

# Characterizing the Software Acceptance Testing and the Inclusion of People with Disabilities by Means of a Systematic Mapping

Lívia Fernanda de Oliveira , Cássio Leonardo Rodrigues  and Renato de Freitas Bulcão-Neto 

**Abstract**—Acceptance testing is a test technique where the final user evaluates the actual use of the software. In this test, software must meet the acceptance criteria defined on the requirement engineering phase to be approved. This paper describes the participation of the person with a disability in acceptance testing in terms of testing approaches, tools and user feedback. By performing a systematic mapping, we analyzed 609 articles, of which 58 were considered to answer four research questions. We identified that acceptance tests have no standard and present a dispersed variety of approaches and tools. We also few acceptance tests studies applied to testers with disabilities, although this approach has been rising in recent years. Considering accessibility for all users when including them in acceptance tests can ensure greater reach of users to systems.

**Index Terms**—Software Testing, Acceptance Testing, Disabled User, Systematic Mapping, Secondary Study

## I. INTRODUÇÃO

O teste de software consiste na verificação dinâmica de um software para determinar se ele fornece comportamentos esperados, em um conjunto finito de casos de teste adequadamente selecionados do domínio de execução, geralmente infinito. O teste de software é realizado em diferentes níveis ao longo dos processos de desenvolvimento e manutenção. O teste de aceitação determina se um sistema satisfaz seus critérios de aceitação, geralmente verificando os comportamentos do sistema em relação aos requisitos do cliente, podendo revelar inconformidades não identificadas em testes realizados em outros níveis. No teste de aceitação, o cliente ou o representante do cliente especifica ou realiza diretamente atividades para verificar se seus requisitos foram atendidos [1], [2].

De acordo com o Código de Ética de Engenharia de Software [3], os engenheiros de software devem trabalhar de forma consistente com o interesse público. Em particular, devem considerar questões de deficiência física, alocação de recursos, desvantagem socioeconômica e outros fatores que podem reduzir o acesso aos benefícios do software. Os engenheiros de software devem contribuir para uma sociedade justa, na qual todos os indivíduos tenham oportunidades

iguais de participar ou se beneficiar do uso de software, independentemente de raça, sexo, religião, idade, deficiência ou outros fatores semelhantes. Além disso, no Brasil, o direito da inclusão da pessoa com deficiência é juridicamente assegurado pelo Estatuto da Pessoa com Deficiência [4], contemplando acessibilidade nos sistemas de informação e comunicação. Diante desta demanda, este artigo apresenta um mapeamento sistemático sobre o teste de aceitação, caracterizando as abordagens de teste aceitação e identificando a inclusão da Pessoa com Deficiência no papel de cliente.

A literatura mostra que há estudos sistemáticos com testes de aceitação, como identificado na Seção II. No entanto, esses estudos não abordam a participação da pessoa com deficiência como testadores do software, além de serem estudos que visam explorar domínios específicos. Assim, o objetivo deste trabalho é mapear estudos primários que executaram testes de aceitação, identificando a participação da pessoa com deficiência, abordagens de teste e ferramentas para automação das atividades. Adicionalmente, este trabalho classifica as deficiências dos usuários que participaram dos testes, analisando os métodos de coleta de suas respostas. Assim sendo, a contribuição deste trabalho é ser o primeiro estudo secundário que aborda o teste de aceitação de software com ênfase na inclusão da pessoa com deficiência.

O artigo está organizado da seguinte maneira: a Seção II elenca trabalhos relacionados ao problema proposto ou que descreveram propostas em área próxima a este trabalho, destacando seus objetivos e resultados obtidos; a Seção III descreve os métodos utilizados para a condução do estudo e as etapas definidas para a realização de um mapeamento sistemático; a Seção IV apresenta os resultados obtidos a partir da extração dos dados; a Seção V apresenta as discussões e a Seção VII finaliza com as conclusões.

## II. TRABALHOS RELACIONADOS

No contexto geral da Engenharia de Software, diversos trabalhos têm discutido a inclusão de pessoas com deficiência e grupos minoritários [5] [6] [7] [8] [9], [10]. Silveira e Prikladnicki [5] realizaram um mapeamento sistemático sobre como a diversidade está sendo discutida em Engenharia de Software e Metodologias Ágeis. Eles identificaram 221 artigos qualificados em seu mapeamento sistemático. Esses trabalhos qualificados identificaram Gênero (129), LGBTQI (2), Idade (10), Raça (7), Cultural (67) e Pessoas com Deficiência (1).

Rodríguez *et al.* [6], por meio de uma revisão sistemática da literatura, abordam percepção da diversidade na Engenharia

Lívia Fernanda de Oliveira is with Instituto de Informática, Universidade Federal de Goiás, Goiânia, Goiás, Brasil e-mail: liviafernanda.oliveira@gmail.com.

Cássio Leonardo Rodrigues is with Instituto de Informática, Universidade Federal de Goiás, Goiânia, Goiás, Brasil e-mail: cassio@ufg.br.

Renato de Freitas Bulcão-Neto is with Instituto de Informática, Universidade Federal de Goiás, Goiânia, Goiás, Brasil e-mail: rbulcao@ufg.br.

de Software. Os autores consideram quatro aspectos da diversidade percebida que acreditam ser os mais relevantes para a comunidade de Engenharia de Software: gênero sexual, faixa etária, raça e nacionalidade. Embora a pessoa com deficiência não tenha sido contemplada nos critérios de inclusão da revisão sistemática, os autores destacam a necessidade de incluir esse aspecto em trabalhos futuros sobre diversidade em Engenharia de Software.

Diversos trabalhos abordaram a revisão de literatura em Teste de Software. O mapeamento sistemático realizado por Silveira-Neto *et al.* [7] avaliou o estado da arte sobre pesquisa e prática a respeito de testes em linhas de produto de software. Os autores investigaram quais níveis de teste comumente aplicáveis no desenvolvimento de sistemas únicos também são usados nas abordagens de linhas de produto de software. No entanto, os autores não encontraram evidências sobre o impacto para linhas de produto de software de não realizar um nível de teste específico.

Weiss *et al.* [8] realizaram uma revisão sistemática da literatura com foco em trabalhos de pesquisa empírica que abordam o método de desenvolvimento orientado a testes de aceitação. As questões de pesquisa abordaram aplicações e *frameworks* para o desenvolvimento orientado a testes de aceitação, sendo que inclusão de pessoas com deficiência não foi abordada no trabalho.

Santos *et al.* realizaram uma revisão sistemática para identificar as técnicas de teste de aceitação disponíveis e comparar suas principais características [9], [10]. Em seguida, a revisão embasou a definição de critérios técnicos e legais para a seleção de técnicas de testes de aceitação que podem ser utilizadas em editais de licitação para contratação de software. A revisão sistemática não abordou a inclusão da pessoa com deficiência no teste de aceitação.

Este estudo diferencia-se dos trabalhos apresentados por abordar o teste de aceitação caracterizando: as abordagens para este nível de teste, a inclusão ou exclusão da pessoa com deficiência e ferramentas empregadas para suporte e automação do teste de aceitação. Os resultados do mapeamento sistemático podem embasar a proposição e melhorias de técnicas de teste de aceitação que incluem pessoas com deficiência, por exemplo, pessoas cegas, pessoas surdas e pessoas com deficiência motora.

### III. MÉTODOS

Esta seção apresenta os métodos utilizados para a condução deste trabalho. Serão discutidas as fases que definiram a sua elaboração utilizando questões de pesquisa, critérios PICOC (do Inglês, *Population, Intervention, Comparison, Outcomes and Context*) e critérios de inclusão e exclusão para a seleção inicial dos artigos até a obtenção dos estudos selecionados para a extração de dados.

O mapeamento sistemático é um método de pesquisa que auxilia no desenvolvimento do estado da arte de determinado assunto, organizando os resultados obtidos a partir de estudos primários [11]. O desenvolvimento é conduzido pelas fases de planejamento, condução e publicação (Fig. 1).

Na fase de planejamento, foram definidos o objetivo e o protocolo, que envolvia a definição das questões de pesquisa,

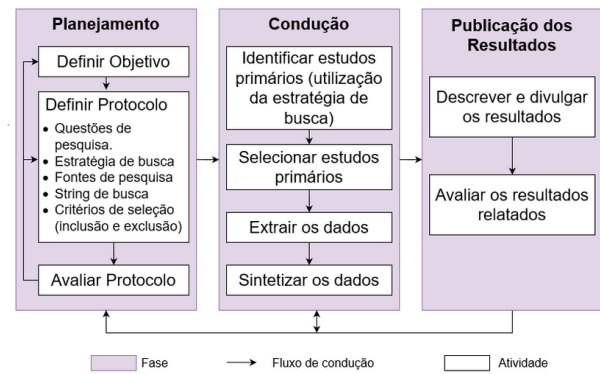


Fig. 1. Fases consideradas para este mapeamento sistemático. Fonte: Adaptação dos autores.

e a estratégia de busca onde seriam incluídas as *strings* de busca. Os estudos selecionados passaram pelos critérios de inclusão e exclusão em duas etapas: a primeira com a leitura dos metadados e a segunda com a leitura completa dos artigos.

A definição dos critérios de qualidade foi opcional, já que essa avaliação não é considerada necessária para mapeamentos sistemáticos, conforme preconizado por Scannavino *et al.* [11].

Na fase de condução, os estudos primários foram identificados e selecionados para que seus dados fossem extraídos e sintetizados a fim de responder às questões de pesquisa propostas na fase de planejamento.

Por fim, os resultados foram descritos e divulgados na fase de publicação dos resultados.

#### A. Questões de Pesquisa

Com o tema sobre testes de aceitação, foram introduzidas as seguintes questões de pesquisa:

**QP1.** Quais as abordagens de teste foram utilizadas nos testes de aceitação?

Esta pergunta busca caracterizar as abordagens de teste de aceitação, as metodologias e os artefatos utilizados no estudo. Esta caracterização é importante para subsidiar a escolha de abordagens de teste de aceitação e também para a proposição de melhorias nas abordagens de modo a contemplar a pessoa com deficiência.

**QP2.** Quais foram os grupos com deficiência contemplados nos estudos de testes de aceitação?

Essa questão visa identificar se de fato está havendo inclusão para todo e qualquer usuário. Assim, busca-se saber se o teste de aceitação é acessível para o usuário testador com deficiência ou se pode ser utilizado para testar sistemas acessíveis.

**QP3.** Quais as ferramentas utilizadas no teste de aceitação?

A automação de atividades de teste é imprescindível para a produtividade e qualidade da Engenharia de Software. Assim, as oportunidades de inclusão da pessoa com deficiência também devem ser caracterizadas em termos das ferramentas utilizadas nos testes de aceitação.

**QP4.** Como os relatórios de teste de aceitação são apresentados?

Um resultado importante do teste de aceitação é o relatório de teste, que sumariza os resultados da atividade. Esta questão

visa avaliar a maneira pela qual o usuário comunica as respostas dos testes de aceitação por ele realizados. Também procura-se identificar como o usuário com deficiência é contemplado nas devolutivas do teste de aceitação.

### B. Critérios PICOC

O conjunto de critérios PICOC, derivado de pesquisas na área da medicina, auxilia a formulação da *string* de busca, e é representado pelo acrônimo de População, Intervenção, Comparação ou Controle e *Outcome* (termo em inglês que significa resultados), além do termo Contexto, inserido para auxiliar as pesquisas em Engenharia de Software [11].

Para a estruturação das questões de pesquisa, foram definidos como critérios PICOC:

- População: testes de aceitação investigados nas literaturas científicas publicadas nas principais fontes de pesquisa.
- Intervenção: processos de software que derivam e utilizam os testes de aceitação para avaliar os produtos gerados para determinada população de usuários.
- Comparação: não se aplica neste estudo por se tratar de um mapeamento sistemático [11].
- Resultados: técnicas e ferramentas de teste de aceitação a serem aplicados ao usuário.
- Contexto: avaliação dos estudos realizados sobre os testes de aceitação.

A partir da leitura de artigos que referenciavam esta área e dos critérios PICOC, as palavras-chave *acceptance test*, *software process*, *technique* e *tool* foram consideradas, incluindo o sinônimo *acceptance testing*, o que levou à *string* de busca final:

("acceptance test" OR "acceptance testing") AND ("software process" OR "techniques" OR "tools").

A *string* de busca foi inserida nas fontes ACM Digital Library, IEEE Xplore e Engineering Village no dia 14 de julho de 2020. Para cada fonte de busca, houve uma adaptação na *string* de busca, aplicando filtros próprios de cada uma delas. Os metadados dos artigos foram extraídos em formato *bibtex* e inseridos na ferramenta Parsifal [12], onde puderam ser analisados de forma mais organizada e criteriosa.

Os estudos duplicados foram identificados e removidos e os demais passaram por duas etapas pelos critérios de seleção: na primeira com a leitura dos metadados e na segunda com a leitura do estudo na íntegra.

### C. Critérios de Inclusão e Exclusão

O critério que considerou o estudo elegível para este mapeamento sistemático é que “proponha ou relate a aplicação de teste de aceitação para processos de software”.

Os critérios de exclusão considerados foram:

**CE1.** O estudo não relata aplicação de teste de aceitação de software, sistemas computacionais ou na área de engenharia de software;

**CE2.** O estudo não possui resumo;

**CE3.** O estudo possui apenas o resumo;

**CE4.** O estudo não está disponível para consulta;

**CE5.** O estudo não está escrito em inglês ou em português ou foi publicado antes de 2010;

**CE6.** O estudo não é um estudo primário;

**CE7.** O estudo é uma repetição de outro já publicado;

**CE8.** O estudo não é um artigo;

**CE9.** O estudo apenas cita o processo de teste de aceitação, mas não apresenta nenhuma técnica para especificar os cenários de teste;

**CE10.** O processo cita o teste de aceitação, porém não é na área de engenharia de software ou sistemas computacionais.

### D. Extração de Dados

A fase seguinte foi a condução da extração de dados considerando o campo do formulário de extração de dados. Este formulário comportava os seguintes campos:

- 1) ID do estudo
- 2) Ano de publicação
- 3) Proposta do estudo
- 4) Critérios do teste de aceitação
- 5) Testes executados relacionados ao processo
- 6) Identificação dos artefatos
- 7) Abordagem(ns) do teste de aceitação
- 8) Testes de aceitação automatizados
- 9) Ferramenta(s) utilizada(s) no teste de aceitação
- 10) Objetivos do teste de aceitação
- 11) Metodologia de coleta de respostas dos usuários
- 12) Resultados
- 13) Grupo de usuários avaliados

O campo do item 8 é de única escolha, assumindo os valores Sim ou Não. Já o campo do item 10 é de múltipla escolha, podendo assumir os seguintes valores:

- Adaptação dos requisitos
- Execução de teste de regressão
- Validação com os requisitos
- Validação de software
- Demonstração de nova abordagem de teste de aceitação

O campo do item 13 era de múltipla escolha, podendo assumir os seguintes valores:

- Usuário cego
- Usuário com deficiência de atenção e aprendizado
- Usuário com deficiência mental
- Usuário com deficiência motora
- Usuários surdo
- Usuários não técnicos
- Usuários técnicos

Os demais campos eram caixas de texto, onde as respostas foram descritas.

## IV. RESULTADOS

Inicialmente foram encontrados 684 artigos, sendo que os mesmos estavam distribuídos da seguinte forma nas bases consultadas:

- 398 artigos pela ACM Digital Library;
- 212 artigos pela Engineering Village;
- 74 artigos pela IEEE Xplore.

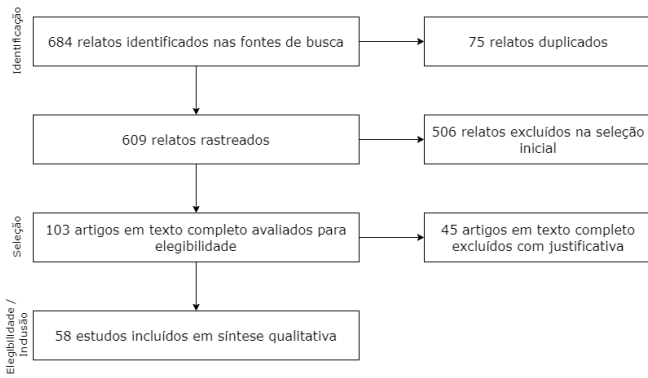


Fig. 2. Quantidade de artigos nas etapas de critérios de seleção do processo de mapeamento sistemático. Fonte: Adaptação dos autores.

A Fig. 2 sumariza a quantidade de artigos das diferentes etapas de seleção do mapeamento sistemático. Dentre os 684 artigos, 75 estavam duplicados e foram excluídos, restando 609 artigos que passaram pela primeira etapa dos critérios de inclusão e exclusão definidos.

Na seleção foram avaliados os metadados (título e resumo) e aplicados os critérios de inclusão e exclusão a cada um dos artigos. O artigo era incluído ao obedecer ao critério de inclusão e excluído ao considerar pelo menos um critério de exclusão. Nesta etapa foram excluídos 506 artigos, sendo: 454 artigos excluídos pelo CE1, 8 pelo CE6, 8 pelo CE8 e 36 pelo CE9. A quantidade de artigos excluídos por critério de exclusão durante todo o processo de seleção pode ser observado na Tabela I. Ao final foram incluídos 103 artigos para aplicação da extração de dados durante a leitura do conteúdo.

Durante a fase de extração de dados, os artigos foram lidos e analisados integralmente. No total, mais 45 artigos foram excluídos segundo os critérios de exclusão definidos, sendo que dois artigos foram excluídos pelo critério CE3, três pelo critério CE4, um pelo critério CE5, um pelo critério CE8, trinta e três pelo critério CE9 e três pelo critério CE10 (Tabela I).

Como prova de conceito, durante a busca foram encontrados os artigos: “Automated Acceptance Tests as Software Requirements: An Experiment to Compare the Applicability of Fit Tables and Gherkin Language” [13] e “A systematic literature review to support the selection of user acceptance testing techniques” [14]. Tais artigos foram utilizados como base para montar a *string* de busca. Porém, este último não foi incluído neste mapeamento por se tratar de um estudo secundário, violando a CE6.

Inicialmente a *ACM Digital Library* foi a fonte que mais contribuiu com artigos relevantes, sendo 58,2% antes da exclusão dos artigos. Após a aplicação dos critérios de seleção, a fonte mais relevante passou a ser o *Engineering Village*, com 55% dos artigos relevantes. A relação dos artigos relevantes selecionados encontra-se ao final (Tabela VI).

## V. DISCUSSÃO

A partir da extração de dados, pode-se selecionar os pontos mais importantes para responder às questões de pesquisas

elaboradas e realizar discussões sobre os resultados obtidos.

### QP1. Quais as abordagens de teste foram utilizadas nos testes de aceitação?

Para responder à QP1, Quais as abordagens de teste de aceitação foram empregadas?, foi feita uma análise dos artigos considerando a metodologia, a abordagem e os artefatos utilizados em cada estudo. As abordagens foram variadas e não foi possível identificar uma padronização nas propostas de teste de aceitação. Neste caso, os estudos foram organizados conforme a Tabela II. A primeira coluna considera o contexto em que o teste de aceitação foi abordado. A segunda coluna caracteriza o princípio abordado pelo estudo. A terceira coluna registra os estudos identificados.

Cinco contextos principais foram identificados dentre todos os trabalhos. O contexto que mais abrange estudos é a Aplicação de Teste de Aceitação, com 31 estudos. Assim, 53,45% dos estudos tratam da aplicação do teste de aceitação em algum software, mas sem apresentar uma proposta de teste de aceitação propriamente e somente mencionam que o teste de aceitação foi aplicado no trabalho. Nos quatro contextos restantes, os autores apresentam abordagens de teste de aceitação propriamente. A Geração de Casos de Teste foi abordada em 10 estudos. A Evolução e/ou refatoração de teste de aceitação foi tratada em cinco estudos. A Automação do Teste de Aceitação foi o contexto de nove estudos. Por último, o Gerenciamento de Teste de Aceitação englobou três estudos.

Dos 31 estudos que mencionam a aplicação do teste de aceitação, 15 deles aplicam o teste de aceitação usando uma avaliação pelo usuário em termos de valores dentro de uma escala Likert. No contexto da pessoa com deficiência, a escala Likert pode requerer desenvolvimento específico para se tornar adequada para um determinado grupo. Nos trabalhos de Chaveiro *et al* [15], [16], com foco na avaliação da qualidade de vida da pessoa surda, a simples tradução de português para Língua Brasileira de Sinais (Libras) de uma escala Likert resultou na representação de valores diferentes da escala por um mesmo sinal em Libras. Para diferenciar os valores da escala em Libras, os autores identificaram outros sinais explorando grupos focais. A quantidade de itens da escala Likert também pode influenciar sua aplicação em determinados grupos. Fang *et al* [17] identificaram que uma escala de três itens pode ser utilizada no lugar de uma com cinco itens para avaliação da qualidade de vida de pessoas com deficiência intelectual. Nos estudos analisados, os autores não mencionam adaptações na escala Likert para se adequar às demandas de acessibilidade do grupo.

No contexto da Automação do Teste de Aceitação, o estudo S50 explora requisitos de acessibilidade no teste de aceitação. O estudo consiste em um conjunto de ações executáveis de testes de aceitação que simulam a navegação por teclado em aplicações web. Esse conjunto de ações é usado para testar a interface de acordo com o modelo de interação para usuários com deficiência, realizando estratégias de navegação básicas e padrão, tais como navegação por tabulação e teclas de seta, em Aplicações de Internet Rica.

### QP2. Quais foram os grupos com deficiência contemplados nos estudos de testes de aceitação?

TABELA I

NÚMERO TOTAL DE ESTUDOS REMOVIDOS POR CRITÉRIO DE EXCLUSÃO AO LONGO DE TODA A FASE DE CONDUÇÃO.  
 FONTE: PRÓPRIA.

Atividade	CE1	CE2	CE3	CE4	CE5	CE6	CE7	CE8	CE9	CE10	Total
Busca automática	454	0	0	0	0	8	0	8	36	0	506
Extração de dados	0	0	2	3	1	0	0	1	35	3	45
<b>Total</b>	<b>454</b>	<b>0</b>	<b>2</b>	<b>3</b>	<b>1</b>	<b>8</b>	<b>0</b>	<b>9</b>	<b>71</b>	<b>3</b>	<b>551</b>

TABELA II

ABORDAGENS ENCONTRADAS NOS ESTUDOS ANALISADOS.  
 FONTE: PRÓPRIA.

Contexto em que o TA é abordado	Princípio abordado	Estudos
Aplicação de Teste de Aceitação.	Avaliação do software testado usando escala Likert	S01, S02, S04, S12, S14, S16, S18, S19, S21, S25, S26, S36, S41, S43, S51
	Opinião do usuário sobre o software	S15, S31, S34, S39, S49, S56
	Casos de uso do software	S17, S23, S27, S53
	Cenários de aplicação do software	S03, S06, S09, S20
Geração de Casos de Teste	Tarefas para computação de alto desempenho	S08
	Desenvolvimento ágil.	S07
	Casos de Uso	S40, S42, S57
	Especificação Gherking e/ou FitTables	S22, S30
	Linhas de produto de software	S11
	Diagrama de interação com o usuário	S37
Evolução e/ou refatoração de teste de aceitação	Indução matemática para o MapReduce	S05
	Árvores de utilidade	S29
	Regras de negócio	S44
Automação do Teste de Aceitação	Processamento de linguagem natural	S24, S28
	Evolução de código de teste	S47, S54
	Palavras chaves	S10
Gerenciamento de Teste de Aceitação	Diagrama de interação com o usuário	S32
	Interface gráfica do software	S13, S33, S45
	Processamento de linguagem natural	S46
	Requisitos de segurança	S35
	Linguagem específica de domínio	S38
	Casos de uso	S48
Planejamento de teste de aceitação	Análise de acessibilidade de software	S50
	Avaliação de feedback de teste de aceitação	S52
	Avaliação de performance de teste de aceitação	S55
Planejamento de teste de aceitação	S58	

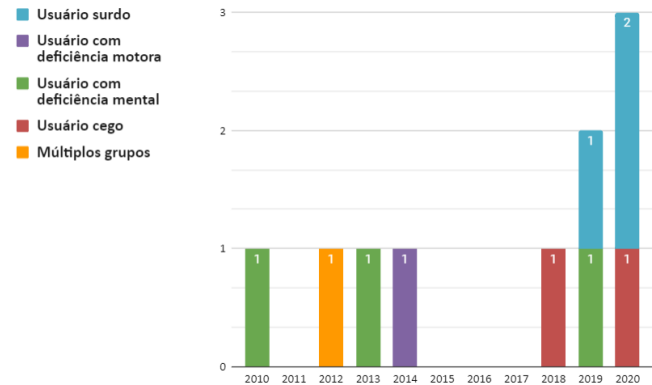


Fig. 3. Estudos de testes de aceitação realizados com os grupos de usuários com deficiência entre os anos de 2010 a 2020. Fonte: Própria.

O mapeamento sistemático identificou 4 grupos de usuários com deficiência: usuários cegos, usuários surdos, usuários com deficiência mental e usuários com deficiência motora. O perfil dos testadores foi variado dentro de cada grupo específico, contando com voluntários totalmente cegos, parcialmente cegos, voluntários totalmente surdos e/ou mudos, crianças com TDAH, crianças com Síndrome de Down, idosos com doença de Parkinson, além de voluntários sem deficiência que também participaram dos testes ou auxiliando os testadores, como intérpretes de língua de sinais, parentes e professores acompanhantes.

Os estudos S01, S03, S04, S14, S15, S19, S43, S49, S50 e S56 relataram testes de aceitação envolvendo grupos de testadores com deficiência. Ao longo do tempo, foi possível perceber que houve uma maior realização de estudos de testes de aceitação com estes grupos nos últimos dois anos (Fig. 3).

Dentre os trabalhos que contemplaram o usuário com deficiência, apenas o estudo S50 abordou a automatização da execução do teste de aceitação. Os demais estudos não foram automatizados e os usuários testadores participaram ao longo de todo o processo de teste de aceitação.

### QP3. Quais as ferramentas utilizadas no teste de aceitação?

As ferramentas utilizadas nos estudos também sofreram ampla variação (Tabela III). Muitos artigos buscaram experimentar novas ferramentas ou combinações de várias delas, cada estudo com seu propósito no teste de aceitação. Entretanto, 28 estudos não informaram ou não utilizaram uma ferramenta específica para o teste de aceitação. Por esse motivo receberam a denominação "Não especificado". As ferramentas mais utilizadas foram o "Cucumber" e o Fitness, aparecendo

TABELA III  
FERRAMENTAS ENCONTRADAS NOS ESTUDOS  
ANALISADOS. FONTE: PRÓPRIA.

Ferramentas identificadas	Estudos
MUCAT	S35
UCAT	S57
Cenários US-UID	S37
Cucumber	S11 S21 S30 S46
Fitness	S21 S32 S35 S57
Selenium IDE	S33
JUnit	S32
Selenium	S32
GuideGen	S24 S28
InterMod	S07
JAutomate	S45
Legend	S38
Harmony	S08
Pyccuracy	S50
ReDSeT	S40
ReFit	S47 S54
TaRGeT	S42
Test-duo	S48
ROBOT	S48
Acceclipse	S48
Ukwikora	S10
Plugin elaborado para o Eclipse	S58
Planilhas Eletrônicas	S17

em quatro estudos. As ferramentas "GuideGen" e "ReFit" apareceram em dois estudos cada um.

Outros estudos demonstraram combinações de ferramentas de teste, como "Fitness", "JUnit" e "Selenium" no estudo S32, "Cucumber" e Fitness no estudo S22 e "Test-Duo", "ROBOT" e "Acceclipse" no estudo S48.

Encontrou-se também combinações de ferramentas diversas para facilitar o uso da ferramenta de teste de aceitação, como foi o caso do estudo S24, utilizando "Text-Diff" e "SyntaxNet" como ferramentas de verificação de caracteres para auxiliar a ferramenta "GuideGen", utilizada na automatização do teste de aceitação.

Houve estudos apresentando novas ferramentas para testes de aceitação, como o "Ukwikora" (S10), o editor de diagrama de interação de usuário "US-UID", para a criação e execução de cenários de testes (S32) e o "GuideGen" (S24).

Houve ainda adaptação de ferramenta para testes de aceitação, como o "Pyccuracy" que foi uma espécie de adaptação a partir do Selenium (S50).

#### QP4. Como os relatórios de teste de aceitação são apresentados?

Muitos estudos analisados demonstraram que a coleta dos resultados foi feita por "questionários de pesquisa" e "formulários em escala *likert*", sendo cada um deles ocorrendo em 10 estudos. "Coleta de *feedback* com sugestões e entrevistas com o usuário" apareceu em dois estudos cada. Nestes casos os testadores usuários validaram os sistemas pela interação direta de suas respostas e conclusões (Tabela IV).

Nos testes automatizados os resultados finais eram obtidos a partir da análise dos "resultados dos registros gerados pelo código ou pelo sistema em execução" junto com a comparação com os resultados esperados. Este tipo de coleta ocorreu em 24 estudos.

TABELA IV  
MÉTODOS DE COLETA DE RESULTADOS DOS TESTES DE  
ACEITAÇÃO. FONTE: PRÓPRIA.

Coleta de respostas	Estudos
Coleta de feedback com sugestões	S20 S56
Confirmação da realização da tarefa de negócio no caso de teste	S44
Entrevista com o usuário	S28 S39
Formulário de avaliação com sugestões	S32
Formulário de avaliação e validação em vídeo	S04
Método de estímulos constantes	S31
Não especificado	S03
Questionário de pesquisa	S2 S09 S12 S13 S15 S22 S23 S24 S37 S49
Questionário de pesquisa, conversação e observação	S29
Questionário escala likert	S01 S14 S16 S18 S21 S25 S26 S41 S43 S51
Questionário escala likert com folha de resposta especial	S19
Questionário padronizado para cálculo da escala de usabilidade de sistemas	S36
Reporte verbal (CR) e formulário documentado (FS)	S52
Resultados dos registros gerados pelo sistema	S05 S06 S07 S08 S10 S11 S17 S30 S32 S33 S35 S38 S40 S42 S45 S46 S47 S48 S50 S53 S54 S55 S57 S58
Tabela de escore pass/fail	S27

Na avaliação dos estudos realizados com usuários com deficiência, foi identificado "formulário de avaliação e validação em vídeo" para usuários surdos (S4), porém os vídeos eram mais direcionados para as instruções de uso do software do que na execução dos testes.

Houve também um estudo que definiu a coleta em um "questionário de escala *likert* com folha de resposta especial" para usuários cegos (S19). Porém os autores do estudo não detalharam o que seria esta folha de resposta especial.

Outros estudos realizados envolvendo pessoas com deficiência apresentaram os seguintes métodos de coletas de respostas: "resultados de registros gerados pelo sistema" (S50), "coleta de *feedback* com sugestões" (S56), "questionário de pesquisa" (S15, S49), "questionário em escala *likert*" (S1, S14, S43).

Apenas um dos estudos realizado com usuários surdos não especificou a maneira como esses dados foram coletados (S3), além de ter sido um estudo pouco detalhado em relação ao teste de aceitação aplicado. Neste caso a coleta foi caracterizada como "não especificado".

#### A. Cruzamento dos Resultados das Questões de Pesquisa da Perspectiva de Inclusão

Esta seção apresenta um relacionamento entre as respostas das questões de pesquisa, mas considerando somente os trabalhos que realizaram o teste de aceitação caracterizando a inclusão da pessoa com deficiência, conforme Tabela V. É importante destacar que não se pode afirmar a ausência de participantes com deficiência nos demais estudos. Dependendo da severidade da deficiência do usuário, ele pode ter participado ou ter sido contemplado indiretamente no teste de aceitação.



Os estudos S04, S19 e S50 abordam diretamente a acessibilidade no teste de aceitação. Isto porque os trabalhos apresentam algum tratamento específico para inclusão do usuário com deficiência na coleta de respostas. O estudo S04 apresenta vídeos em língua de sinais sobre as instruções de uso e de avaliação do sistema. O estudo S19 provê folhas de respostas especiais para os estudantes cegos. O estudo S50 aborda a automação do teste de aceitação com requisitos de acessibilidade. O estudo atende majoritariamente o grupo de usuários cegos, podendo dar o suporte a inclusão de qualquer outro grupo que precise utilizar o software sem a manipulação de um mouse, por exemplo, via navegação por teclado.

Por considerarem a acessibilidade diretamente no teste de aceitação, os estudos S04, S19 e S50 permitem a participação do usuário com deficiência no teste. No caso dos demais estudos, o teste de aceitação é aplicado sem acessibilidade ao usuário com deficiência ou é aplicado a algum grupo de pessoas que não possuem a deficiência, mas que está associado ao usuário, como familiares e professores. Uma lacuna de tais trabalhos é falta de discussão sobre a efetividade do teste de aceitação desta forma.

Outro ponto pouco discutido nos estudos identificados é ausência de referenciais de acessibilidade para o desenvolvimento do software e para o teste de aceitação. Isto dificulta identificar relações entre a deficiência do usuário submetido ao teste com o a abordagem aplicada a ele em relação forma de coleta de dados. Somente o estudo S50 declara o documento de referência para se alcançar a acessibilidade. Tal estudo é baseado no WCAG (*Web Content Accessibility Guidelines*) [18], que estabelece diretrizes para acessibilidade na web. As diretrizes também fornecem critérios objetivos e testáveis, que podem ser avaliados com uma combinação de testes automáticos e avaliação por seres humanos. Por ser baseado em critérios de acessibilidade que são objetivos e testáveis, o estudo S50 tem como outro benefício a proposição da ferramentas para automação do teste de aceitação inclusivo.

### *B. Necessidade da Inclusão de Pessoas com Deficiência nos Processos de Software*

Dos 10 estudos encontrados que contemplaram grupos de usuários com deficiências nos testes de aceitação, 9 abordaram o contexto de Aplicação de Teste de Aceitação (Tabela V). Nestes estudos, os usuários com deficiência não profissionais na área de tecnologia participaram ativamente dos testes, avaliando e/ou validando os sistemas desenvolvidos para eles. A presença de usuários testadores com deficiência nestes contextos pode tornar a validação mais autêntica devido à experiência deste usuário estar mais alinhada à realidade para qual o sistema se destina.

O trabalho de Silva *et al.* [19] avaliou a presença de usuários surdos na fase de engenharia de requisitos, o que possibilitou o desenvolvimento de sistemas Web acessíveis para estes usuários. Este fato reforça a importância da inclusão de usuários e/ou profissionais com deficiência nas etapas de desenvolvimento de software, tendo em vista que a acessibilidade é uma realidade cada dia mais presente em diversos ambientes sociais.

Nos estudos analisados, entretanto, não se verificou a presença de profissionais da área de tecnologia com deficiência participando da fase de testes ou em qualquer outra fase.

## VI. AMEAÇAS À VALIDADE

Embora o estudo tenha seguido os critérios de um mapeamento sistemático, pode-se considerar que há ameaças a sua validade. As ameaças de design para a validade do construto abrangem questões relacionadas ao projeto do experimento e sua capacidade de refletir o construto a ser estudado [77]. Neste contexto, uma ameaça de validade deste trabalho é a impossibilidade de garantir que todos os trabalhos de teste de aceitação foram incluídos no mapeamento. Para lidar com esta ameaça, a string de busca foi montada de forma iterativa e incremental para se escolher os termos-chave e operadores lógicos compatíveis com conteúdos disponíveis nos metadados dos artigos. Adicionalmente, os artigos [13] e [14] foram utilizados como artigos de controle para ajuste da string de busca. Considera-se também como ameaça à validade deste estudo um número pequeno de fontes pesquisadas. Apesar de terem sido utilizadas três fontes, *ACM Digital Library*, *Engineering Village* e *IEEE Xplore*, as duas primeiras são fontes híbridas contendo artigos em repositórios de múltiplas editoras e associações científicas. As três fontes se justificam pelo direcionamento específico nas áreas de tecnologia, engenharia de software e processos de software que elas oferecem, o que acarretou em resultados apropriados para esta pesquisa. De todo modo, o protocolo do mapeamento pode ainda ser enriquecido com a técnica de *snowballing* [78] sobre os 58 estudos resultantes.

As ameaças de validade interna contemplam as influências que ameaçam a conclusão sobre uma possível relação causal entre tratamento e resultado [77]. Uma ameaça de validade interna é a ausência do teste de sensibilidade. Uma análise de sensibilidade é uma parte importante de uma meta-análise, pois visa determinar a robustez dos resultados observados de acordo com as suposições feitas na realização da análise [79]. A variação de terminologia nos trabalhos e a diferença dos perfis das pessoas no processo da revisão sistemática podem dificultar a aplicação dos critérios de inclusão e exclusão e da extração das informações. Neste sentido, para identificar dúvidas, ambiguidades e contradições, as etapas realizadas foram validadas por um terceiro revisor. Os problemas identificados foram resolvidos com a participação de todos os autores.

Apesar da QP2 buscar testes de aceitação realizados por usuários com deficiências, a *string* de busca não foi adaptada para esta abrangência, o que representa uma ameaça à validade. Entretanto, ao tentar inserir, palavras como *disability*, *deaf user* ou *blind user* nas pesquisas, a string de busca para testes de aceitação retornou uma quantidade insuficiente de artigos. Portanto, foi necessária uma busca mais generalizada e, dentro dos resultados obtidos, realizar uma pesquisa criteriosa de estudos almejados. Outra ameaça é a dificuldade de se caracterizar o participante do teste de aceitação como pertencente a um grupo com deficiência. Dependendo da severidade da deficiência do usuário, ele pode

TABELA V

RESULTADOS DE TODAS AS QUESTÕES PARA TRABALHOS COM INCLUSÃO DA PESSOA COM DEFICIÊNCIA. FONTE: PRÓPRIA.

ID	Grupo	Contexto	Princípio	Ferramenta	Coleta de respostas
S01	Usuário cego	Aplicação de TA	Avaliação usando escala Likert	-	Questionário com escala Likert
S03	Usuário surdo	Aplicação de TA	Cenários de aplicação do software	-	-
S04	Usuário surdo	Aplicação de TA	Avaliação usando escala Likert	-	Formulário com questões em vídeo
S14	Usuário surdo	Aplicação de TA	Avaliação usando escala Likert	-	Questionário com escala Likert
S15	Usuário com deficiência mental	Aplicação de TA	Opinião do usuário	-	Questionário de pesquisa
S19	Usuário cego	Aplicação de TA	Avaliação usando escala Likert	-	Escala Likert especial
S43	Usuário com deficiência motora	Aplicação de TA	Avaliação usando escala Likert	-	Questionário com escala Likert
S49	Usuário com deficiência mental	Aplicação de TA	Opinião do usuário	-	Questionário de pesquisa
S50	Múltiplos grupos	Automação do TA	Análise de Acessibilidade	Pyccuracy	Registros gerados pelo sistema
S56	Usuário com deficiência mental	Aplicação de TA	Opinião do usuário	-	Coleta de feedback sugestões

TABELA VI

ESTUDOS PRIMÁRIOS SELECIONADOS. FONTE: PRÓPRIA.

ID	Título do artigo
S01	Talkie: An Assistive Web-based Educational Application Using Audio Files and Speech Technology for the Visually Impaired [20]
S02	e-ALEAP: Online Reviewer System for Alternative Learning Equivalency and Accreditation Program [21]
S03	Using Online Handwritten Character Recognition in Assistive Tool for Students with Hearing and Speech Impairment [22]
S04	TEXT2FSL: A Filipino Sign Language Phrase Translator Tool for Deaf and Mute [23]
S05	Testing MapReduce program using Induction Method [24]
S06	Evaluation framework for automatic privacy auditing tools for hospital data breach detections: A case study [25]
S07	Improving Agile Software Development Methods by means of User Objectives: An End User Guided Acceptance Test-Driven Development Proposal [26]
S08	Harmony: A Harness Monitoring System for the Oak Ridge Leadership Computing Facility [27]
S09	Perfect Match: Facilitating Study Partner Matching [28]
S10	Ukwikora: Continuous inspection for keyword-driven testing [29]
S11	SPL-AT Gherkin: A gherkin extension for feature oriented testing of software product lines [30]
S12	Application of Collision Detection Algorithm and Scoring Health Point in Fighting Games with Android-Based Augmented Reality Technology [31]
S13	Augmented testing: Industry feedback to shape a new testing technology [32]
S14	BridgeApp: An Assistive Mobile Communication Application for the Deaf and Mute [33]
S15	A gamified learning app for children with ADHD [34]
S16	The User Acceptance Test of An "ICT Adoption for Education" Framework [35]
S17	Framework Research on the Implementation of Automated Test User Requirements [36]
S18	Web based expert system for diagnosis of cattle disease [37]
S19	PINDOTS: An Assistive Six-dot Braille Cell Keying Device on Basic Notation Writing for Visually Impaired Students with IoT Technology [38]
S20	In-Patient Medication Delivery in Mobile App and Out-Patient Online Lab Results for Hospitals [39]
S21	Automated Multi-factor Analytics for Cheat-Proofing Attendance-taking [40]
S22	Automated acceptance tests as software requirements: An experiment to compare the applicability of fit tables and Gherkin language [13]
S23	E-commerce development using AngularJS framework and RESTful API [41]
S24	GuideGen - A Tool for Keeping Requirements and Acceptance Tests Aligned [42]
S25	e-RICE: An Expert System using Rule-Based Algorithm to Detect, Diagnose, and Prescribe Control Options for Rice Plant Diseases in the Philippines [43]
S26	Development of Industry Academe Linkage Alumni and Placement Portal [44]
S27	Event management solution using web application platform [45]
S28	Aligning requirements and acceptance tests via automatically generated guidance [46]
S29	Deriving high-priority acceptance test cases using utility trees: A case study [47]
S30	SS-BDD: Automated Acceptance Testing for Spreadsheets [48]
S31	SERF: optimization of socially sourced images using psychovisual enhancements [49]
S32	A web framework for test automation: User scenarios through user interaction diagrams [50]
S33	Test driven development of web applications: A lightweight approach [51]
S34	Development of an integrated e-health tool for people with, or at high risk of cardiovascular disease. [52]
S35	Using security robustness analysis for early-stage validation of functional security requirements [53]
S36	Component-based workspace awareness support for composite web applications [54]
S37	Creating user scenarios through user interaction diagrams by non-technical customers [55]
S38	Legend: an agile DSL toolset for web acceptance testing [56]
S39	Evaluation of a guidelines-based e-health decision support system for primary health care in South Africa [57]
S40	Automating acceptance testing with tool support [58]
S41	Guiding novice database developers in database schema creation [59]
S42	Comparing model-based testing with traditional testing strategies: An empirical study [60]
S43	Improving mouse controlling and movement for people with parkinsons disease and involuntary tremor using adaptive path smoothing technique via B-spline [61]
S44	Business-driven acceptance testing methodology and its practice for e-government software systems [62]
S45	JAutomate: A tool for system- and acceptance-test automation [63]
S46	Automatically generating tests from natural language descriptions of software behavior [64]
S47	ReFit: A Fit test maintenance plug-in for the Eclipse refactoring plug-in [65]
S48	Test-Duo: A framework for generating and executing automated acceptance tests from use cases [66]
S49	"SynMax": A mathematics application tool for Down syndrome children [67]
S50	Using acceptance tests to validate accessibility requirements in RIA [68]
S51	Online tutoring system in college: Case study in private education [69]
S52	Supporting acceptance testing in distributed software projects with integrated feedback systems: Experiences and requirements [70]
S53	Explicit use-case representation in object-oriented programming languages [71]
S54	Automated acceptance test refactoring [72]
S55	An industrial case study on speeding up user acceptance testing by mining execution logs [73]
S56	Game-based learning courseware for children with learning disabilities [74]
S57	Developing comprehensive acceptance tests from use cases and robustness diagrams [75]
S58	Pattern-based generation of test plans for open distributed processing systems [76]



pode ter participado ou ter sido contemplado indiretamente no teste de aceitação.

Ameaças à validade externa são condições que limitam a capacidade de generalizar os resultados de um experimento para fora do contexto em que ele foi executado [77]. A acessibilidade é assunto de interesse de outras áreas da Engenharia de Software tais como Engenharia de Requisitos [19] e Experiência do Usuário. A inclusão de estudos em tais áreas pode constituir um problema de coesão de assuntos de modo a dificultar que os resultados sejam utilizados no desenvolvimento de propostas para o teste de aceitação. Desta forma, este trabalho concentrou-se somente em estudos sobre o teste de aceitação. Ressalta-se que testes de aceitação são utilizados nas mais diversas áreas de conhecimento. Este fato demandou um maior esforço para identificar os estudos mais apropriados para o tema, contando com uma rigorosa extração de dados obtida por um formulário estruturado.

Por fim, as ameaças à validade da conclusão estão relacionadas com questões que afetam a capacidade de tirar a conclusão correta sobre as relações entre o tratamento e o resultado de um experimento [77]. A respeito desta ameaça, o trabalho enfatizou o risco de que a implementação do mapeamento não fosse semelhante entre diferentes pessoas que a aplicam ou entre diferentes ocasiões. As atividades do mapeamento foram realizadas em conformidade com os protocolos de mapeamento sistemático e também por meio de ferramentas para auxiliar o processo.

## VII. CONCLUSÃO E TRABALHOS FUTUROS

Tendo como foco a caracterização da inclusão de pessoas com deficiência na teste de aceitação de software, este mapeamento sistemático analisou inicialmente 609 artigos, dos quais 58 artigos foram considerados para responder quatro questões de pesquisa. As questões de pesquisa e os termos de busca foram definidos de modo a caracterizar o teste de aceitação no contexto geral e também em relação à inclusão da pessoa com deficiência. Verificou-se que os trabalhos de testes de aceitação não apresentam padronização consolidada, já que vários estudos demonstraram abordagens e ferramentas distintas. No entanto, foi possível identificar cinco contextos principais nos quais o teste de aceitação foi abordado. O contexto que mais abrange estudos é a Aplicação de Teste de Aceitação, mas sem apresentar uma proposta de teste de aceitação propriamente. Nos contextos restantes, os autores apresentam abordagens de teste de aceitação tratando da: Geração de Casos de Teste, Evolução e/ou refatoração de teste de aceitação, Automação do Teste de Aceitação e, por último, Gerenciamento de Teste de Aceitação.

O estudo também verificou, no espaço de tempo avaliado, a presença de usuários com deficiência como testadores. Foi possível analisar que houve uma crescente participação deste grupo nos estudos realizados nos últimos dois anos, aparentemente pelo aumento no interesse em inclusão e acessibilidade digitais. No total, identificaram-se 10 estudos com testes de aceitação com a inclusão de usuários com deficiência. Exceto pelo estudo S50, o foco foi o desenvolvimento de produtos destinados a essa parcela da

população e o teste acabou sendo tratado de maneira simples, com poucos detalhes, não especificando a maneira como foi elaborado nem detalhando a aplicação e a interação do usuário com o teste.

Em termo de trabalhos futuros, os resultados do mapeamento sistemático podem embasar a proposição e melhorias de técnicas de teste de aceitação que incluem pessoas com deficiência, por exemplo, pessoas cegas, pessoas surdas e pessoas com deficiência motora. Para tanto, as abordagens e ferramentas identificadas podem ser estendidas para satisfazer as demandas específicas de cada deficiência. Para esta parte é importante considerar a contribuição dos trabalhos da área de Interação Humano-Computador. Além disso, é importante observar que o teste de aceitação é uma atividade associada aos usuários para os quais o software foi desenvolvido [80]. Sendo importante investigar também a inclusão dos grupos com deficiência como parte interessada do desenvolvimento do software.

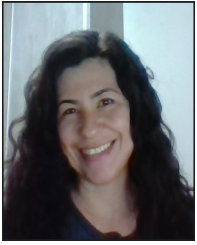
Com desdobramento deste mapeamento sistemático, pretende-se desenvolver uma abordagem de teste de aceitação a ser utilizada para validar sistemas acessíveis para o usuário surdo que se comunica usando língua de sinais.

## REFERENCES

- [1] P. Bourque and R. E. Fairley, eds., *SWEBOK: Guide to the Software Engineering Body of Knowledge*. Los Alamitos, CA: IEEE Computer Society, version 3.0 ed., 2014.
- [2] K. Naik and P. Tripathy, *Software Testing and Quality Assurance: Theory and Practice*. Wiley, 2011.
- [3] D. Gotterbarn, "How the new software engineering code of ethics affects you," *IEEE software*, vol. 16, no. 6, pp. 58–64, 1999.
- [4] Brasil, "Lei nº 13.146, de 6 de julho de 2015. institui a lei brasileira de inclusão da pessoa com deficiência (estatuto da pessoa com deficiência).," *Diário Oficial da República Federativa do Brasil*, 2015.
- [5] K. Silveira and R. Prikladnicki, "A systematic mapping study of diversity in software engineering: A perspective from the agile methodologies," pp. 7–10, 05 2019.
- [6] G. Rodriguez, R. Nadri, and M. Nagappan, "Perceived diversity in software engineering: a systematic literature review," *Empirical Software Engineering*, vol. 26, 09 2021.
- [7] P. A. d. M. Silveira-Neto, I. d. Carmo Machado, J. D. McGregor, E. S. de Almeida, and S. R. de Lemos Meira, "A systematic mapping study of software product lines testing," *Information and Software Technology*, vol. 53, pp. 407–423, May 2011.
- [8] J. Weiss, A. Schill, I. Richter, and P. Mandl, "Literature review of empirical research studies within the domain of acceptance testing," in *2016 42th Euromicro Conference on Software Engineering and Advanced Applications (SEAA)*, pp. 181–188, 2016.
- [9] E. C. D. Santos, P. Vilain, and D. H. Longo, "A systematic literature review to support the selection of user acceptance testing techniques," in *Proceedings of the 40th International Conference on Software Engineering: Companion Proceedings, ICSE '18*, (New York, NY, USA), p. 418–419, Association for Computing Machinery, 2018.
- [10] E. C. dos Santos, "Aplicação de técnicas de testes de aceitação automatizados para especificação de software em editais de licitação," Master's thesis, Universidade Federal de Santa Catarina, Florianópolis, Santa Catarina, Brasil, 2019.
- [11] K. R. F. Scannavino, E. Y. Nakagawa, S. C. P. F. Fabbri, and F. C. Ferrari, *Revisão Sistemática da Literatura em Engenharia de Software: teoria e prática*. Elsevier Brasil, 2017.
- [12] "http://parsif.al/,"
- [13] E. C. dos Santos and P. Vilain, "Automated acceptance tests as software requirements: An experiment to compare the applicability of fit tables and gherkin language," in *Lecture Notes in Business Information Processing*, pp. 104–119, Springer International Publishing, 2018.
- [14] E. C. D. Santos, P. Vilain, and D. H. Longo, "A systematic literature review to support the selection of user acceptance testing techniques," in *Proceedings of the 40th International Conference on Software Engineering: Companion Proceedings*, ACM, may 2018.

- [15] N. Chaveiro, S. B. R. Duarte, A. R. d. Freitas, M. A. Barbosa, C. C. Porto, and M. P. d. A. Fleck, "Instrumentos em língua brasileira de sinais para avaliação da qualidade de vida da população surda," *Revista de Saúde Pública*, vol. 47, pp. 616–623, 2013.
- [16] N. Chaveiro, C. L. Rodrigues, S. B. Duarte, R. Rodrigues de Oliveira Garcia, A. Ribeiro de Freitas, L. Rodrigues de Oliveira, K. Rocha Gomes da Silva, and C. C. Porto, "A tool to evaluate the quality of life of deaf people using whoqol instruments," in *2018 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, pp. 3871–3876, 2018.
- [17] J. Fang, M. P. Fleck, A. Green, K. McVilly, Y. Hao, W. Tan, R. Fu, and M. Power, "The response scale for the intellectual disability module of the whoqol: 5-point or 3-point?," *Journal of Intellectual Disability Research*, vol. 55, no. 6, pp. 537–549, 2011.
- [18] "Web content accessibility guidelines (wcag) overview." <https://www.w3.org/WAI/standards-guidelines/wcag/>. Accessed: 2021-05-10.
- [19] A. C. F. Silva, C. L. Rodrigues, N. Chaveiro, R. R. de Oliveira Garcia, S. B. Duarte, M. A. R. Araújo, V. B. Santos, K. R. G. da Silva, P. M. S. Rodrigues, L. M. Oliveira, and C. C. S. Sousa, "Lessons learned about oral-auditory and visual-spatial communication in requirements engineering with deaf stakeholders," in *SAC '20: The 35th ACM/SIGAPP Symposium on Applied Computing, online event, [Brno, Czech Republic], March 30 - April 3, 2020* (C. Hung, T. Cerný, D. Shin, and A. Bechini, eds.), pp. 1379–1386, ACM, 2020.
- [20] J. R. D. Cruz, R. A. Gazmin, A. E. K. Guillen, and M. J. C. Samonte, "Talkie: An Assistive Web-based Educational Application Using Audio Files and Speech Technology for the Visually Impaired," in *Proceedings of the 2020 The 6th International Conference on Frontiers of Educational Technologies*, (Tokyo Japan), pp. 39–42, ACM, June 2020.
- [21] N. J. Y. Bergantin, A. L. Buenaventura, K. A. M. Cordobin, M. J. C. Samonte, A. K. D. Balan, and G. B. Resuello, "e-ALEAP: Online Reviewer System for Alternative Learning Equivalency and Accreditation Program," in *Proceedings of the 2020 The 6th International Conference on Frontiers of Educational Technologies*, (Tokyo Japan), pp. 1–5, ACM, June 2020.
- [22] M. J. C. Samonte, A. R. I. Garcia, B. J. D. Valencia, and M. J. S. Ocampo, "Using Online Handwritten Character Recognition in Assistive Tool for Students with Hearing and Speech Impairment," in *Proceedings of the 2020 11th International Conference on E-Education, E-Business, E-Management, and E-Learning*, (Osaka Japan), pp. 189–194, ACM, Jan. 2020.
- [23] M. Co, J. K. Balan, J. D. Dolendo, J. C. De Goma, and M. J. C. Samonte, "TEXT2FSL: A Filipino Sign Language Phrase Translator Tool for Deaf and Mute," in *Proceedings of the 2020 The 6th International Conference on Frontiers of Educational Technologies*, (Tokyo Japan), pp. 82–86, ACM, June 2020.
- [24] A. K. Rai and A. K. Malviya, "Testing MapReduce program using Induction Method," in *2020 IEEE International Students' Conference on Electrical, Electronics and Computer Science (SCECS)*, (Bhopal, India), pp. 1–5, IEEE, Feb. 2020.
- [25] T. Yesmin and M. W. Carter, "Evaluation framework for automatic privacy auditing tools for hospital data breach detections: A case study," *International Journal of Medical Informatics*, vol. 138, p. 104123, June 2020.
- [26] B. Losada, J.-M. López-Gil, and M. Urretavizcaya, "Improving Agile Software Development Methods by means of User Objectives: An End User Guided Acceptance Test-Driven Development Proposal," in *Proceedings of the XX International Conference on Human Computer Interaction - Interacción '19*, (Donostia, Gipuzkoa, Spain), pp. 1–4, ACM Press, 2019.
- [27] C. Kuchta, R. D. Budiardja, and V. G. V. Larrea, "Harmony: A Harness Monitoring System for the Oak Ridge Leadership Computing Facility," in *Proceedings of the Practice and Experience in Advanced Research Computing on Rise of the Machines (learning)*, (Chicago IL USA), pp. 1–6, ACM, July 2019.
- [28] T. N. Thanh, M. Morgan, M. Butler, and K. Marriott, "Perfect Match: Facilitating Study Partner Matching," in *Proceedings of the 50th ACM Technical Symposium on Computer Science Education*, (Minneapolis MN USA), pp. 1102–1108, ACM, Feb. 2019.
- [29] R. Rwemalika, M. Kintis, M. Papadakis, Y. Le Traon, and P. Lorrach, "Ukwikora: continuous inspection for keyword-driven testing," in *Proceedings of the 28th ACM SIGSOFT International Symposium on Software Testing and Analysis - ISSA 2019*, (Beijing, China), pp. 402–405, ACM Press, 2019.
- [30] T. Tuglular and S. Sensulun, "SPL-AT Gherkin: A Gherkin Extension for Feature Oriented Testing of Software Product Lines," in *2019 IEEE 43rd Annual Computer Software and Applications Conference (COMPSAC)*, (Milwaukee, WI, USA), pp. 344–349, IEEE, July 2019.
- [31] A. N. Rachman, R. N. Shofa, E. N. F. Dewi, and A. Hidayat, "Application of Collision Detection Algorithm and Scoring Health Point in Fighting Games with Android-Based Augmented Reality Technology," in *2019 2nd International Conference of Computer and Informatics Engineering (IC2IE)*, (Banyuwangi, Indonesia), pp. 216–221, IEEE, Sept. 2019.
- [32] M. Nass, E. Alegroth, and R. Feldt, "Augmented Testing: Industry Feedback To Shape a New Testing Technology," in *2019 IEEE International Conference on Software Testing, Verification and Validation Workshops (ICSTW)*, (Xi'an, China), pp. 176–183, IEEE, Apr. 2019.
- [33] M. J. C. Samonte, R. A. Gazmin, J. D. S. Soriano, and M. N. O. Valencia, "BridgeApp: An Assistive Mobile Communication Application for the Deaf and Mute," in *2019 International Conference on Information and Communication Technology Convergence (ICTC)*, (Jeju Island, Korea (South)), pp. 1310–1315, IEEE, Oct. 2019.
- [34] R. A. Supangan, L. A. S. Acosta, J. L. S. Amarado, E. B. Blancaflor, and M. J. C. Samonte, "A gamified learning app for children with ADHD," in *Proceedings of the 2nd International Conference on Image and Graphics Processing - ICGIP '19*, (Singapore, Singapore), pp. 47–51, ACM Press, 2019.
- [35] S. Nurjanah, H. B. Santoso, and Z. A. Hasibuan, "The User Acceptance Test of An "ICT Adoption for Education" Framework," in *Proceedings of the 10th International Conference on Computer Modeling and Simulation - ICCMS 2018*, (Sydney, Australia), pp. 129–133, ACM Press, 2018.
- [36] Y. Xiao, L. Xiao, and Z. Li, "Framework Research on the Implementation of Automated Test User Requirements," in *Proceedings of the 2018 2nd International Conference on Computer Science and Artificial Intelligence - CSAI '18*, (Shenzhen, China), pp. 190–194, ACM Press, 2018.
- [37] E. K. Gebre-Amanuel, F. G. Tadesse, and A. T. Assalif, "Web based expert system for diagnosis of cattle disease," in *Proceedings of the 10th International Conference on Management of Digital EcoSystems - MEDES '18*, (Tokyo, Japan), pp. 66–73, ACM Press, 2018.
- [38] D. A. Martillano, A. F. D. Chowdhury, J. C. M. Dellosa, A. A. Murcia, and R. J. P. Mangoma, "PINDOTS: An Assistive Six-dot Braille Cell Keying Device on Basic Notation Writing for Visually Impaired Students with IoT Technology," in *Proceedings of the 2018 2nd International Conference on Education and E-Learning - ICEEL 2018*, (Bali, Indonesia), pp. 41–47, ACM Press, 2018.
- [39] M. J. C. Samonte, R. C. D. Mullen, J. M. P. Bañaga, W. Y. Cortes, and J. A. V. D. Calzada, "In-Patient Medication Delivery in Mobile App and Out-Patient Online Lab Results for Hospitals," in *Proceedings of the 2nd International Conference on E-Society, E-Education and E-Technology - ICSET 2018*, (Taipei, Taiwan), pp. 155–159, ACM Press, 2018.
- [40] S. Tachmammedov, Y. K. Hooi, and K. S. Kalid, "Automated Multi-factor Analytics for Cheat-Proofing Attendance-taking," in *Proceedings of the 2018 7th International Conference on Software and Computer Applications - ICSCA 2018*, (Kuantan, Malaysia), pp. 183–188, ACM Press, 2018.
- [41] A. Hidayati and R. Nabila, "E-commerce development using AngularJS framework and RESTful API," *IOP Conference Series: Materials Science and Engineering*, vol. 403, p. 012063, Oct. 2018.
- [42] S. Hotomski and M. Glinz, "GuideGen: a tool for keeping requirements and acceptance tests aligned," in *Proceedings of the 40th International Conference on Software Engineering: Companion Proceedings*, (Gothenburg Sweden), pp. 49–52, ACM, May 2018.
- [43] R. C. Morco, F. B. Calanda, J. A. Bonilla, M. J. S. Corpuz, J. E. Avestro, and J. M. Angeles, "e-RICE: An Expert System using Rule-Based Algorithm to Detect, Diagnose, and Prescribe Control Options for Rice Plant Diseases in the Philippines," in *Proceedings of the 2017 International Conference on Computer Science and Artificial Intelligence - CSAI 2017*, (Jakarta, Indonesia), pp. 49–54, ACM Press, 2017.
- [44] G. J. P. Rosales and A. C. Lagman, "Development of Industry Academic Linkage Alumni and Placement Portal," in *Proceedings of the 2017 International Conference on Information Technology - ICIT 2017*, (Singapore, Singapore), pp. 184–189, ACM Press, 2017.
- [45] M. R. L. Perez, A. C. Lagman, and R. T. Adao, "Event Management Solution Using Web Application Platform," in *Proceedings of the 2017 International Conference on Information Technology - ICIT 2017*, (Singapore, Singapore), pp. 206–211, ACM Press, 2017.
- [46] S. Hotomski, E. B. Charrada, and M. Glinz, "Aligning Requirements and Acceptance Tests via Automatically Generated Guidance," in *2017 IEEE*

- 25th International Requirements Engineering Conference Workshops (REW), (Lisbon, Portugal), pp. 339–342, IEEE, Sept. 2017.
- [47] P. Cruz and H. Astudillo, “Deriving high-priority acceptance test cases using utility trees: A case study,” in *2017 36th International Conference of the Chilean Computer Science Society (SCCC)*, (Arica), pp. 1–8, IEEE, Oct. 2017.
- [48] L. Almeida, E. Cirilo, and E. A. Barbosa, “SS-BDD: Automated Acceptance Testing for Spreadsheets,” in *Proceedings of the 1st Brazilian Symposium on Systematic and Automated Software Testing - SAST*, (Maringa, Parana, Brazil), pp. 1–10, ACM Press, 2016.
- [49] A. Koehl and H. Wang, “SERF: optimization of socially sourced images using psychovisual enhancements,” in *Proceedings of the 7th International Conference on Multimedia Systems - MMSys '16*, (Klagenfurt, Austria), pp. 1–12, ACM Press, 2016.
- [50] D. H. Longo, P. Vilain, L. P. da Silva, and R. d. S. Mello, “A web framework for test automation: user scenarios through user interaction diagrams,” in *Proceedings of the 18th International Conference on Information Integration and Web-based Applications and Services - iiWAS '16*, (Singapore, Singapore), pp. 458–467, ACM Press, 2016.
- [51] D. Clerissi, M. Leotta, G. Reggio, and F. Ricca, “Test Driven Development of Web Applications: A Lightweight Approach,” in *2016 10th International Conference on the Quality of Information and Communications Technology (QUATIC)*, (Lisbon, Portugal), pp. 25–34, IEEE, Sept. 2016.
- [52] L. Neubeck, G. Coorey, D. Peiris, J. Mulley, E. Heeley, F. Hersch, and J. Redfern, “Development of an integrated e-health tool for people with, or at high risk of, cardiovascular disease: The Consumer Navigation of Electronic Cardiovascular Tools (CONNECT) web application,” *International Journal of Medical Informatics*, vol. 96, pp. 24–37, Dec. 2016.
- [53] M. El-Attar and H. A. Abdul-Ghani, “Using security robustness analysis for early-stage validation of functional security requirements,” *Requirements Engineering*, vol. 21, pp. 1–27, Mar. 2016.
- [54] G. Blichmann, C. Radeck, S. Hahn, and K. Meißner, “Component-based workspace awareness support for composite web applications,” in *Proceedings of the 17th International Conference on Information Integration and Web-based Applications & Services - iiWAS '15*, (Brussels, Belgium), pp. 1–10, ACM Press, 2015.
- [55] D. H. Longo and P. Vilain, “Creating user scenarios through user interaction diagrams by non-technical customers,” in *Proceedings of the 27th International Conference on Software Engineering and Knowledge Engineering*, KSI Research Inc. and Knowledge Systems Institute Graduate School, jul 2015.
- [56] T. M. King, G. Nunez, D. Santiago, A. Cando, and C. Mack, “Legend: an agile DSL toolset for web acceptance testing,” in *Proceedings of the 2014 International Symposium on Software Testing and Analysis - ISSTA 2014*, (San Jose, CA, USA), pp. 409–412, ACM Press, 2014.
- [57] V. Horner and A. Coleman, “Evaluation of a guidelines-based e-health decision support system for primary health care in South Africa,” in *Proceedings of the Southern African Institute for Computer Scientist and Information Technologists Annual Conference 2014 on SAICSIT 2014 Empowered by Technology - SAICSIT '14*, (Centurion, South Africa), pp. 142–147, ACM Press, 2014.
- [58] T. Straszak and M. Smialek, “Automating Acceptance Testing with tool support,” pp. 1569–1574, Sept. 2014.
- [59] R. Ahmad, W. A. Chyi, A. Sarlan, and R. Kasbon, “Guiding novice database developers in database schema creation,” in *2014 IEEE Conference on e-Learning, e-Management and e-Services (IC3e)*, (Hawthorn, Australia), pp. 64–69, IEEE, Dec. 2014.
- [60] A. Marques, F. Ramalho, and W. L. Andrade, “Comparing Model-Based Testing with Traditional Testing Strategies: An Empirical Study,” in *2014 IEEE Seventh International Conference on Software Testing, Verification and Validation Workshops*, (OH, USA), pp. 264–273, IEEE, Mar. 2014.
- [61] S. Y. Bani Hashem, N. A. Mat Zin, N. F. Mohd Yatim, and N. Mohamed Ibrahim, “Improving Mouse Controlling and Movement for People with Parkinson’s Disease and Involuntary Tremor Using Adaptive Path Smoothing Technique via B-Spline,” *Assistive Technology*, vol. 26, pp. 96–104, Apr. 2014.
- [62] N. Nomura, Y. Kikushima, and M. Aoyama, “Business-Driven Acceptance Testing Methodology and Its Practice for E-Government Software Systems,” in *2013 20th Asia-Pacific Software Engineering Conference (APSEC)*, (Bangkok), pp. 99–104, IEEE, Dec. 2013.
- [63] E. Alegroth, M. Nass, and H. H. Olsson, “JAutomate: A Tool for System- and Acceptance-test Automation,” in *2013 IEEE Sixth International Conference on Software Testing, Verification and Validation*, (Luxembourg, Luxembourg), pp. 439–446, IEEE, Mar. 2013.
- [64] S. Kamalakar, S. H. Edwards, and T. M. Dao, “Automatically Generating Tests from Natural Language Descriptions of Software Behavior,” in *Proceedings of the 8th International Conference on Evaluation of Novel Approaches to Software Engineering*, (Angers, France), pp. 238–245, SciTePress - Science and Technology Publications, 2013.
- [65] M. Druk and M. Kropp, “ReFit: A Fit test maintenance plug-in for the Eclipse refactoring plug-in,” in *2013 3rd International Workshop on Developing Tools as Plug-Ins (TOPI)*, (San Francisco, CA, USA), pp. 7–12, IEEE, May 2013.
- [66] C.-Y. Hsieh, C.-H. Tsai, and Y. C. Cheng, “Test-Duo: A framework for generating and executing automated acceptance tests from use cases,” in *2013 8th International Workshop on Automation of Software Test (AST)*, (San Francisco, CA, USA), pp. 89–92, IEEE, May 2013.
- [67] A. Shafie, W. F. Wan Ahmad, N. Mohd., J. J. Barnachea, M. F. Taha, and R. L. Yusuff, ““SynMax”: A Mathematics Application Tool for Down Syndrome Children,” in *Advances in Visual Informatics* (D. Hutchison, T. Kanade, J. Kittler, J. M. Kleinberg, F. Mattern, J. C. Mitchell, M. Naor, O. Nierstrasz, C. Pandu Rangan, B. Steffen, M. Sudan, D. Terzopoulos, D. Tygar, M. Y. Vardi, G. Weikum, H. B. Zaman, P. Robinson, P. Olivier, T. K. Shih, and S. Velastin, eds.), vol. 8237, pp. 615–626, Cham: Springer International Publishing, 2013. Series Title: Lecture Notes in Computer Science.
- [68] W. M. Watanabe, R. P. M. Fortes, and A. L. Dias, “Using acceptance tests to validate accessibility requirements in RIA,” in *Proceedings of the International Cross-Disciplinary Conference on Web Accessibility - W4A '12*, (Lyon, France), p. 1, ACM Press, 2012.
- [69] Rafidah Mohd Arif and O. O. Khalifa, “Online tutoring system in college: Case study in private education,” in *2012 International Conference on Computer and Communication Engineering (ICCC)*, (Kuala Lumpur, Malaysia), pp. 608–611, IEEE, July 2012.
- [70] O. Liskin, C. Herrmann, E. Knauss, T. Kurpick, B. Rumpe, and K. Schneider, “Supporting Acceptance Testing in Distributed Software Projects with Integrated Feedback Systems: Experiences and Requirements,” in *2012 IEEE Seventh International Conference on Global Software Engineering*, (Porto Alegre, Rio Grande do Sul, Brazil), pp. 84–93, IEEE, Aug. 2012.
- [71] R. Hirschfeld, M. Perscheid, and M. Haupt, “Explicit use-case representation in object-oriented programming languages,” p. 10.
- [72] R. Borg and M. Kropp, “Automated acceptance test refactoring,” in *Proceeding of the 4th workshop on Refactoring tools - WRT '11*, (Waikiki, Honolulu, HI, USA), p. 15, ACM Press, 2011.
- [73] Z. M. Jiang, A. Avritzer, E. Shihab, A. E. Hassan, and P. Flora, “An Industrial Case Study on Speeding Up User Acceptance Testing by Mining Execution Logs,” in *2010 Fourth International Conference on Secure Software Integration and Reliability Improvement*, (Singapore, Singapore), pp. 131–140, IEEE, 2010.
- [74] W. F. W. Ahmad, E. A. P. Akhir, and S. Azmee, “Game-based learning courseware for children with learning disabilities,” in *2010 International Symposium on Information Technology*, (Kuala Lumpur, Malaysia), pp. 1–4, IEEE, June 2010.
- [75] M. El-Attar and J. Miller, “Developing comprehensive acceptance tests from use cases and robustness diagrams,” *Requirements Engineering*, vol. 15, pp. 285–306, Sept. 2010.
- [76] B. Güldali, S. Sauer, P. Winkelhane, H. Funke, and M. Jahnich, “Pattern-based generation of test plans for open distributed processing systems,” in *Proceedings of the 5th Workshop on Automation of Software Test - AST '10*, (Cape Town, South Africa), pp. 119–126, ACM Press, 2010.
- [77] C. Wohlin, P. Runeson, M. Höst, M. C. Ohlsson, and B. Regnell, *Experimentation in Software Engineering*. Springer, 2012.
- [78] C. Wohlin, “Guidelines for snowballing in systematic literature studies and a replication in software engineering,” in *Proceedings of the 18th International Conference on Evaluation and Assessment in Software Engineering*, EASE '14, (New York, NY, USA), Association for Computing Machinery, 2014.
- [79] M. Bown and A. Sutton, “Quality control in systematic reviews and meta-analyses,” *European Journal of Vascular and Endovascular Surgery*, vol. 40, no. 5, pp. 669–677, 2010.
- [80] R. W. Miller, C. T. Collins, S. Developer, S. Software, and Developer, “Acceptance testing,” Tech. Rep. practically, 2001.



**Livia Fernanda de Oliveira** é aluna de mestrado do Programa de Pós-Graduação em Ciência da Computação pelo Instituto de Informática da Universidade Federal de Goiás. Possui interesses em Engenharia de Software, desenvolvimento de sistemas, acessibilidade de sistemas e pesquisas sobre a interação de grupos minoritários com software. Atualmente trabalha na Diretoria de Tecnologia da Informação da Corregedoria do Tribunal de Justiça do Estado de Goiás, atuando na implementação dos sistemas do Conselho Nacional

de Justiça junto ao TJGO.



**Cássio Leonardo Rodrigues** possui graduação em Ciência da Computação pela Universidade Federal de Goiás em 1998, mestrado em Informática pela Universidade Federal de Campina Grande em 2003 e doutorado em Ciência da Computação pela Universidade Federal de Campina Grande em 2010. Atualmente é professor Associado da Universidade Federal de Goiás. Tem experiência na área de Engenharia de Software, atuando principalmente nos seguintes temas: teste de software, projeto de software e engenharia de software com a

participação da pessoa surda.



**Renato de Freitas Bulcão-Neto** recebeu seu título de Doutor em Ciência da Computação e Matemática Computacional pela Universidade de São Paulo, São Carlos, Brasil (2006). Ele é Professor Associado da Universidade Federal de Goiás (UFG), Brasil, desde 2010. Seus interesses de pesquisa incluem Engenharia de Software, Requisitos de Software, Internet das Coisas e Informática em Saúde. Entre em contato com ele em [rbulcao@ufg.br](mailto:rbulcao@ufg.br).