection {
  position: absolute;
  top: 0;
  right: 0;
  margin-right: 10px;
  margin-top: 10px;
  font-size: 15px;
}

.title{
  color:#187498;
  font-family: Roboto;
  font-size: 1.25rem;
  padding: 1.3rem;
  margin: 0;
  text-align: center;
}

.pdfTitle{
  font-size: 1.10rem;
  text-align: left;
  padding: 0.3rem;
  color:#187498;
  margin: 0;
  margin-top: 1cm;
  display: block;
}
html {
  overflow: scroll;
}
body{
  background-color: #b8c6db;
  background-image: linear-gradient(315deg, #b8c6db 0%, #f5f7f7 100%);
  font-family: 'Roboto', sans-serif;
  font-size: 0.9rem;
  display: flex;
  align-items: center;
  justify-content: center;
  overflow: scroll;
  margin: 0;
  min-height: 100vh;
}
.quiz-container{
  position: relative;
  background-color: #fff;
  border-radius: 10px;
  box-shadow: 0 0 10px 2px rgba(100, 100, 100, 0.1);
  width: clamp(320px, 100vw, 600px);
  overflow: hidden;
```

```
  min-height: 320px;
}
.quiz-header{
  padding: 5%;
  padding-bottom: 0;
}

input[type="radio"] {
  accent-color: #187498;
}

.answer-result {
    border-radius: 14px;
  font-size: 0.75rem;
}

.answered-correctly {
    color: #36AE7C;
}

.answered-wrongly {
    color: #EB5353;
}
h3{
  font-family: 'Roboto', sans-serif;
  font-size: 1rem;
  padding: 0.3rem;
  text-align: left;
  margin-top: 0;
}
h4{
  font-family: 'Roboto', sans-serif;
  font-size: 0.9rem;
  color: #A9A9A9;
  font-weight: normal;
  padding: 0.3rem;
  text-align: left;
  margin: 0;
  white-space: pre-line;
}
h5{
  font-family: 'Roboto', sans-serif;
  font-size: 0.75rem;
  color: #A9A9A9;
  font-weight: normal;
  padding: 0.3rem;
  text-align: left;
  margin: 0;
}
ul{
  list-style-type: none;
  padding: 0;
}
ul li{
  font-size: 1rem;
  margin: 1rem 0 0 0;
}

input {
  width: 1rem;
  height: 1rem;
}

ul li label{
```

```
  cursor: pointer;
}
img {
  display: block;
  border: 1px solid #ddd;
  border-radius: 4px;
  max-width: 100%;
  max-height: 450px;
  padding: 0.3rem;
  margin-top: 10px;
  margin-left: auto;
  margin-right: auto;
}
button{
  background-color: #187498;
  color: #fff;
  border: none;
  display: block;
  width: 98%;
  cursor: pointer;
  font-size: 1.0rem;
  font-family: Roboto;
  padding: 1.3rem;
  margin: 1%;
}
button:hover{
  background-color: #187498;
}
button:focus{
  outline: none;
  background-color: #187498;
}

.warning-text {
  color:#be2318;
  font-family: Roboto;
  font-size: 1.0rem;
  padding: 0.5rem 1.3rem 0 1.3rem;
  margin: 0;
  text-align: left;
}

.unselectable {
  -webkit-touch-callout: none;
  -webkit-user-select: none;
  -khtml-user-select: none;
  -moz-user-select: none;
  -ms-user-select: none;
  user-select: none;
}

.vodiapicker{
  display: none;
}

#language-option{
  padding-left: 0px;
  margin: 5 0;
}

#language-option li:hover{
 background-color: #F4F3F3;
}
```

```
#language-option li span, .btn-select li span{
  margin-left: 30px;
}

/* item list */

.option-container{
  display: none;
  width: 100%;
  max-width: 100px;
  box-shadow: 0 6px 12px rgba(0,0,0,.175);
  border: 1px solid rgba(0,0,0,.15);
  border-radius: 5px;
  position: absolute;
  padding: 0 5;
  background-color: #fff;
}

.open{
  display: show !important;
}

.btn-select{
  margin-top: 10px;
  width: 100%;
  max-width: 100px;
  height: 34px;
  border-radius: 5px;
  background-color: #F4F3F3;
  border: 1px solid transparent;
  display: flex;
  align-items: end;
  justify-content: center;
  padding: 0 5;
  margin: 0;
  box-shadow: inset 0 0px 0px 1px #ccc;
}
.btn-select li{
  list-style: none;
  float: left;
  padding-bottom: 0px;
}

.btn-select:hover li{
  margin-left: 0px;
}

.btn-select:hover{
  background-color: #F4F3F3;
  border: 1px solid transparent;
  box-shadow: inset 0 0px 0px 1px #ccc;
}

.btn-select:focus{
   outline:none;
   background-color: #F4F3F3;
}

.lang-select{
  position: absolute;
  top: 0;
  right: 0;
  margin-right: 10px;
  margin-top: 10px;
```

```
}

.lang-image {
  display: inline;
  border: none;
  border-radius: 0;
  max-width: 100%;
  max-height: 100%;
  padding: 0;
  margin-top: 0;
  margin-left: auto;
  margin-right: auto;
  width: 30px;
  height: 22px;
  overflow: hidden;
}
```

B1.3 scripts

B1.3.1 aix_checklist.js

```
import ChecklistItem from "./checklist_item.js";
import getQuizData from "./quiz_data.js";
import { getLabels, getLanguage, setLanguage } from "./translate.js";

window.jsPDF = window.jspdf.jsPDF;
window.html2canvas = html2canvas;

let quizItems;
let quizData;
const quizContainer = document.getElementById('quizzes');
let startButtonSubscription;
let backToHomeButtonSubscription;
let currentQuiz = 0;
let answers = [];
let answerFragment = undefined;
let savingPDF = false;
let count = 0;


const toggleSelection = () => {
        let expandTab = document.querySelector(".option-container");
        expandTab.style.display = expandTab.style.display == "block" ? "none" : "block";
}

document.querySelector(".btn-select").onclick = toggleSelection;

var langArray = [];
let language = getLanguage();
document.querySelectorAll('.vodiapicker option').forEach((lang) => {
  let img = lang.getAttribute("data-thumbnail");
  let value = lang.getAttribute("value");
  let item = '<li><img class="lang-image" src="'+ img +'" alt="" value="'+value+'"/></li>';
  langArray.push(item);
  let languageOption = document.createRange().createContextualFragment(item);

        //change button stuff on click
  languageOption.firstElementChild.addEventListener("click", (event) => {
        console.log("clicked", event.target);
```

```
        let imgElement = event.target.localName == "li" ? event.target.firstElementChild :
event.target;
        let img = imgElement.getAttribute("src");
        let value = imgElement.getAttribute('value');
        let item = '<li><img class="lang-image" src="'+ img +'" alt="" /></li>';
        document.querySelector('.btn-select').innerHTML = item;
        document.querySelector('.btn-select').setAttribute('value', value);
        setLanguage(value);
        configurePages();
        toggleSelection();
  });

  document.querySelector('#language-option').append(languageOption);

  if (value == language) {
        document.querySelector(".btn-select").innerHTML = item;
        document.querySelector(".btn-select").setAttribute('value', language);
  }

})


function loadQuiz() {
        for (let i = 0; i < quizData.length; i++) {
        let checklist_item = new ChecklistItem(quizData[i]);
        quizContainer.append(checklist_item.getFragment());
        quizItems.push(checklist_item);
        }

}

const handleStart = () => {
        document.getElementById('home').style.display = 'none';
        quizItems[0].setVisible(true);
}

const handleBackToHome = () => {
        answers = [];
        currentQuiz = 0;
        document.getElementById('results').style.display = 'none';
        document.getElementById('home').style.display = '';
        document.getElementById("show-result").innerHTML = "";
        answerFragment = undefined;
}

const handleSubmit = (answer) => {
        answers.push(answer);
        if (currentQuiz != quizItems.length - 1) {
        quizItems[currentQuiz].setVisible(false);
        quizItems[++currentQuiz].setVisible(true);
        } else {
        quizItems[currentQuiz].setVisible(false);
        backToHomeButtonSubscription =
document.getElementById('back-to-home').addEventListener('click', handleBackToHome);
        document.getElementById('results').style.display = '';
        showAnswers();
        }
}

function showAnswers() {
        let score = 0;
        let listItemsHTML = ""
        const symbols = {"correct": "+", "incorrect": "-", "NA": "#"}
        for (let i=0; i < quizData.length; i++) {
```

```
        let currentQuizData = quizData[i];
        let cssClass = "answered-wrongly";
        let symbol = symbols["incorrect"];

        if (answers[i] == currentQuizData.correct) {
        score++;
        cssClass = "answered-correctly";
        symbol = symbols["correct"];
        } else if (answers[i] == "c") {
        cssClass = ""
        symbol = symbols["NA"];
        }

        listItemsHTML += `
        <li style="margin: 0 0 0 0">
        <div class="answer-result">
                <h5 class="${cssClass}">${symbol}${currentQuizData.question}</h5>
        </div>
        </li>`
        }

        let weight = answers.filter(item => item != "c").length;

        let answerHTML = `
        <h3>${Math.floor(score / weight * 100)}% ${getLabels("RESULT_LABEL")}</h3>
        <ul>
        ${listItemsHTML}
        </ul>
        `

        answerFragment = document.createRange().createContextualFragment(answerHTML);

        document.getElementById("show-result").append(answerFragment);
}

function handleNavigation() {
        startButtonSubscription = document.getElementById('start').addEventListener('click',
handleStart);

        for (let i = 0; i < quizItems.length; i++) {
        quizItems[i].subscribe('onSubmit', handleSubmit);
        }


}

const savePDF = () => {
        var doc = new jsPDF('p', 'pt', 'a4');

        let pdfElement = document.createElement("div");
        pdfElement.style = "width: 580px; margin: 0; padding: 0 20px";

        let title = document.createRange().createContextualFragment(`<b class="pdfTitle"
id="titlePDF">${getLabels("TITLE_PDF")}</b>`);

        let footer = document.createRange().createContextualFragment(`<img
src="images/cne_logo.png" style="width:150px; border: none">`);

        pdfElement.append(title);
        let resultNode = document.getElementById('show-result').cloneNode(true);
        let h5Group = resultNode.getElementsByTagName("h5");
        console.log("H5 GROUP: ", h5Group);
        for (let i = 0; i < h5Group.length; i++) {
        h5Group[i].style.fontSize = "8pt";
```

```
        }
        pdfElement.append(resultNode);
        pdfElement.append(footer);

        doc.html(pdfElement, {
        callback: function (doc) {
        doc.save();
        },
        x: 10,
        y: 10
        });
}


document.getElementById('save-pdf').addEventListener('click', savePDF);

configurePages();
function configurePages() {
        quizData = getQuizData();
        document.getElementById('start').innerText = getLabels("START");
        document.getElementById('description').innerText = getLabels("DESCRIPTION");
        document.getElementById('title1').innerText = getLabels("TITLE");
        document.getElementById('title2').innerText = getLabels("TITLE");
        document.getElementById('back-to-home').innerText = getLabels("BACK_TO_HOME");
        document.getElementById('save-pdf').innerText = getLabels("SAVE_PDF");

        quizItems = [];
        quizContainer.innerHTML = "";

        loadQuiz();
        handleNavigation();
}
```

B1.3.2 checklist_item.js

```
import Observable from "./observer.js";
import { getLabels } from "./translate.js";

const quizTemplate = `
<div class="quiz-container" style="display:none" id="checklist-item">
        <p class="title" id="title"></p>
        <div class="quiz-header">
        <h3 id="question"></h3>
        <h4 id="explanation"></h4>
        <img id="imageUrl">
        <p id="required-warning" class="warning-text"></p>
        <ul>
        <li>
        <label class="unselectable">
                <input type="radio" name="answer" id="a" class="answer">
                <span id="a-text"><span/>
        </label>
        </li>
        <li>
        <label class="unselectable">
                <input type="radio" name="answer" id="b" class="answer">
                <span id="b-text"><span/>
        </label>
        </li>
        <li>
        <label id="c-item" class="unselectable">
```

```
                <input type="radio" name="answer" id="c" class="answer">
                <span id="c-text"><span/>
        </label>
        </li>
        </ul>
        </div>
        <button id="submit"></button>
</div>
`;

class ChecklistItem extends Observable {
        constructor(quizData) {
        super();
        this.quizData = quizData;
        this.fragment = document.createRange().createContextualFragment(quizTemplate);

        this.checklistItem = this.fragment.getElementById('checklist-item');
        this.answerEls = this.fragment.querySelectorAll('.answer')

        this.cItem = this.fragment.getElementById('c-item');

        this.requiredWarning = this.fragment.getElementById('required-warning');
        this.requiredWarning.style.visibility = "hidden";
        this.requiredWarning.innerText = getLabels("REQUIRED");

        this.fragment.getElementById('submit').addEventListener("click", () => {
        this.handleSubmit.call(this);
        }, false);

        this.fragment.getElementById('question').innerText = quizData.question;
        this.fragment.getElementById('explanation').innerHTML = quizData.explanation;
        this.fragment.getElementById("title").innerText = getLabels("TITLE");
        this.fragment.getElementById("a-text").innerText = getLabels("YES");
        this.fragment.getElementById("b-text").innerText = getLabels("NO");
        this.fragment.getElementById("c-text").innerText = getLabels("NA");
        this.fragment.getElementById("submit").innerText = getLabels("NEXT");


        this.fragment.getElementById('imageUrl').src = quizData.imageUrl;
        this.cItem.style.visibility = quizData.hasDoesNotApply ? "visible" : "hidden";

        this.deselectAnswers();
        this.getSelected();
        }

        deselectAnswers() {
        this.answerEls.forEach(answerEl => answerEl.checked = false)
        }

        getSelected() {
        let answer;
        this.answerEls.forEach(answerEl => {
        if(answerEl.checked) {
                answer = answerEl.id;
        }
        })
        return answer;
        }

        getFragment() {
        return this.fragment;
        }

        handleSubmit() {
```

```
        let selected = this.getSelected();

        if (selected) {
        this.requiredWarning.style.visibility = "hidden";
        this.notify("onSubmit", this.getSelected());
        } else {
        this.requiredWarning.style.visibility = "visible";
        }

        }

        setVisible(visible) {
        this.checklistItem.style.display = visible ? '' : 'none';
        }

}

export default ChecklistItem;
```

B1.3.3 observer.js

```
observer.js

class Observable {

        constructor() {
        this.observers = {};
        }

        subscribe(topic, func) {
        if (!this.observers[topic])
        this.observers[topic] = [];

        this.observers[topic].push(func);
        }

        unsubscribe(topic, func) {
        if (!this.observers[topic])
        return;

        this.observers[topic] = this.observers[topic].filter(subscriber => subscriber !==
func);
        }

        notify(topic) {
        if (!this.observers[topic])
        return;

        let args = [];

        for (let i = 1; i < arguments.length; i++) {
        args.push(arguments[i]);
        }

        this.observers[topic].forEach(observer => observer(args));
        }
}

export default Observable;
```

B1.3.4 quiz_data.js

```
import { getQuestionLabel, getExplanationLabel, getImageURL } from "./translate.js";

function getQuizData() {
        return [
        {
        question: getQuestionLabel(0),
        explanation: getExplanationLabel(0),
        hasDoesNotApply: false,
        correct: "a",
        imageUrl: getImageURL(0),
        },
        {
        question: getQuestionLabel(1),
        explanation: getExplanationLabel(1),
        hasDoesNotApply: false,
        correct: "a",
        imageUrl: getImageURL(1),
        },
        {
        question: getQuestionLabel(2),
        explanation: getExplanationLabel(2),
        hasDoesNotApply: false,
        correct: "a",
        imageUrl: getImageURL(2),
        },
        {
        question: getQuestionLabel(3),
        explanation: getExplanationLabel(3),
        hasDoesNotApply: false,
        correct: "a",
        imageUrl: getImageURL(3),
        },
        {
        question: getQuestionLabel(4),
        explanation: getExplanationLabel(4),
        hasDoesNotApply: false,
        correct: "a",
        imageUrl: getImageURL(4),
        },
        {
        question: getQuestionLabel(5),
        explanation: getExplanationLabel(5),
        hasDoesNotApply: false,
        correct: "a",
        imageUrl: getImageURL(5),
        },
        {
        question: getQuestionLabel(6),
        explanation: getExplanationLabel(6),
        hasDoesNotApply: false,
        correct: "a",
        imageUrl: getImageURL(6),
        },
        {
        question: getQuestionLabel(7),
        explanation: getExplanationLabel(7),
        hasDoesNotApply: true,
        correct: "a",
        imageUrl: getImageURL(7),
        },
```

```
{
question: getQuestionLabel(8),
explanation: getExplanationLabel(8),
hasDoesNotApply: true,
correct: "a",
imageUrl: getImageURL(8),
},
{
question: getQuestionLabel(9),
explanation: getExplanationLabel(9),
hasDoesNotApply: true,
correct: "a",
imageUrl: getImageURL(9),
},
{
question: getQuestionLabel(10),
explanation: getExplanationLabel(10),
hasDoesNotApply: false,
correct: "a",
imageUrl: getImageURL(10),
},
{
question: getQuestionLabel(11),
explanation: getExplanationLabel(11),
hasDoesNotApply: false,
correct: "a",
imageUrl: getImageURL(11),
},
{
question: getQuestionLabel(12),
explanation: getExplanationLabel(12),
hasDoesNotApply: false,
correct: "a",
imageUrl: getImageURL(12),
},
{
question: getQuestionLabel(13),
explanation: getExplanationLabel(13),
hasDoesNotApply: false,
correct: "a",
imageUrl: getImageURL(13),
},
{
question: getQuestionLabel(14),
explanation: getExplanationLabel(14),
hasDoesNotApply: true,
correct: "a",
imageUrl: getImageURL(14),
},
{
question: getQuestionLabel(15),
explanation: getExplanationLabel(15),
hasDoesNotApply: false,
correct: "a",
imageUrl: getImageURL(15),
},
{
question: getQuestionLabel(16),
explanation: getExplanationLabel(16),
hasDoesNotApply: true,
correct: "a",
imageUrl: getImageURL(16),
},
{
```

```
        question: getQuestionLabel(17),
        explanation: getExplanationLabel(17),
        hasDoesNotApply: true,
        correct: "a",
        imageUrl: getImageURL(17),
        },
        {
        question: getQuestionLabel(18),
        explanation: getExplanationLabel(18),
        hasDoesNotApply: true,
        correct: "a",
        imageUrl: getImageURL(18),
        },
        {
        question: getQuestionLabel(19),
        explanation: getExplanationLabel(19),
        hasDoesNotApply: true,
        correct: "a",
        imageUrl: getImageURL(19),
        },
        {
        question: getQuestionLabel(20),
        explanation: getExplanationLabel(20),
        hasDoesNotApply: false,
        correct: "a",
        imageUrl: getImageURL(20),
        },
        {
        question: getQuestionLabel(21),
        explanation: getExplanationLabel(21),
        hasDoesNotApply: true,
        correct: "a",
        imageUrl: getImageURL(21),
        },
        {
        question: getQuestionLabel(22),
        explanation: getExplanationLabel(22),
        hasDoesNotApply: true,
        correct: "a",
        imageUrl: getImageURL(22),
        },
        {
        question: getQuestionLabel(23),
        explanation: getExplanationLabel(23),
        hasDoesNotApply: true,
        correct: "a",
        imageUrl: getImageURL(23),
        },
        ];
}

export default getQuizData;
```

B1.3.5 translate.js

```
let language = window.navigator.language.startsWith('pt') ? 'pt_br' : 'en_us';

const LABELS = {
        en_us : {
        questions: [
```

```
        "Does the app make it clear what kind of object it can classify?",
        "Does the app explain which classes it can classify?",
        "Does the app make it clear how well it can do the image classification?",
        "Does the app provide understandable explanations?",
        "Does the app show instructions on how to take pictures with adequate quality?",
        "Does the app visualize the status while processing the classification?",
        "Does the app make it clear that there is uncertainty regarding the outcome of the
classification?",
        "Does the app indicate uncertainty in a way that is understandable by the target
audience?",
        "Does the app demonstrate the results in a useful way?",
        "Does the app make it clear that there is uncertainty when using the classification
result as part of another feature?",
        "Does the app provide information about how the ML model was developed?",
        "Does the app provide information about the use of user's photos being classified?",
        "Does the app support error recovery?",
        "Does the app help the user to recover from possible errors?",
        "Does the app indicate when the user tries to classify objects outside of its scope?",
        "Does the app show a warning when the system is not able to classify the photo with a
minimum level of confidence?",
        "Does the app allow the user to ask for a verification of the results by human
experts?",
        "The app allows users with knowledge in the application domain to send feedback
regarding the classification result?",
        "The app prohibits users without knowledge in the application domain to send feedback
regarding the classification result?",
        "The app makes the purpose of sending feedback clear?",
        "Is the app free of bias?",
        "Does the app indicate the proper precautions for taking pictures?",
        "Does the app highlight the risks involved with a potential misclassification?",
        "Does the app show visual elements of alert in case the classified object can cause
physical harm to humans?"
        ],

        explanations: [
        "The app presents the classes that it is able to distinguish (e.g., plants, dogs
breeds) <u>before</u> the user can classify an image (e.g., on the home screen).",
        "The app lists all classes it is able to classify.",
        "The app presents the user with the degree of its performance (e.g. accuracy)
<u>before</u> the user can classify an image.",
        "The app only uses terminology understandable by the target audience, avoiding
technical jargon, when presenting expectations and limitations.",
        "The app presents instructions/tips to guide the user to take pictures with adequate
quality for classification.",
        "The app presents a progress bar to visualize the processing during classification.",
        "The result of the classification is presented in a way that makes clear that there is
uncertainty in relation to this result.",
        "The classification result is presented in an understandable way for the target user,
e.g. using categorical values such as high/medium/low or very likely/probable/not likely or
presenting the n-best answer alternatives. The presentation of only confidence percentages is
avoided.\nN/A: if the app does not indicate uncertainty with the classification result",
        "The result of the classification is being presented in a way that helps the user to
make a decision according to the use case (e.g. when classifying venomous spiders it also
indicates what kind of medical assistance should be sought).\nNA: if the sole purpose is only
the indication of a class label",
        "The app introduces the uncertainty of the classification result in some way, even
when it is used directly as part of another functionality (e.g., marking locations on a map
that provide the classified object).\nNA: Image classification is not integrated into other
functionality",
        "The app shows information about the development of the ML model, including eg.
information about the amount of images used in training, by whom the images were labeled, what
type of ML model is being used and its performance.",
        "There is information about how the user photos to be classified are used and if they
are (or are not) persistently stored. If they are stored, the conditions of this storage and
```

```
access to these images is specified.",
        "The app allows the user to easily redo the classification when a classification error
occurs (e.g. keeping the button to take a picture).",
        "The app explains what to do when a classification error occurs (e.g. asking the user
to take another photo).",
        "If the image is of an object outside the scope of the app (e.g. a glass in a dog
classification app), the app displays as a result the information that this is an object
outside the scope of this app's classification.\nNA: if the app aims to classify any object",
        "In cases where it is not possible to classify the photo with sufficient confidence
(e.g. < 70%), the app alerts the user informing that it was unable to classify this image
(instead of showing results even with a low confidence level).",
        "The app provides a means of contacting domain experts to verify the classification
result.\nN/A: If the app does not need the safety of human validation.",
        "The app makes it possible for a user with domain knowledge to indicate whether a
classification result is correct or not. \nN/A: if the app targets an audience without domain
knowledge",
        "The app does not offer the option to send feedback on the correctness of
classification results to users who might not provide such feedback correctly.\nN/A: if the
app targets an audience with domain knowledge",
        "The app allows the user to understand how her/his feedback may affect the
functioning, and guides the user to provide it carefully.\nNA: in case the app does not
provide the possibility of feedback",
        "There is no reinforcement of social bias, prejudice or use of inappropriate
terminology in the user interface.",
        "In cases in which taking a picture could be risky for the user (e.g., trying to take
a picture of a venomous snake) guidelines are presented for doing so safely and alerting the
user of the danger.\nN/A: in case the photos can be taken without danger",
        "The app indicates the consequences of a possible misclassification, specifically in
cases where this could result in physical harm to humans.\nN/A: in case classification errors
do not carry many risks",
        "The app uses visual elements  (e. g. color and/or icon) to alert the user to the
hazard by the classified object. \nNA: in case the scope of the app does not contain hazardous
objects",
        ],

        imageURL: [
        "images/en-us/item1.jpg",
        "images/en-us/item2.jpg",
        "images/en-us/item3.jpg",
        "images/en-us/item4.jpg",
        "images/en-us/item5.jpg",
        "images/en-us/item6.jpg",
        "images/en-us/item7.jpg",
        "images/en-us/item8.jpg",
        "images/en-us/item9.jpg",
        "images/en-us/item10.jpg",
        "images/en-us/item11.jpg",
        "images/en-us/item12.jpg",
        "images/en-us/item13.jpg",
        "images/en-us/item14.jpg",
        "images/en-us/item15.jpg",
        "images/en-us/item16.jpg",
        "images/en-us/item17.jpg",
        "images/en-us/item18.jpg",
        "images/en-us/item19.jpg",
        "images/en-us/item20.jpg",
        "images/en-us/item21.jpg",
        "images/en-us/item22.jpg",
        "images/en-us/item23.jpg",
        "images/en-us/item24.jpg",
        ],

        misc : {
        "TITLE" : "AIX Checklist - Image Classification",
```

```
        "DESCRIPTION" : "This checklist can be used to perform a heuristic evaluation of the
interface design of an intelligent image classification app. The checklist contains 24 items
and is based on existing AI heuristics from both academic literature and commercial guidelines
for using AI.",
        "START" : "START",
        "SAVE_PDF": "SAVE PDF",
        "BACK_TO_HOME": "BACK TO HOME",
        "NEXT" : "NEXT",
        "YES" : "Yes",
        "NO" : "No",
        "NA" : "Not applicable",
        "RESULT_LABEL": "of the checklist items are met",
        "REQUIRED": "Required",
        "TITLE_PDF": "Results of the heuristic evaluation - AIX - image classification",
        }
        },

        pt_br: {

        questions: [
        "O app deixa claro qual tipo de objeto ele pode classificar?",
        "O app explica quais classes ele pode classificar?",
        "O aplicativo deixa claro o quão bem ele pode fazer a classificação de imagens?",
        "O aplicativo fornece explicações compreensíveis?",
        "O app mostra dicas de como tirar fotos com qualidade adequada?",
        "O app visualiza o status durante o processamento da classificação?",
        "O app deixa claro que existe incerteza em relação ao resultado da classificação?",
        "O app indica a incerteza de forma compreensível pelo público alvo?",
        "O app demonstra os resultados de forma útil?",
        "O app deixa claro que existe incerteza quando utiliza o resultado da classificação em
outra funcionalidade?",
        "O app fornece informações sobre como o modelo de ML foi desenvolvido?",
        "O app disponibiliza informações sobre o uso das fotos do usuário usadas na
classificação?",
        "O app permite a recuperação de erros?",
        "O app ajuda o usuário a se recuperar de possíveis erros?",
        "O app indica quando se trata de objetos fora do seu escopo de classificação?",
        "O app mostra um aviso quando o sistema não é capaz de classificar uma foto com
confiança suficiente?",
        "O app permite que o usuário solicite a verificação do resultado por especialistas
humanos?",
        "O app permite que usuários com conhecimento no domínio do aplicativo possam enviar
feedback referente ao resultado da classificação?",
        "O app proíbe usuários sem conhecimento no domínio do aplicativo enviem feedback
referente ao resultado  da classificação?",
        "O app deixa claro o propósito de enviar feedback?",
        "O app está livre de viés?",
        "O app indica os devidos cuidados para tirar fotos?",
        "O app destaca os riscos envolvidos com um possível erro de classificação?",
        "O app mostra elementos visuais de alerta caso o objeto classificado possa causar
danos físicos a humanos?",
        ],

        explanations: [
        "O app apresenta as classes que é capaz de distinguir (p.ex plantas, raças de
cachorro) <u>antes</u> do usuário poder classificar uma imagem. (p.ex. na tela home).",
        "O app lista todas as classes que é capaz de distinguir",
        "O app apresenta ao usuário o grau do seu desempenho (p.ex. acurácia) <u>antes</u> do
usuário poder classificar uma imagem. p.ex. na tela home).",
        "O app utiliza apenas uma terminologia compreensível pelo público alvo, evitando
jargão técnico, ao apresentar as expectativas e limitações.",
        "O app apresenta instruções/dicas para guiar o usuário a tirar fotos com qualidade
adequada para a classificação.",
        "O app apresenta elementos para visualizar o status do progresso durante o
```

```
processamento da classificação.",
        "O resultado da classificação é apresentado indicando que existe uma incerteza em
relação a este resultado.",
        "O resultado da classificação é apresentado de forma compreensível para o usuário
alvo, p. Ex. usando valores categóricos como alto/médio/baixo ou muito provável/provável/pouco
provável  ou apresentando as n-melhores alternativas de resposta. É evitada a apresentação de
apenas percentuais de confiança.\nN/A: caso o app não indica incerteza com o resultado da
classificação",
        "O resultado da classificação está sendo apresentado de forma que ajude o usuário a
tomar uma decisão de acordo com o caso de uso (p.ex. na classificação de aranhas peçonhentas a
indicação de que tipo de assistência médica deve ser procurada).\nN/A: caso o único objetivo é
a classificação sem nenhuma outra finalidade",
        "O app apresenta a incerteza do resultado de classificação de alguma forma, mesmo
quando é usado diretamente como parte de outra funcionalidade (p.ex.  marcar locais em um mapa
que oferecem o objeto classificado).\nN/A: A classificação de imagens é a única função do
app",
        "O app mostra informações sobre o desenvolvimento do modelo de ML, incluindo p.ex.
informações sobre a quantidade de imagens usadas no treinamento, por quem as imagens foram
rotuladas, que tipo de modelo de ML está sendo usado e qual o desempenho.",
        "Existe informação sobre como as fotos do usuário a serem classificadas são utilizadas
e se elas são (ou não) armazenadas persistentemente. Caso sejam armazenadas, as condições
deste armazenamento e o acesso a estas imagens é especificado.",
        "O app permite que o usuário possa facilmente refazer a classificação quando ocorre um
erro de classificação (p.ex. manter o botão de tirar foto).",
        "O app explica o que fazer quando ocorre um erro de classificação (p.ex. Solicitando
tirar outra foto).",
        "Caso a imagem seja de um objeto não sendo classificado pelo app (p.ex. um copo num
app de classificação de cachorros), o app apresenta como resultado a informação que se trata
de um objeto fora do escopo de classificação deste app.\nN/A: caso o app visa classificar
qualquer objeto",
        "Em casos em que não é possível classificar uma foto com confiança suficiente (p.ex <
70%), o app alerta ao usuário informando que não foi capaz de classificar esta foto (ao invés
de mostrar resultados mesmo com grau de confiança baixa).",
        "O app fornece um meio de entrar em contato com especialistas de domínio para
verificar o resultado da classificação.\nN/A: Caso o app não precise da segurança da validação
humana.",
        "O app possibilita que um usuário com conhecimento de domínio  possa indicar se um
resultado de classificação está correto ou não.\nN/A: caso o app se direciona a um público
alvo sem conhecimento de domínio",
        "O app está sem a opção de feedback de corretude de resultado de classificação no caso
em que seus usuários não poderiam fornecê-lo corretamente.\nN/A: caso o app se direciona a um
público alvo com conhecimento de domínio",
        "Caso o app forneça a possibilidade de feedback, o app permite que o usuário entenda
como seu feedback pode afetar o funcionamento, sendo orientado a fornecê-lo com cuidado.\nN/A:
caso que o app não fornece a possibilidade de feedback",
        "A interface do usuário está livre de reforço de viés social, preconceito ou uso de
terminologia inapropriada.",
        "No caso em que a captura de fotos pode ser arriscada para o usuário (p.ex. tentando
tirar foto de uma cobra peçonhenta) são apresentadas diretrizes para fazê-lo de um modo seguro
e alertar o usuário sobre o perigo.\nN/A: caso a captura de fotos pode ser feito sem perigo",
        "O app indica as consequências de um possível erro de classificação, especificamente
em casos em que isso pode resultar em danos físicos a humanos.\nN/A: caso que erros de
classificação não trazem muitos riscos",
        "O app mostra de forma visual (p.ex. usando cor e/ou ícone) para alertar o usuário
sobre o perigo pelo objeto classificado.\nN/A: caso o escopo do app não contêm objetos
perigosos."
        ],

        imageURL: [
        "images/pt-br/item1.jpg",
        "images/pt-br/item2.jpg",
        "images/pt-br/item3.jpg",
        "images/pt-br/item4.jpg",
        "images/pt-br/item5.jpg",
```

```
        "images/pt-br/item6.jpg",
        "images/pt-br/item7.jpg",
        "images/pt-br/item8.jpg",
        "images/pt-br/item9.jpg",
        "images/pt-br/item10.jpg",
        "images/pt-br/item11.jpg",
        "images/pt-br/item12.jpg",
        "images/pt-br/item13.jpg",
        "images/pt-br/item14.jpg",
        "images/pt-br/item15.jpg",
        "images/pt-br/item16.jpg",
        "images/pt-br/item17.jpg",
        "images/pt-br/item18.jpg",
        "images/pt-br/item19.jpg",
        "images/pt-br/item20.jpg",
        "images/pt-br/item21.jpg",
        "images/pt-br/item22.jpg",
        "images/pt-br/item23.jpg",
        "images/pt-br/item24.jpg",
        ],


        misc : {
        "TITLE" : "Checklist AIX - Classificação de Imagens",
        "DESCRIPTION": "Este checklist pode ser utilizado para realizar uma avaliação
heurística do design de interface de um app inteligente para classifcação de imagens. O
checklist contem 24 itens e é baseado em heurísticas existentes de IA em ambos a literatura
acadêmica e diretrizes comerciais do uso de IA.",
        "START" : "COMEÇAR",
        "SAVE_PDF": "SALVAR PDF",
        "BACK_TO_HOME": "VOLTAR AO INÍCIO",
        "NEXT" : "PRÓXIMO",
        "YES" : "Sim",
        "NO" : "Não",
        "NA" : "Não se aplica",
        "RESULT_LABEL": "dos itens do checklist são atendidos",
        "REQUIRED": "Obrigatório",
        "TITLE_PDF": "Resultados da avaliação heurística - AIX - classificação de imagens",
        }

        }
}

function getQuestionLabel(quizNO) {
        return LABELS[language].questions[quizNO];
}

function getExplanationLabel(quizNO) {
        return LABELS[language].explanations[quizNO];
}

function getImageURL(quizNO) {
        return LABELS[language].imageURL[quizNO];
}

function getLabels(key) {
        return LABELS[language]["misc"][key];
}

function setLanguage(lang) {
        console.log("lang: ", lang);
        language = lang;
}
```

```
function getLanguage() {
        return language;
}

export {getQuestionLabel, getExplanationLabel, getImageURL, getLabels, getLanguage,
setLanguage};
```

APÊNDICE C - Artigo

DESENVOLVIMENTO DE UM *CHECKLIST* DE AVALIAÇÃO DE HEURÍSTICAS DE EXPERIÊNCIA DE USUÁRIO DE APLICATIVOS COM INTELIGÊNCIA ARTIFICIAL

Gustavo Dirschnabel

Grupo de Qualidade de Software (GQS)
Universidade Federal de Santa Catarina (UFSC) - Florianópolis/SC

gustavo.dirschnabel@grad.ufsc.br

Abstract. Despite the existence of User Experience (UX) research on traditional mobile applications, there does not exist much information on how to adapt the UX heuristics for the context of apps with Artificial Intelligence (AI). In this paper, based on AIX principles, a set of heuristics for the evaluation of user interfaces for image classification intelligent apps is defined and operationalized by the development of a checklist. The heuristics and checklist are validated through an initial case study with user interfaces of 101 Android apps. The aim of this study is to contribute to the improvement of the user experience of mobile applications aimed at image classification.

Resumo. Apesar de já existir pesquisa sobre *User Experience* (UX) em aplicativos móveis tradicionais, ainda não existe muita informação sobre como adaptar as heurísticas de UX para o contexto de aplicativos inteligentes (IA). Assim, no presente projeto, com base em princípios de AIX, é definido um conjunto de heurísticas para a avaliação de interfaces de aplicativos móveis com inteligência artificial para classificação de imagens, operacionalizada por meio do desenvolvimento de um *checklist*. As heurísticas e o *checklist* são validados por meio de um estudo de caso com interfaces de apps Android. Visa-se como resultado contribuir para a melhoria da experiência de usuário de aplicativos móveis voltados a classificação de imagens.

1. Introdução

Recentemente, tecnologias de *Machine Learning* (ML) estão sendo incorporadas amplamente em aplicativos móveis. Nestes, ML pode ser utilizada com processamento completamente em nuvem, para grandes conjuntos de dados e modelos complexos, ou com processamento no dispositivo [Dai et al., 2020]. Um tipo de aplicativo que utiliza ML são os classificadores de imagem, que realizam a tarefa de categorizar imagens em uma de várias classes pré-definidas [Rawat, 2017].

Implementações de ML em dispositivos móveis enfrentam uma série de desafios, como poder computacional, bateria, pouca memória, tempo de resposta e riscos de privacidade [Dai et al., 2020]. Entre esses uma das principais questões se

refere a UX desses aplicativos inteligentes. Aplicativos com IA apresentam características diferentes que os tradicionais, onde destaca-se a questão de apresentação de resultados de forma probabilística, e a adaptação/melhoria ao longo do uso, que representa um modelo mental diferente ao qual os usuários estão acostumados. Essas características, se não tratadas com cuidado, podem compor um problema de usabilidade que leva o usuário à tomada de decisões erradas, frustração e abandono do software [Google, 2022a]. Usabilidade é definida como a capacidade de um usuário de completar seus objetivos com o software com eficácia, eficiência e satisfação [ISO/IEC, 2010]. Os fatores de usabilidade representam boa parte das características da qualidade de uso de software, no ponto de vista do usuário final [ISO/IEC, 2010; ISO/IEC, 2011]. Já UX é definida como uma combinação de percepções e respostas do usuário, como emoções, crenças, preferências, conforto, comportamentos e realizações que advém de períodos anteriores, durante e após o uso do software [ISO/IEC, 2020].

Estes desafios de projeto para UX em sistemas inteligentes estão levando à criação da área de AIX - AI Experience Design [Subramonyam et al., 2021], propondo diretrizes para aplicativos com IA como, por exemplo, para desenvolvimento de interfaces de usuário. O principal motivador para aplicação de diretrizes em projetos de ML é desenvolver software com melhor UX, e que tenha boa percepção do usuário nos quesitos de oferecer mais controle, confiabilidade e uma sensação de produtividade [Li et al., 2022].

Nesse contexto já estão sendo propostos alguns conjuntos de heurísticas para a avaliação de design de interface de usuário, inclusive de grandes empresas de tecnologia, como da Microsoft [Li et al., 2022], Apple (2022) e Google (2022a). Esta avaliação heurística consiste em inspeções da interface considerando um conjunto de princípios de usabilidade, conhecidos como heurísticas [NIELSEN, 1994]. O conjunto de heurísticas de usabilidade mais comumente utilizadas para o desenvolvimento de aplicações desktop foram desenvolvidas por Nielsen [1994].

Porém, observando essa variedade de conjuntos existentes, surge a pergunta: Até que ponto esses conjuntos são semelhantes ou abordam heurísticas distintas? Atualmente também ainda não existe um checklist, feito a partir dessas diretrizes, para auxiliar a realização de uma avaliação heurística de interfaces de aplicativos móveis. Além disso, essas heurísticas propostas são mais voltadas a sistemas de recomendação e assim se observa-se a falta de heurísticas mais específicas voltadas à classificação de imagens. Em geral, existe pouca pesquisa sobre o assunto no âmbito acadêmico [Li et al., 2022].

2. Estado da arte

Um mapeamento sistemático da literatura foi realizado com o objetivo de levantar o estado da arte sobre heurísticas de avaliação de interface de usuário de aplicativos inteligentes. O mapeamento segue a metodologia proposta por Petersen, Vakkalanka e Kuzniarz (2015) e responde à pergunta de pesquisa: Quais conjunto de heurísticas existem para a avaliação do design de interface de aplicativos inteligentes para

smartphones Android. Seu foco principal está na avaliação de aplicativos que possuem funcionalidades de classificação de imagens. No entanto, são consideradas abordagens de avaliação de aplicativos inteligentes em geral, desde que contenham princípios que possam ser aplicados para classificação de imagens.

A pergunta de pesquisa é decomposta nas seguintes perguntas de análise:

PA1. Quais conjuntos de heurísticas de avaliação do design de interface existem e quais suas características?

PA2. Quais são as heurísticas destes modelos?

PA3. Qual o suporte existente para avaliação utilizando os conjuntos de heurísticas encontrados (como checklist ou ferramenta). Até que ponto essa avaliação foi automatizada?

PA4. Como o conjunto de heurísticas foi desenvolvido e avaliado?

As buscas foram realizadas nas principais bases de dados e bibliotecas digitais da área da computação: ACM Digital Library, IEEE Xplore Digital Library, Wiley e Scopus. Além dessas bases de dados, com o intuito de abranger uma maior gama de publicações, foram realizadas buscas no Google Scholar, que indexa um grande conjunto de dados de diversas fontes de produção científica [Haddaway et al., 2015]. Buscas informais foram utilizadas para calibrar a string de busca, a qual foi definida com base nos termos relevantes da pergunta de pesquisa do mapeamento. Durante essas buscas fora do protocolo do mapeamento, notou-se a existência de poucos artigos relevantes para o foco específico da pesquisa, e foram encontradas diretrizes propostas por corporações sem a publicação de um artigo correspondente. Essas diretrizes foram encontradas anexadas sob a página ML+DESIGN(2022), e este último foi incluído como base.

Portanto, decidiu-se ampliar o escopo das buscas, resultando nos termos de busca, sinônimos e termos similares. Assim, definiu-se a seguinte string de busca genérica:

(heuristic OR guidelines OR principle* OR recommendation*) AND (GUI OR UI OR “user experience” OR UX OR “user interface” OR usability) AND (evaluat* OR assess* OR identif*) AND (“artificial intelligence” OR AI OR “machine learning” OR ML OR “deep learning” OR DL) AND (“human-ai” OR AIX OR XAI OR IML OR “AI Experience” OR “human in the loop” OR “Human-centered”).

As buscas foram realizadas em outubro de 2022 em três etapas. Na primeira etapa foi aplicada a string de busca nas bases de dados. Na segunda etapa, foram aplicados os critérios de inclusão e exclusão sob os resumos dos resultados mais relevantes de cada base (limitando-se a 200 obras em cada), resultando em uma lista de artigos potencialmente relevantes. Na terceira etapa, todo o texto dos artigos potencialmente relevantes foi analisado, aplicando novamente os critérios de inclusão/exclusão e de qualidade. Como resultado foram identificadas 7 publicações relevantes (Tabela 1).

Tabela 1 - Resultado da busca

Base	Resultados da busca	Artefatos Analisados	Quantidade de publicações potencialmente relevantes	Publicações relevantes
ACM	39	39	5	4
IEEE	10	10	0	0
Google Scholar	77	77	1	0
ML+Design	8	8	4	3
Scopus	63	63	2	0
Wiley	4	4	0	0
Total (sem duplicatas)				7

Como resultado dessa revisão sistemática foi observada a existência de poucas pesquisas voltadas ao desenvolvimento de heurísticas para avaliação de AIX. Entre as publicações encontradas, apesar de não indicarem um escopo específico, é possível perceber que grande parte concentra-se em sistemas de recomendação, por exemplo a proposta de Amershi et al. (2019) com quase a metade das 18 heurísticas voltadas a sistemas que adaptam suas funcionalidades baseados na ações do usuário. Não foi encontrado nenhum conjunto de heurísticas projetado especificamente para a tarefa de classificação de imagens.

Alguns dos conjuntos de heurísticas encontrados apresentaram formatos variados, como Mohseni et al. (2021), em que as heurísticas para avaliação de interfaces gráficas são um subconjunto de heurísticas de todo o sistema, incluindo também heurísticas voltadas ao estabelecimento de requisitos e objetivos para a explicabilidade no sistema, e heurísticas para o projeto de algoritmos interpretáveis. No conjunto da Apple (2022), os 60 princípios estão fracamente agrupados em o que se resume a 9 heurísticas extensas.

Observa-se também a falta de suporte à avaliação heurística nas pesquisas encontradas, basicamente com nenhuma apresentando checklists ou uma ferramenta de suporte para (semi-) automatizar a avaliação. Além da apresentação das heurísticas, a maioria das pesquisas limita-se ao método de comparação com um exemplo e contraexemplo de aplicação das heurísticas.

Assim os resultados dessa revisão sistemática indicam a falta de um conjunto de heurísticas de AIX projetado especificamente para aplicativos de classificação de imagens, bem como o suporte à avaliação.

3. Desenvolvimento de Heurísticas e *checklist*

No escopo desta pesquisa, é definido um novo conjunto de heurísticas de usabilidade de IA a partir da metodologia de Rusu et al. (2011), focando em aplicativos móveis de classificação de imagens. Inserido no contexto de pesquisa da iniciativa Computação na

Escola/INCoD/INE/UFSC, visa-se também a customização destas heurísticas em um contexto educacional. Para apoiar a operacionalização da avaliação heurística é desenvolvido também um checklist a partir desse conjunto de heurísticas.

Para iniciar o processo de criação do novo conjunto de heurísticas foi feita uma análise dos conjuntos de heurísticas encontrados como resultado do levantamento do estado da arte para destacar as informações relevantes providas por cada um. Foi proposto um mapeamento de todas as heurísticas, identificando a correlação entre os conjuntos encontrados e explicitando todas as diretrizes que compõem estas heurísticas.

Observou-se pelo mapeamento que há, entre as publicações, uma concordância maior sobre a importância dos princípios que tratam de apresentação inicial das capacidades do sistema de IA, devolução de controle ao usuário em caso de erro, coleta de feedback para aperfeiçoamento contínuo do modelo de ML e formatação dos resultados para apresentação para o usuário. Por outro lado, 17 diretrizes estão presentes em apenas uma das conjuntos analisadas, com diversas questões tendo importância atribuída em somente um destes. Como exemplo, somente Amershi et al. (2019) trata da questão de respeito a normas sociais e o conjunto da Google (2022a) é o único a ressaltar a importância de não confundir o usuário com uma UI diferente do padrão para o dispositivo, que prejudicaria ainda mais o modelo mental do usuário.

Em seguida, a partir de um estudo de caso envolvendo aplicativos da iniciativa computação na escola [Computação na Escola, 2023] e aplicativos comerciais da Google Play, foi realizado um refinamento do mapeamento em que foram escolhidas diretrizes relevantes para o contexto de aplicativos de classificação de imagens. Foram considerados aspectos como incapacidade de retreinamento automático e obtenção de explicações de modelos do *Google Teachable Machine* [Google, 2022b], a ausência de personalização da experiência, a necessidade de responsabilidade com a segurança do usuário e questões sociais para filtrar as diretrizes. O resultado foi formalizado em um conjunto de 8 heurísticas seguindo um modelo padrão de Rusu et al (2011), que foi decomposto em uma versão inicial de um *checklist*, consistindo em 20 itens respondidos com a escala de “Não”, “Sim” ou item opcional (Não se aplica, N/A), contando com explicações em cada item e com uma imagem de exemplo de aplicação associado a ele, como pode ser visto na Figura 1.

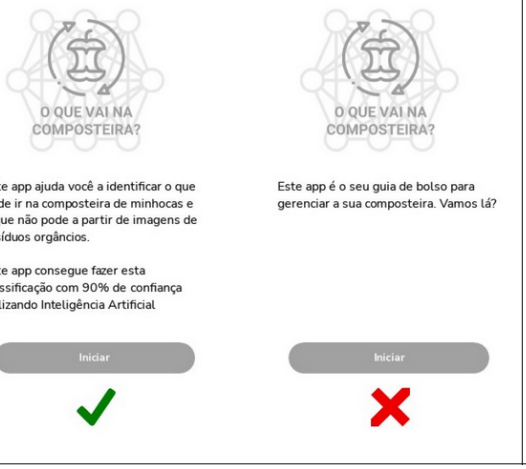
Heurística	Item de *checklist*	Explicação	Exemplo	Escala de resposta
Deixar as expectativas e limitações explícitas	1. O app deixa claro o que pode fazer?	O app apresenta as categorias que é capaz de distinguir antes do usuário poder classificar uma imagem.		Sim, Não

Figura 1. Exemplo de um item do *checklist*

Para verificar a qualidade do conjunto de heurísticas e do checklist desenvolvido foi realizada uma avaliação com um painel de especialistas. Entretanto, enquanto Rusu et al. (2011) sugere uma comparação entre as heurísticas de Nielsen (1994) com as desenvolvidas, neste trabalho optou-se por solicitar que os especialistas julguem aspectos como a completude, corretude, consistência e ambiguidade das heurísticas e o checklist a partir da aplicação dele em uma avaliação heurística, seguindo a metodologia descrita por Lawshe (1975).

A avaliação por painel de especialistas foi executada em abril de 2023 por seis pesquisadores das áreas de ciência da computação e design. A avaliação foi feita em dois passos: primeiramente o avaliador aplicou o *checklis*t para avaliar um app com classificação de imagens e em seguida avaliou os fatores de qualidade do *checklist*. A cada especialista foi associado um aplicativo inteligente de classificação de imagens diferente. As instruções, o checklist e o questionário de avaliação foram enviados em forma de um formulário online em que as heurísticas e itens do checklist eram apresentados e imediatamente utilizados para avaliar este aplicativo inteligente. Após essa tarefa, cada especialista respondeu uma sequência de perguntas para relatar os problemas que identificaram durante o processo.

A maioria dos especialistas consideraram o checklist útil e aplicável no contexto de avaliação de desempenho no ensino de computação, e o modelo corretamente decomposto em heurísticas e checklist. Foram identificados pontos fracos na distribuição de itens opcionais e nas explicações dos itens, que foram consideradas insuficientes e causavam uma grande dependência nas imagens de exemplo para a execução da avaliação. As questões levantadas pelo painel de especialistas foram utilizadas em um processo de refinamento das heurísticas e *checklist*, que resultou na especificação da versão v0.2 deste.

4. Avaliação Empírica

Com o objetivo de iniciar a avaliação de confiabilidade e validade da estrutura das heurísticas e *checklist* foi realizado uma análise estatística de caráter exploratório sobre a versão v0.2. Seguindo a metodologia Goal/Question/Metric [Basili et al., 1994], foram definidas as seguintes perguntas de análise:

Confiabilidade:

P1. Existe evidência de consistência interna do checklist?

Validade da estrutura:

P2. Existe evidência de validade convergente do checklist?

P3. Como fatores subjacentes influenciam as respostas dos itens do checklist?

Coleta de dados. Os dados utilizados nesta análise foram pontuações atribuídas a aplicativos inteligentes de classificação de imagens a partir de uma avaliação heurística utilizando a versão 0.2 do checklist. A amostra consistiu em 101 avaliações heurísticas em aplicativos da plataforma Google Play com classificação de imagens e disponíveis gratuitamente, selecionados de modo aleatório. As avaliações foram realizadas no mês de maio/2023.

Uma característica percebida nos dados coletados foi a predominância de aplicativos com características similares, que se encaixam em sua grande maioria em diversos critérios na categoria de resposta "Não se aplica". Em 8 itens nos dados coletados 70% ou mais das respostas eram "Não se aplica". Esses aspectos e o tamanho da amostra ainda relativamente pequeno contribuem para que as análises executadas sejam em grande parte, apenas uma indicação dos resultados esperados com dados mais representativos e em maior número.

Preparação dos dados para análise. Para a execução das análises, os dados devem ser codificados numericamente. Como respostas do tipo "Não se aplica" representam a ausência daquele item, seria recomendável a remoção total dos "N/A"s, associando números apenas para "Sim" e "Não". Porém, os métodos de análise falharam com essa codificação, retornando valores de erro. Para contornar esse problema, foram executadas as análises com duas configurações alternativas:

Alternativa 1: Codificação apenas de "Sim" e "Não", com exclusão de "N/A". Itens com mais de 90% de respostas "N/A" foram excluídos da análise, resultando em apenas os itens da Tabela 15.

Alternativa 2: Codificação de todas as respostas.

Metodologia. A consistência interna do checklist foi medida a partir do coeficiente alfa de Cronbach (1951) e o coeficiente ômega de Mcdonald (2013). Para determinar se as dimensões (heurísticas) estão bem definidas no checklist, análises de correlação foram realizadas analisando a correlação entre itens (correlação policórica) e correlação item total. Por último, uma análise fatorial foi executada para determinar o número de fatores que influenciam nas respostas dos itens.

Tabela 3 - *Checklist* v0.2

Heurística	Item de *checklist*	Explicação do item	Escala de resposta
Deixar as expectativas e limitações explícitas	1. O app deixa claro quais classes ele pode classificar?	O app apresenta as classes que é capaz de distinguir antes do usuário poder classificar uma imagem. (p.ex. na tela home).	Sim, Não
	2. O aplicativo deixa claro o quão bem ele pode fazer a classificação de imagens?	O app apresenta ao usuário o grau do seu desempenho (p.ex. acurácia) antes do usuário poder classificar uma imagem. p.ex. na tela home).	Sim, Não
	3. O aplicativo fornece explicações compreensíveis?	O app utiliza apenas uma terminologia compreensível pelo público alvo, evitando jargão técnico, ao apresentar as expectativas e limitações.	Sim, Não
Apoiar o uso efetivo	4. O app mostra dicas de como tirar fotos com qualidade adequada?	O app apresenta instruções/dicas para guiar o usuário a tirar fotos com qualidade adequada para a classificação.	Sim, Não
	5. O app visualiza o status durante o processamento da classificação?	O app apresenta elementos para visualizar o status do progresso durante o processamento da classificação.	Sim, Não
Apoiar a compreensão do usuário sobre incerteza e a confiança do modelo	6. O app deixa claro que existe incerteza em relação ao resultado da classificação?	O resultado da classificação é apresentado indicando que existe uma incerteza em relação a este resultado.	Sim, Não
	7. O app indica a incerteza de forma compreensível pelo público alvo?	O resultado da classificação é apresentado de forma compreensível para o usuário alvo, p. Ex. usando valores categóricos como alto/médio/baixo ou muito provável/provável/pouco provável ou apresentando as n-melhores alternativas de resposta. É evitada a apresentação de apenas percentuais de confiança. N/A: caso o app não indica incerteza com o resultado da classificação	Sim, Não, N/A
	8. O app demonstra os resultados de forma útil?	O resultado da classificação está sendo apresentado de forma que ajude o usuário a tomar uma decisão de acordo com o caso de uso (p.ex. na classificação de aranhas peçonhentas a indicação de que tipo de assistência médica deve ser procurada). N/A: caso o único objetivo é a classificação sem nenhuma outra finalidade	Sim, Não, N/A
	9. O app deixa claro que existe incerteza quando utiliza o resultado da classificação em outra funcionalidade?	O app apresenta a incerteza do resultado de classificação de alguma forma, mesmo quando é usado diretamente como parte de outra funcionalidade (p.ex. marcar locais em um mapa que oferecem o objeto classificado). N/A: A classificação de imagens é a única função do app	Sim, Não, N/A
	10. O app fornece informações sobre como o modelo de ML foi desenvolvido?	O app mostra informações sobre o desenvolvimento do modelo de ML, incluindo p.ex. informações sobre a quantidade de imagens usadas no treinamento, por quem as imagens foram rotuladas, que tipo de modelo de ML está sendo usado e qual o desempenho.	Sim, Não
Assegurar privacidade e segurança de dados	11. O app disponibiliza informações sobre o uso das fotos do usuário usadas na classificação?	Existe informação sobre como as fotos do usuário a serem classificadas são utilizadas e se elas são (ou não) armazenadas persistentemente. Caso sejam armazenadas, as condições deste armazenamento e o acesso a estas imagens é especificado.	Sim, Não
Falhar graciosamente e suportar a recuperação de erros	12. O app permite a recuperação de erros?	O app permite que o usuário possa facilmente refazer a classificação quando ocorre um erro de classificação (p.ex. manter o botão de tirar foto).	Sim, Não
	13. O app ajuda o usuário a se recuperar de possíveis erros?	O app explica o que fazer quando ocorre um erro de classificação (p.ex. Solicitando tirar outra foto).	Sim, Não
	14. O app indica quando se trata de objetos fora do seu escopo de classificação?	Caso a imagem seja de um objeto não sendo classificado pelo app (p.ex. um copo num app de classificação de cachorros), o app apresenta como resultado a informação que se trata de um objeto fora do escopo de classificação deste app. N/A: caso o app visa classificar qualquer objeto	Sim, Não, N/A
	15. O app mostra um aviso quando o sistema não é capaz de classificar uma foto com confiança suficiente?	Em casos em que não é possível classificar uma foto com confiança suficiente (p.ex < 70%), o app alerta ao usuário informando que não foi capaz de classificar esta foto (ao invés de mostrar resultados mesmo com grau de confiança baixa).	Sim, Não
	16. O app permite que o usuário solicite a verificação do resultado por especialistas humanos?	O app fornece um meio de entrar em contato com especialistas de domínio para verificar o resultado da classificação. N/A: Caso o app não precise da segurança da validação humana.	Sim, Não, N/A
Possibilitar coleta de *feedback*	17. O app permite que usuários com conhecimento no domínio do aplicativo possam enviar *feedback* referente ao resultado da classificação?	O app possibilita que usuários com conhecimento de domínio possam indicar se um resultado de classificação está correto ou não.	Sim, Não, N/A

do usuário		N/A: caso o app se direciona a um público alvo sem conhecimento de domínio	
	18. O app proíbe usuários sem conhecimento no domínio do aplicativo enviem *feedback* referente ao resultado da classificação?	O app está sem a opção de *feedback* de corretude de resultado de classificação no caso em que seus usuários não poderiam fornecê-lo corretamente. N/A: caso o app se direciona a um público alvo com conhecimento de domínio	Sim, Não, N/A
	19. O app deixa claro o propósito de enviar *feedback*?	Caso o app forneça a possibilidade de *feedback*, o app permite que o usuário entenda como seu *feedback* pode afetar o funcionamento, sendo orientado a fornecê-lo com cuidado. N/A: caso que o app não fornece a possibilidade de *feedback*	Sim, Não, N/A
Mitigar viés	20. O app está livre de viés?	A interface do usuário está livre de reforço de viés social, preconceito ou uso de terminologia inapropriada.	Sim, Não
Considerar os riscos ao usuário	21. O app indica os devidos cuidados para tirar fotos?	No caso em que a captura de fotos pode ser arriscada para o usuário (p.ex. tentando tirar foto de uma cobra peçonhenta) são apresentadas diretrizes para fazê-lo de um modo seguro e alertar o usuário sobre o perigo. N/A: caso a captura de fotos pode ser feito sem perigo	Sim, Não, N/A
	22. O app destaca os riscos envolvidos com um possível erro de classificação?	O app indica as consequências de um possível erro de classificação, especificamente em casos em que isso pode resultar em danos físicos a humanos. N/A: caso que erros de classificação não trazem muitos riscos	Sim, Não, N/A
	23. O app mostra elementos visuais de alerta caso o objeto classificado possa causar danos físicos a humanos?	O app mostra de forma visual (p.ex. usando cor e/ou ícone) para alertar o usuário sobre o perigo pelo objeto classificado. N/A: caso o escopo do app não contêm objetos perigosos	Sim, Não, N/A

Resultados. Para a Alternativa 1, o valor de alfa de cronbach foi 0,63 e o de ômega total foi 0,72, na Alternativa 2 foram de 0,57 e 0,77 respectivamente**.** Considerando valores aceitáveis acima de 0,7 [Devellis, 2021; Mcneish, 2018], estes resultados tendem a sugerir que o checklist possui uma consistência interna suficiente, com os coeficientes de ômega total obtidos atingindo a meta.

A correlação policórica para a Alternativa 1 resultou em itens fortemente correlacionados dentro de suas heurísticas, salvo por dois pares, todos os outros apresentaram valores acima do limite de 0,29 de Cohen (2013). A alternativa 2, entretanto, apresentou diversas intercorrelações abaixo do limite e valores negativos, que sugerem que um item não é pontuado caso o par seja. A correlação item total em ambos os casos foi fraca, com apenas a metade dos itens atingindo valores acima de 0,29, e com os menores valores resultando em um aumento na consistência interna se removidos. Essa grande ausência de correlação entre todos os itens, indica a necessidade de revisar ou até mesmo remover itens do checklist.

Já a análise de fatores mostrou-se inconclusiva em geral. O número de dimensões obtidas em ambos os casos foi menor que o número de heurísticas proposto. Em ambos os casos o índice de Kaiser-Meyer-Olkin foi observado maior que 0,5, permitindo a análise fatorial [Brown, 2015]. Nesta a carga dos fatores sugeriu um agrupamento das heurísticas em um subconjunto na Alternativa 1, mas na Alternativa 2 não apresentou semântica alguma. Por último, em ambos os casos, se apenas um fator "AIX" for considerado na análise, a carga dos itens revelou-se baixa, indicando que talvez o *checklist* não seja efetivo na avaliação de AIX e a possível necessidade de reduzi-lo.

5. Desenvolvimento da ferramenta de suporte

Com o objetivo de oferecer um suporte para executar uma avaliação heurística utilizando o checklist proposto, uma ferramenta web foi desenvolvida. O sistema foi implementado utilizando tecnologias web (HTML, CSS e Javascript), e idealizado como um "quiz", em que cada item do checklist é respondido como uma pergunta e ao final o resultado é apresentado.

Como parte do resultado é apresentado uma lista das respostas e o percentual dos itens satisfeitos, calculada a partir da seguinte fórmula, onde n é a quantidade de respostas "Sim" e m a quantidade de respostas "Não":

$$n * 100 / (n + m) \quad (1)$$

Os requisitos da ferramenta foram divididos em requisitos funcionais e não funcionais:

Requisitos Funcionais:

RF1 - Responder o checklist: O sistema deve apresentar cada item do checklist (incluindo: nome, explicação e imagem de exemplo) e a respectiva escala de resposta, permitindo responder cada item. As respostas devem ser armazenadas pelo sistema.

RF2 - Calcular o percentual dos itens satisfeitos: Ao submeter a resposta para o último item do checklist, deve ser calculado o percentual de respostas "Sim" em relação à quantidade total de respostas "Sim" e "Não".

RF3 - Visualizar resultado: Ao submeter a resposta para o último item do checklist, deve ser apresentado uma lista dos itens, indicando de forma visual a resposta (Sim-verde, Não - vermelho, NA-cinza junto com ícones indicativos) e o percentual calculada.

RF4 - Exportar PDF: A apresentação dos resultados deve ser exportável para o formato PDF.

Requisitos Não Funcionais:

RNF1 - Idiomas: O sistema deve estar disponível em inglês e português brasileiro.

RNF2 - O sistema deve ter uma interface estreita para possibilitar a sua visualização em tela dividida com um outra aba do navegador aberta no App Inventor.

RNF3 - Responsividade Mobile: O sistema deve poder ser utilizável nos navegadores de telefones celulares sem a necessidade de scroll horizontal. Botões devem ser grandes o suficiente para serem utilizáveis por toque.

O sistema foi implementado como uma única página web, em que a navegação entre as diferentes telas se dá pela mudança de qual componente está atualmente visível, com um script principal orquestrando a visibilidade dos componentes e mantendo o estado das variáveis usadas na avaliação. A estrutura das telas de pergunta foi especificada em um único ponto, contendo todos os elementos gráficos comuns destas, com a especificação de uma pergunta se dando pela instanciação de uma classe que preenche os campos deste modelo com os dados. A navegação entre perguntas e o armazenamento de respostas foi feito a partir do padrão Observer [Gamma et al., 1995], em que o script principal é um receptor de eventos de submissão de respostas de cada item, e reage armazenando a resposta e incrementando um índice que indica qual pergunta é visível. Quando o último resultado é armazenado, o script navega a uma tela de resultados (Figura 3). Nesta, a resposta de cada pergunta é apresentada com um esquema de cores (Verde - “Sim”, Vermelho - “Não” e Cinza - “N/A” e ícones indicativos) e é apresentado o resultado da avaliação, indicando o percentual dos itens satisfeitos.As informações de resultado podem ser exportadas para PDF com um botão na tela de resultados.

A ferramenta é disponibilizada de forma gratuita pelo site da iniciativa Computação na Escola/INCOD/INE/UFSC: http://apps.computacaonaescola.ufsc.br/aix/.

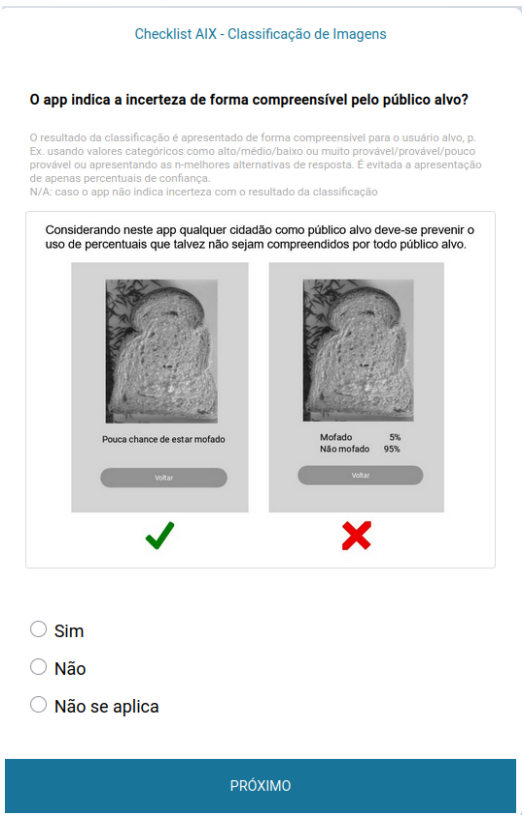


Figura 2 - Tela de resultados da ferramenta

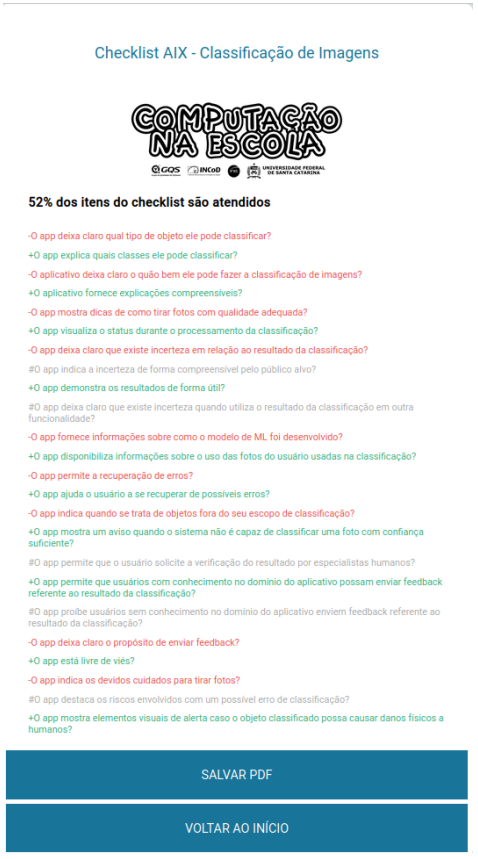


Figura 3 - Tela de resultados da ferramenta

6. Conclusão

A principal contribuição deste trabalho está na elaboração de um conjunto de heurísticas e *checklist* para a avaliação heurística de aplicativos inteligentes de classificação de imagens. Mesmo já existindo outras propostas para AIX, a maioria atualmente é mais voltado para sistemas de recomendação e assim esta customização das heurísticas para classificação de imagens pode ser considerada inédita. Espera-se que a definição destas heurísticas possa ajudar tanto no design de aplicativos deste tipo quanto na sua avaliação, contribuindo desta maneira a melhoria da UX e assim a qualidade dos sistemas.

Como trabalhos futuros, sugere-se a realização de uma análise estatística com um conjunto de dados maior e mais variado, seguido do aperfeiçoamento das heurísticas e *checklist*. Alternativamente, recomenda-se o estudo de uma possível automação da avaliação utilizando o *checklist*.

Referências

Amershi, S. et al. (2019) "Guidelines for human-AI interaction", Proceedings of the CHI Conference on Human Factors in Computing Systems. Glasgow, Scotland Uk, p. 1-13.

Apple (2022) "Human Interface Guidelines", In: https://developer.apple.com/design/human-interface-guidelines/technologies/machine-learning/introduction/

Brown, A. (2015) "Confirmatory factor analysis for applied research", Guilford publications.

Caldiera, V. R., Basili, G. and Rombach, H. D. (1994) "The goal question metric approach". Encyclopedia of software engineering, p. 528-532.

Cohen, J. (2013) "Statistical power analysis for the behavioral sciences". Routledge.

Computação na Escola (2023) "Apps desenvolvidos por jovens no programa PodeCrer do Instituto Pe. Vilson Groh", In: https://computacaonaescola.ufsc.br/appsivg2022/

Cronbach, L. J. (1951) "Coefficient alpha and the internal structure of tests", psychometrika, v. 16, n. 3, p. 297-334.

Dai, X., Spasić, I., Chapman, S. and Meyer, B. (2020) "The state of the art in implementing machine learning for mobile apps: A survey", 2020 SoutheastCon, p. 1-8.

Devellis, R F.; Thorpe, C T. (2021) "Scale development: Theory and applications", Sage publications.

Gamma, E. et al. (1995) "Design patterns: elements of reusable object-oriented software", Pearson Deutschland GmbH.

Google (2022a) "People + AI Guidebook" (2022a), In: https://pair.withgoogle.com/guidebook.

Google (2022b) "Teachable Machine - Google", In: <https://teachablemachine.withgoogle.com/>.

Haddaway, N. R. et al. (2015) “The role of Google Scholar in evidence reviews and its applicability to grey literature searching”, PloS one, v. 10, n. 9, p. e0138237.

ISO/IEC (2010) “ISO 9241-210:2010 Ergonomics of human-system interaction — Part 210: Human-centred design for interactive systems”, In: https://www.iso.org/obp/ui/#iso:std:iso:9241:-210:ed-1:v1:en.

ISO/IEC (2011) “ISO/IEC 25010:2011 Systems and software engineering — Systems and software Quality Requirements and Evaluation (SQuaRE) — System and software quality models”, In: https://iso25000.com/index.php/en/iso-25000-standards/iso-25010.

ISO/IEC. (2020) “ISO 9241-110:2020 Ergonomics of human-system interaction — Part 110: Interaction principles”, In: https://www.iso.org/obp/ui/#iso:std:iso:9241:-210:ed-1:v1:en.

Lawshe, C. H. et al. (1975) “A quantitative approach to content validity”, Personnel psychology, v. 28, n. 4, p. 563-575.

Li, T., Vorvoreanu, M., Debellis, D. and Amershi, S (2022) “Assessing Human-AI Interaction Early through Factorial Surveys: A Study on the Guidelines for Human-AI Interaction”, ACM Transactions on Computer-Human Interaction.

Mcdonald, R. P. (2013), “Test theory: A unified treatment”. psychology press.

Mcneish, D. (2018) “Thanks coefficient alpha, we’ll take it from here”, Psychological methods, v. 23, n. 3, p. 412.

ML+Design (2022) “MACHINE LEARNING + DESIGN”. In: <https://machinelearning.design/>. Acesso em: 08/10/2022

Mohseni, S., Zarei, N. and Ragan, E. D (2021) “A multidisciplinary survey and framework for design and evaluation of explainable AI systems”, ACM Transactions on Interactive Intelligent Systems, v. 11, n. 3-4, p. 1-45.

Nielsen, J. (1994) “Enhancing the explanatory power of usability heuristics”, Proceedings of the SIGCHI conference on Human Factors in Computing Systems. Morristown: New Jersey. p. 152-158.

Petersen, K., Vakkalanka, S. and Kuzniarz, L. (2015) “Guidelines for conducting systematic mapping studies in software engineering: An update”, Information and Software Technology, v. 64, p. 1-18.

Rawat, W. and Wang, Z. (2017) “Deep convolutional neural networks for image classification: A comprehensive review.”, Neural computation, v. 29, n. 9, p. 2352-2449.

Rusu, C. et al. (2011) “A Methodology to establish usability heuristics”, In: Research Gate <https://www.researchgate.net/publication/229040164_A_Methodology_to_establish_usability_heuristics> .

Subramonyam, H., Seifert, C. and Adar, E. (2021) “Towards a process model for co-creating AI experiences”, Designing Interactive Systems Conference. p. 1529-1543.