

**UNIVERSIDADE DO ESTADO DO AMAZONAS
ESCOLA SUPERIOR DE TECNOLOGIA**

KLIRSSIA MATOS ISAAC SAHDO

**PRIMA: UMA FERRAMENTA PARA AUTOMAÇÃO DE REVISÃO DE
TESTES APLICADA À PLATAFORMA ANDROID MOBILE**

Manaus
2023

KLIRSSIA MATOS ISAAC SAHDO

**PRIMA: UMA FERRAMENTA PARA AUTOMAÇÃO DE REVISÃO DE
TESTES APLICADA À PLATAFORMA ANDROID MOBILE**

Pesquisa desenvolvida durante a disciplina de Trabalho de Conclusão de Curso II e apresentada à banca avaliadora do Curso de Engenharia Elétrica da Escola Superior de Tecnologia da Universidade do Estado do Amazonas, como pré-requisito para obtenção do título de Engenheira Eletricista.

Orientador: Jozias Parente de Oliveira.

Manaus
2023

Universidade do Estado do Amazonas – UEA
Escola Superior de Tecnologia - EST

Reitor:

André Luiz Nunes Zogahib

Vice-Reitor:

André Luiz Nunes Zogahib

Diretora da Escola Superior de Tecnologia:

Ingrid Sammyne Gadelha Figueiredo

Coordenador do Curso de <Nome do Curso>:

Israel Gondres Torné

Banca Avaliadora composta por:

Data da defesa: 16/03/2023.

Prof. Jozias Parente de Oliveira, Dr (Orientador)

Prof. Fábio de Sousa Cardoso, Dr

Prof. Bruno da Gama Monteiro, Ms

CIP – Catalogação na Publicação

Sahdo, Klirssia Matos Isaac

PRIMA: uma ferramenta para automação de revisão de testes aplicada à plataforma Android mobile / Klirssia Matos Isaac Sahdo; [orientado por] Jozias Parente de Oliveira. – Manaus: 2023.
44 p.: il.

Trabalho de Conclusão de Curso (Graduação em Engenharia Elétrica).
Universidade do Estado do Amazonas, 2023.

1. Android. 2. Mobile. 3. Desenvolvimento de Testes. 4. Revisão de Testes. I. de Oliveira, Jozias Parente.

KLIRSSIA MATOS ISAAC SAHDO

**PRIMA: UMA FERRAMENTA PARA AUTOMAÇÃO DE REVISÃO DE
TESTES APLICADA À PLATAFORMA ANDROID MOBILE**

Pesquisa desenvolvida durante a disciplina de Trabalho de Conclusão de Curso II e apresentada à banca avaliadora do Curso de Engenharia Elétrica da Escola Superior de Tecnologia da Universidade do Estado do Amazonas, como pré-requisito para obtenção do título de Engenheira Eletricista.

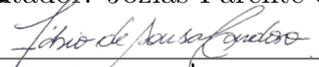
Nota obtida: 10,0 (Dez vírgula zero _____)

Aprovado em 23 / 03 / 2022

Área de concentração: Engenharia Elétrica

BANCA EXAMINADORA


Orientador: Jozias Parente de Oliveira


Avaliador: Fábio de Sousa Cardoso


Avaliador: Bruno da Gama Monteiro

Manaus

2023

Dedicatória

Dedico este trabalho aos meus pais que fizeram muitas renúncias para me proporcionar qualidade de vida e boas condições de estudo. Meus pais me proporcionaram as coisas mais valiosas que alguém pode ter: amor e conhecimento. Sou grata a Deus pelas oportunidades e bênção para chegar onde estou.

AGRADECIMENTO

Agradeço aos meus professores do curso de Engenharia Elétrica que sempre procuraram formas variadas e didáticas de explicar os conhecimentos mais diversos e complexos do ramo da Engenharia. Principalmente, ao professor orientador Jozias Parente de Oliveira, por dedicar atenção e todo o suporte científico com base em sua experiência profissional de vasto conhecimento, e técnicas de escritas para a realização deste trabalho.

À minha irmã Kirssia Matos Isaac Sahdo, por ter sempre se esforçado e se apresentado para mim como um bom exemplo de estudante e profissional, aventurando-se no mundo para conquistar conhecimento e seu espaço, que por sua vez, na Engenharia Civil, que assim como na Engenharia Elétrica, tem um espaço majoritariamente ocupado por homens.

Ao Sidia Instituto de Ciência e Tecnologia por ter viabilizado este estudo e implementação no ramo de testes da Engenharia, contribuindo com materiais de pesquisa, participação de colaboradores e a oportunidade de abordar este tema de pesquisa.

RESUMO

Com os avanços da tecnologia, o desenvolvimento de software torna-se cada vez mais complexo e a necessidade de testar um produto embarcado com ele tornou-se um fator fundamental para evitar falhas e corrigir erros. Esse fator tem grande relevância nos dispositivos móveis Android, que possuem um complexo software e uma ampla variedade de modelos no mercado. A fim de garantir a qualidade para os consumidores, as empresas estão investindo em padrões e criando regras, como é o caso da empresa Google, que disponibiliza um série de requisições para homologação dos seus produtos em aparelhos Android. Os testes relacionados à homologação destes aparelhos eletrônicos para estas regulamentações geram uma grande quantidade de artefatos de teste no final do processo, o qual se torna desafiador para revisão puramente manual do desenvolvedor de teste. Desta forma, o objetivo deste trabalho de conclusão de curso é propor e avaliar uma ferramenta de automação chamada PRIMA, para melhorar a garantia de qualidade nos testes de dispositivos móveis Android. Para análise e verificação da ferramenta proposta, conduziu-se um experimento em um ambiente de testes real em uma empresa do ramo, a fim de se obter dados suficientes para análise do cenário. Foi possível, com os resultados obtidos, compreender que houve uma melhoria de 59% na prevenção de falhas usando a automação de revisão de testes. Esse resultado indica que a PRIMA pode ajudar na prevenção de falhas encontradas no final do processo de homologação de releases Android.

Palavras chave: Android, Mobile, Desenvolvimento de Testes, Revisão de Testes.

ABSTRACT

With the advances of technology, software development is becoming increasingly complex, and the need to test an embedded product with it has become a fundamental factor to avoid failures and fix errors. This factor has great relevance in Android mobile devices, which have complex software and a wide variety of models on the market. In order to ensure quality for consumers, companies are investing in standards and creating rules, as is the case with Google, which provides a series of requirements for homologation of its products on Android devices. Tests related to the homologation of these electronic devices for these regulations generate a large amount of test artifacts at the end of the process, which becomes challenging for pure manual review by the testing developer. Thus, the objective of this thesis is to propose and evaluate an automation tool called PRIMA, to improve quality assurance in Android mobile device testing. To analyze and verify the proposed tool, an experiment was conducted in a real test environment in a particular company in the field, in order to obtain sufficient data for analysis of the scenario. With the results obtained, it was possible to understand that there was a 59% improvement in preventing failures using the test review automation. This result indicates that PRIMA can help prevent failures found at the end of the Android releases homologation process.

Keywords: Android, Mobile, Development Tests, Tests Review.

Lista de Tabelas

1	Ocorrências Mais Frequentes de 2021.	28
2	Ocorrências Inseridas no Experimento.	31
3	Comparação dos Resultados do Experimento da Ferramenta PRIMA. . .	35

Lista de Figuras

1	Ciclo de Vida do Desenvolvimento de Software.	18
2	Simplificação do Processo de Homologação de <i>Releases</i> Android.	19
3	Submissões aprovadas com e sem feedbacks em 2021.	22
4	Submissões aprovadas com e sem feedbacks em 2022.	23
5	Arquitetura da ferramenta de automação PRIMA.	24
6	Classificação de Inconsistências mais Recorrentes em 2021.	29

SUMÁRIO

1	INTRODUÇÃO	12
2	REFERENCIAL TEÓRICO	16
2.1	Trabalhos Relacionados	17
2.2	Homologação de <i>Releases</i> Android	19
2.2.1	Execução de Testes	20
2.2.2	Revisão de Resultados	22
3	METODOLOGIA	24
4	EXPERIMENTO	27
4.1	Definição de Participantes	27
4.2	Escolha dos Testes	28
4.3	Organização do Experimento	31
4.4	Passos do Experimento	32
4.4.1	Cenário A - Revisão sem a ferramenta PRIMA	32
4.4.2	Cenário B - Revisão com a ferramenta PRIMA	32
5	RESULTADOS	34
5.1	Análise dos Resultados	34
5.2	Validade dos Resultados	37
5.2.1	Validade de Conclusão	37
5.2.2	Validade Interna	38
5.2.3	Validade Externa	38
5.2.4	Validade de <i>Constructo</i>	38
6	CONCLUSÃO	40
	REFERÊNCIAS	42

1 INTRODUÇÃO

O processo de desenvolvimento de softwares, sofreu diversas mudanças ao longo dos anos, partindo desde um princípio em que a sua criação e garantia de funcionamento eram responsabilidade somente do programador, não existia preocupação com o ramo de testes (GAROUSI et al., 2020). Dessa forma, promover inovação na qualidade dos produtos e serviços representou um desafio maior diante de um cenário de negócios competitivo, no que diz respeito a qualidade de entrega (MISHRA; OTAIWI, 2020). Os testes se tornaram um fator de suma importância no processo de desenvolvimento, fundamentais para evitar problemas e corrigir erros.

Portanto, empresas fabricantes de dispositivos mobile que pretendem utilizar o sistema Android e serviços da Google para dispositivos móveis - Google Mobile Service (GMS) em seus aparelhos, precisam atender um conjunto de requisitos e regras estabelecidas que estão descritos em um arquivo chamado Compatibility Definition Document (CDD), que quer dizer, Documento de Definição de Compatibilidade (GOOGLE, 2022). Este documento define padrões rigorosos de qualidade, visto a necessidade de garantir o pleno funcionamento do sistema para o usuário final dos dispositivos móveis. Os fabricantes produzem diversos modelos de configurações distintas, como por exemplo, modelos com hardware mais potente e outros de menor custo a fim de atender a demanda do público, de acordo com suas necessidades, totalizando mais de 2 bilhões de smartphones atualmente (POSSEMATO et al., 2021). Uma vez produzidos pelos fabricantes, estes aparelhos podem ser negociados em contrato com operadoras para que apliquem customizações de região e aplicativos próprios. Essas customizações são específicas de cada operadora, que constroem um firmware do sistema que possa ser aplicado nos modelos, também conhecidos como builds ou *releases* (ALURE; PURI, 2021). As regras particulares das operadoras e fabricantes, como aplicativos próprios, são customizações que devem ser validadas também no processo de homologação, por exemplo, na empresa Sidia Instituto de Ciência e Tecnologia, atualmente há 172 modelos diferentes e mais de 60 operadoras da América Latina.

Os processos de manutenção em novos e em antigos modelos ou até mesmo atualizações do sistema operacional, precisam ser testados para assegurar os padrões de qualidade estabelecidos. Estes processos correspondem ao que é denominado como homologação de uma release. Neste contexto, o processo de homologação envolve um conjunto de testes pré-estabelecidos, que são realizados por profissionais da área de qualidade de software. Ao término dos testes, um grande conjunto de artefatos é gerado e uma vez corretos, configuram todos os elementos necessários para se enviar à Google e obter a aprovação da release. Porém, com o grande volume de arquivos de artefatos de teste que precisam ser verificados manualmente (por volta de 500 arquivos), é muito comum que passem erros despercebidos pelos testadores, conseqüentemente, esses erros impactam no tempo de entrega e muitas vezes em retrabalho a fim de corrigir inconsistências identificadas nas *releases*. Esse cenário ressalta a necessidade de soluções que minimizem ou resolvam esses efeitos.

Desta forma, surge por meio deste trabalho, a ideia de criar uma ferramenta de automação para a detecção de inconsistências em artefatos de teste de qualidade no contexto do processo de homologação de *releases* Android. Uma vez, testada esta solução e apresentando um potencial significativo na entrega de um produto mobile, pode-se considerar como um complemento do processo, uma alternativa para os profissionais do ramo. Isso é possível expondo uma avaliação da automação de revisão através de um experimento com a participação de desenvolvedores de teste de software dentro de uma empresa da área com este cenário. Assim, haverá indícios se a ferramenta pode ou não auxiliar o processo de revisão de uma equipe de testes na prevenção de inconsistências ao final do processo de homologação de *releases* Android.

Este trabalho intitulado como "PRIMA: uma ferramenta para automação de revisão de testes aplicada à plataforma Android mobile" que surgiu então diante do complexo cenário de dispositivos com software embarcado para o ramo da telefonia, vem em busca de aplicar e testar a ferramenta de automação Primary Review In Mobile Android (PRIMA) - Revisão Primária em Dispositivos Móveis Android, identificando sua acessibilidade na revisão de artefatos de teste, com base em dados reais de pesquisa e simulação de re-

visão dentro de um ambiente de trabalho que realiza testes mobile constantemente, logo então vale destacar os objetivos específicos deste estudo que são: levantar dados acerca das inconsistências mais comuns encontradas nos artefatos de teste no fim do processo de *releases* de homologação de uma empresa do ramo; implementar a ferramenta de automação, preparando o banco de dados com as informações necessárias para revisão dos testes, quais sejam: a quantidade de regras que precisam ser validadas, e arquivos de resultados específicos onde cada campo pode ser encontrado; conduzir um experimento com o uso da ferramenta na participação de desenvolvedores de teste de software; e, analisar o desempenho da ferramenta PRIMA comparando o tempo de revisão com o uso da ferramenta PRIMA e sem o uso da ferramenta.

Como no atual estado a humanidade exerce atividades essenciais através de dispositivos mobile, como transferências bancárias, acesso à informação, comunicação, urgência e apresentação de documentos digitais, é importante que no manuseio destas aplicações, ocorra um bom funcionamento e não venha existir um efeito contrário ao que de fato é proposto com este recurso que é de facilitar o cotidiano das pessoas. Dentre as principais contribuições é possível citar: a inclusão de uma automação no processo de revisão de artefatos de teste da homologação de *releases* Android, com o intuito de reduzir inconsistências; um levantamento das inconsistências mais comuns no ano de 2021 e 2022 para uma melhor compreensão deste cenário; e, um possível ganho no tempo de aprovação das *releases* devido uma redução de feedbacks.

Vale ressaltar também os motivos que levaram à decisão de se abordar esse tema dentro do universo acadêmico, dentre eles há o fato de que o curso de Engenharia Elétrica aborda com excelência o desenvolvimento tecnológico de dispositivos eletrônicos, e a fase de testes é essencial para sua validação. Fundamentar uma automação que revisa testes de dispositivos eletrônicos mobile, por sua vez, exerce um ganho na rotina de homologação de um produto, que geralmente é trabalhosa e extensa. Além de se trabalhar Engenharia de Qualidade na Engenharia Eletrônica, também se abrange Engenharia de Controle e Automação, e Engenharia de Telecomunicações, sendo um tema interdisciplinar e que pode ajudar muitos estudantes e profissionais vinculados a essas áreas.

E para conformidade dos tópicos a serem abordados de forma clara e objetiva neste trabalho, há uma apresentação dividida em 5 capítulos a partir deste atual.

Capítulo 2 – Referencial Teórico, apresentando toda a fundamentação de escrita deste material com base em trabalhos relacionados, homologação de releases android, como ocorre a execução de testes e revisão dos resultados.

Capítulo 3 – Metodologia, onde são descritas as características da implementação aplicada, o comportamento e a arquitetura do sistema no cenário da ferramenta de automação de revisão de testes.

Capítulo 4 – Experimento, com a apresentação do planejamento, do levantamento de dados, da escolha dos participantes para a criação do escopo do experimento, e por fim, a explanação do procedimento dos cenários do experimento.

Capítulo 5 – Resultados, descrevendo os dados obtidos para um posterior conclusão, gerando uma análise e percepção dedutiva de fatores que podem ter colaborado para a veracidade e confiança das informações (validade dos resultados).

Capítulo 6 – Considerações Finais, concluindo e respondendo as questões levantadas na introdução do trabalho, avaliando a proposta de estudo, implementação e experimento.

2 REFERENCIAL TEÓRICO

O desenvolvimento de softwares tem como um dos seus principais parâmetros de qualidade a disposição do produto para uma bateria de testes. Através deles, ocorre um mapeamento dos recursos existentes e avaliação de desempenho, obtendo-se melhorias do software e conseqüentemente a garantia de performance. Nesse ciclo do produto, há duas maneiras de executar os testes: de forma manual e de forma automatizada. Em comparação, o teste automatizado se apresenta com mais vantagens, enquanto que o teste manual não é tão confiável, tem repetição limitada e velocidade de execução extensa. Diante da comum usabilidade de dispositivos móveis no cotidiano, através de transferências bancárias, saúde e transporte, a automação dos desenvolvimentos móveis ganharam maior notoriedade por pesquisadores e profissionais (LIN; SALEHNAMADI; MALEK, 2020; ATEŞOĞULLARI; MISHRA, 2020; FERME; PAUTASSO, 2018).

Existem empresas no mercado que são fabricantes destes dispositivos, elas são chamadas de Original Equipment Manufacturer (OEM), que significa Fabricantes de Equipamentos Originais. Os OEMs desenvolvem seus modelos particulares e em fábricas próprias (POSSEMATO et al., 2021). Caso estes fabricantes optem por utilizar o Android e os serviços GMS (Google Mobile Service) em seus dispositivos fabricados, deve-se atender um conjunto de regras e requisitos definidos em contrato, cuja licença é denominada como MADA (Mobile Application Distribution Agreement). Assim, os fabricantes realizam contratos com o Google de forma a estipular e atender as condições necessárias para manter a utilização do Android em seus aparelhos. Por exemplo, uma das condições exigidas é a de que ocorram manutenções regulares de segurança a fim de proteger o usuário de aplicativos mal-intencionados, entre outras possíveis falhas de segurança (GOOGLE, 2021a). Este processo de validação é realizado pelos próprios fabricantes que, após a negociação com operadoras de telefonia, aplicam suas modificações, constroem um firmware do sistema que possa ser aplicado nos modelos, também conhecidos como builds ou *releases*. O firmware se refere às aplicações e ao sistema operacional que ditam como um smartphone irá operar (ALURE; PURI, 2021). As *releases* ou atualizações se dividem em estágios que

indicam as etapas em que os softwares estão, são usados alguns termos como; alpha, beta, release candidate, release to manufacture e estável (PIERCE, 2022).

As possibilidades de customização são geradas de forma a atender contratos entre operadoras e fabricantes (GOOGLE, 2021b). A compra de aplicativos a serem implementados de forma exclusiva e as regras de posicionamento de aplicativo próprios de operadoras são algumas das customizações geradas e que devem ser validadas para um então processo de homologação. A empresa de cenário para este estudo trabalha com testes em *releases* Android, de forma a assegurar que todos os requisitos de determinados fabricantes e do Google sejam atendidos nos modelos lançados por estes fabricantes, o chamado processo de homologação de *releases* Android.

O processo de desenvolvimento de um software engloba um conjunto de procedimentos estratégicos organizados a fim de arquitetar, desenvolver e testar o software. O Ciclo de vida de um software é uma estrutura que indica processos e atividades envolvidas no desenvolvimento, operação e manutenção de um software, abrangendo de fato toda a vida de um sistema. Neste ciclo, existem modelos que definem como o software será desenvolvido, lançado, aprimorado e finalizado. Um exemplo de integração dos passos citados em um ciclo de vida do desenvolvimento de um software, é o MS Security Development Lifecycle (SDL). A SDL foi criada como consequência da iniciativa de computação confiável da Microsoft iniciada em 2002 e desenvolve um ciclo de vida padrão através de práticas específicas destinadas a melhorar a segurança e a privacidade dos seus produtos. As atividades são agrupadas em fases, como: planejamento, análises, desenvolvimento, os testes e a integração e o processo de homologação e entrega do sistema.

2.1 TRABALHOS RELACIONADOS

A abordagem da empresa escolhida para este trabalho está inserida na etapa 5 do processo de ciclo de vida: Homologação e Entrega (figura 1). Poucas pesquisas de Engenharia estão focadas na etapa de Homologação de *Releases*, menos ainda para Android. Não há muitos trabalhos com esse foco, mas dois merecem destaque neste contexto (FIGUEIREDO et al., 2022; BERNARDON et al., 2022).

Figura 1 – Ciclo de Vida do Desenvolvimento de Software.



Fonte: Klirssia Sahdo.

O primeiro artigo, fala do processo automatizado de detecção de problemas no processo de análises de capturas de tela Android como um processo manual. A solução apresentada visa reduzir o tempo de homologação de builds android através de uma automação utilizada para detecção de inconsistências em imagens, utilizando algoritmos de Machine Learning como Optical Character Recognition (OCR) e Support Vector Machine (SVM), nomeando os resultados em três categorias: correto, incorreto e não aplicável. Depois que o algoritmo faz a validação, o resultado é apresentado a testadores experientes para revisar manualmente o resultado obtido pela automação. Um experimento foi apresentado neste artigo para comparar a revisão de testadores puramente manual com a revisão de automação, a automação obteve resultados mais assertivos, chegando a 0,99 F1-Score. Um problema neste desenvolvimento de automação identificado pela equipe foi a fragilidade dos elementos de UI, que eram frequentemente modificados dependendo da customização (FIGUEIREDO et al., 2022).

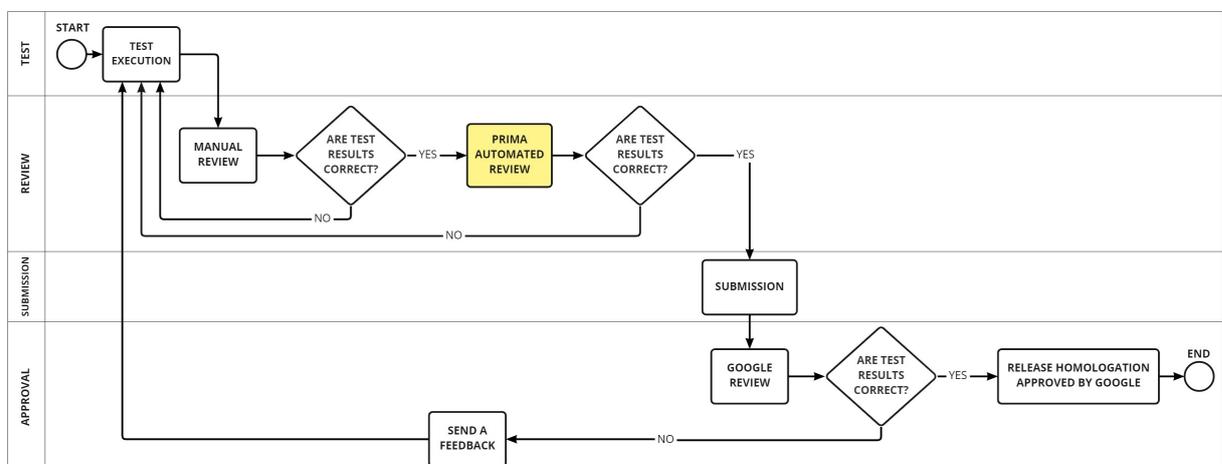
Outro artigo que merece destaque é uma proposta de melhoria no processo de homologação de Builds Android através da implementação do processo de internal review

(BERNARDON et al., 2022). A fim de evitar eventuais inconsistências nos resultados antes de submeter os testes para aprovação da Google, o artigo propõe, em sua metodologia, enviar todos os artefatos de teste para que outro profissional de qualidade de software possa checar dentro dos arquivos se todas as informações estão de acordo com a release a ser homologada. Normalmente, estes artefatos são avaliados por pessoas que não estão diretamente ligadas ao teste, evitando que se possa comprometer a confiança e acurácia dos dados, como a probabilidade de que por estar ativamente participando da geração dos resultados eles já estarão corretos de antemão. A quantidade de arquivos, pontos de revisão e fadiga podem influenciar na análise minuciosa de todo o teste, provocando até mesmo efeitos negativos aos profissionais que têm mais experiência no reconhecimento e análise de falhas. Este trabalho afirma ter apresentado um ganho de 6% aplicando tal metodologia (BERNARDON et al., 2022).

Tendo como base os fatores citados nos trabalhos anteriores, para colaborar com a revisão e minimizar a exaustão neste processo, propõe-se neste trabalho a criação de uma ferramenta que automatiza parte da homologação, acessando os campos desta grande quantidade de arquivos e comparando com os resultados corretos.

2.2 HOMOLOGAÇÃO DE *RELEASES* ANDROID

Figura 2 – Simplificação do Processo de Homologação de *Releases* Android.



Fonte: Klirssia Sahdo.

O foco deste trabalho é na etapa de homologação destas *releases*, ou seja, a fase em

que serão validadas por testadores antes de serem enviadas para aprovação final da Google e, em seguida, disponibilizadas como atualização para o público. Este processo é parte da garantia dos padrões de qualidade de software, já que as *releases* android precisam ser testadas antes de serem disponibilizadas para os usuários. Esse cenário é explicado através da figura 2, que apresenta uma versão simplificada do processo, iniciando-se a partir do momento que os testadores recebem um pedido de teste de uma nova releas conforme a seguintes fases:

1. Execução dos testes: Um pedido de homologação é recebido pela equipe, em que por meio de uma versão específica de software, realizam os testes necessários.
2. Revisão dos resultados: A partir dos resultados obtidos, os testadores realizam uma revisão dos artefatos gerados de forma manual. Se todos os arquivos estiverem corretos, os resultados se encontram prontos para submissão.
3. Revisão por meio da ferramenta PRIMA: Através dos artefatos gerados e após a revisão de forma manual.
4. Submissão e Revisão da Google: A equipe realiza a submissão dos resultados e a Google recebe os arquivos para verificar se todos os requisitos estão sendo cumpridos.

2.2.1 Execução de Testes

A partir do momento em que o produto estiver pronto e com o sistema operacional Android embarcado, uma gama de testes pré-definidos pela Google é requerida. Os testes serão responsáveis por verificar a segurança, o desempenho e a compatibilidade do dispositivo com sistema operacional Android e aplicativos pertencentes à Google. Os testes são baseados no Tradefed - Trade Federation, um framework contínuo desenvolvido para testes em aparelhos Android (GOOGLE, 2020)

Partindo deste princípio, os testes são aplicados a começar pelo Conjunto de Testes de Compatibilidade (CTS). O CTS abrange um conjunto variado de testes, que garante a compatibilidade das implementações da estrutura do Android entre parceiros Original Equipment Manufacturer e entre versões de plataforma. Em algumas versões de Android,

o CTS pode chegar a ter mais de 2 milhões de casos de teste executados. O CTS Verifier, por sua vez, é um complemento ao CTS, ele fornece testes para Interfaces de Programação de Aplicação (API) e funções que não podem ser testadas em um dispositivo estacionário sem entrada manual, testes estes que não podem ser automatizados, consequentemente executados no Device Under Test (DUT) como: a qualidade de áudio, tela sensível ao toque, acelerômetro e câmera.

Além do CTS, há o Vendor Test Suite (VTS), que consiste em um conjunto de atividades e casos de teste voltados a ajudar o aprimoramento da confiabilidade e a conformidade do sistema Android, como também de softwares de sistemas de baixo nível. Além desses xTS testes, existem outros que OEMs só poderão obter por meio da Google, pois não se encontram disponíveis no mercado. A começar pelo GMS Test Suite (GTS), ele engloba um conjunto de verificações automatizadas pertencentes à Google, direcionado a validar a presença ou ausência dos aplicativos de GMS integrados nos dispositivos, além de verificar se quesitos contratuais alinhados entre os OEMs e as operadoras foram atendidos. O Security Test Suite (STS) é um novo conjunto de testes de segurança para testes do CTS, com atualizações mensais, auxilia OEMs a validar se os dispositivos não estão vulneráveis, verificando se as vulnerabilidades foram corretamente corrigidas e se os testes de segurança foram devidamente aplicados.

Todas essas suítes de teste são consideradas como testes automatizados. Além dos testes automatizados, existem os testes manuais, eles precisam da interação humana para realizar etapas do processo. Como por exemplo, verificação de imagens, tirar fotos relacionadas à imagem de inicialização do modelo, sensores, giroscópios, entre outros. Assim, os testes manuais e específicos compõem os artefatos de teste para se homologar uma release.

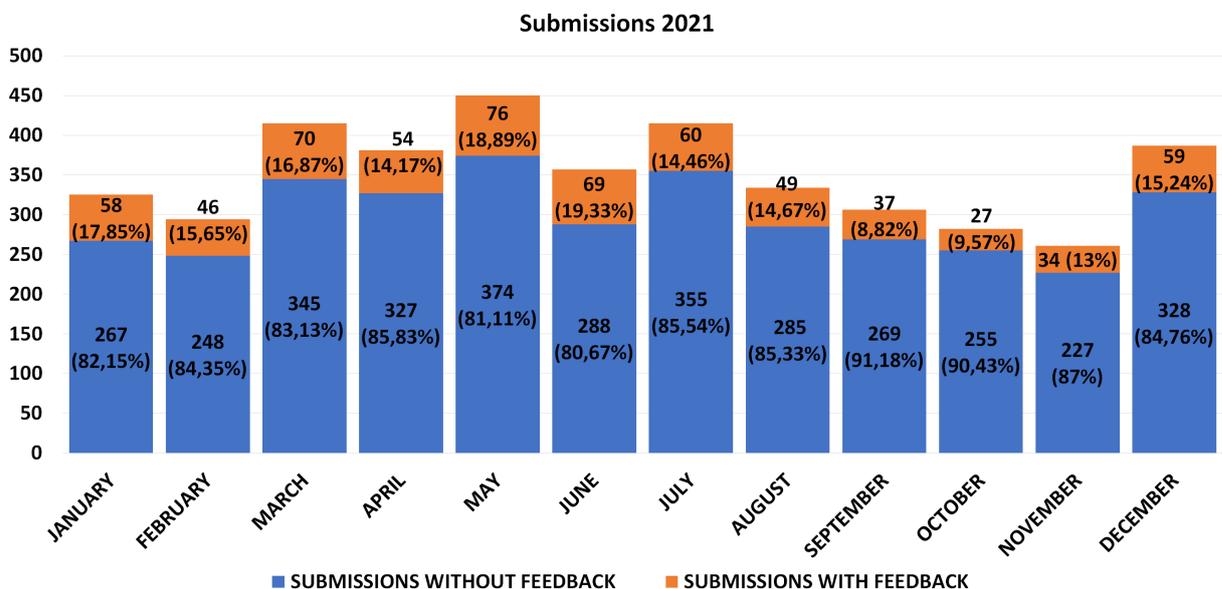
Todos estes testes citados são executados por profissionais da área de qualidade de software que, ao receber uma nova release de atualização referente a um modelo de fabricante, realizam o processo de embarcar o firmware dessa nova release no modelo, e aplicam o conjunto de testes citados anteriormente a fim de homologá-lo. Porém, em nossa empresa, trabalhamos com *releases* na América Latina, aplicando manutenções e

testes de builds para mais de 172 modelos, em mais de 60 operadoras. Considerando que para cada operadora são gerados mais de 10 artefatos de teste, ao validar a release de um modelo para que este seja liberado para público, necessitamos que um testador consiga revisar mais de 500 artefatos por teste. Considerando que a quantidade de pontos de revisão em cada um destes testes seja grande, conseqüentemente temos um imenso esforço e a necessidade de maior atenção para que nenhuma inconsistência venha a passar no resultado final.

2.2.2 Revisão de Resultados

As inconsistências de resultados são definidas como não conformidades em relação aos atributos e regras necessárias de cada teste específico. A ocorrência de uma ou mais inconsistências no conjunto de artefatos de teste é chamada de Feedback. Caso algum resultado de teste possua Feedback, a release será rejeitada pela Google, de tal forma deverá ser submetida novamente com todos os resultados corretos.

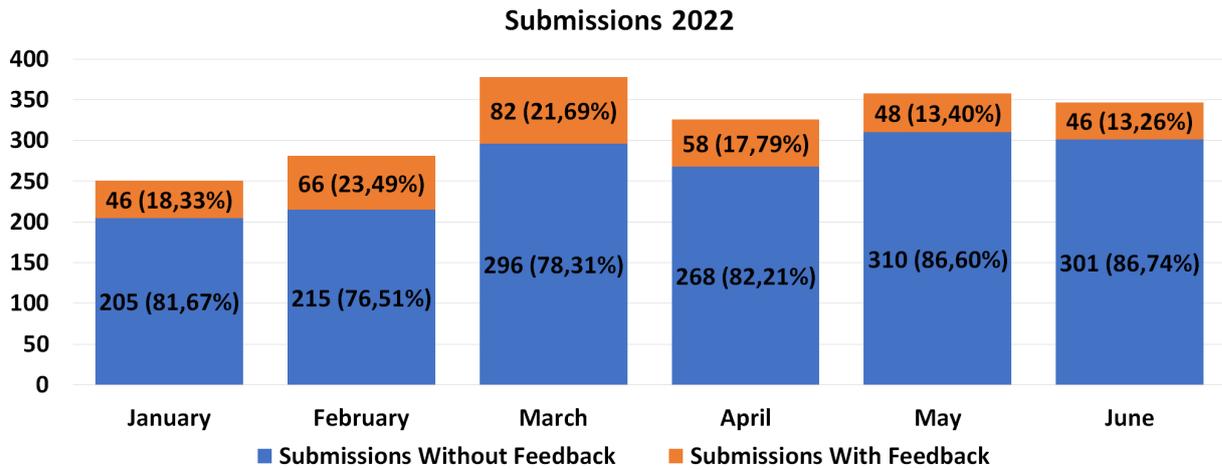
Figura 3 – Submissões aprovadas com e sem feedbacks em 2021.



Fonte: Sidia, 2021.

Em 2021, houve 4207 resultados submetidos, 639 destes dados continham uma ou mais inconsistências que geraram atrasos e repetição do procedimento, refletindo uma

Figura 4 – Submissões aprovadas com e sem feedbacks em 2022.



Fonte: Sidia, 2022.

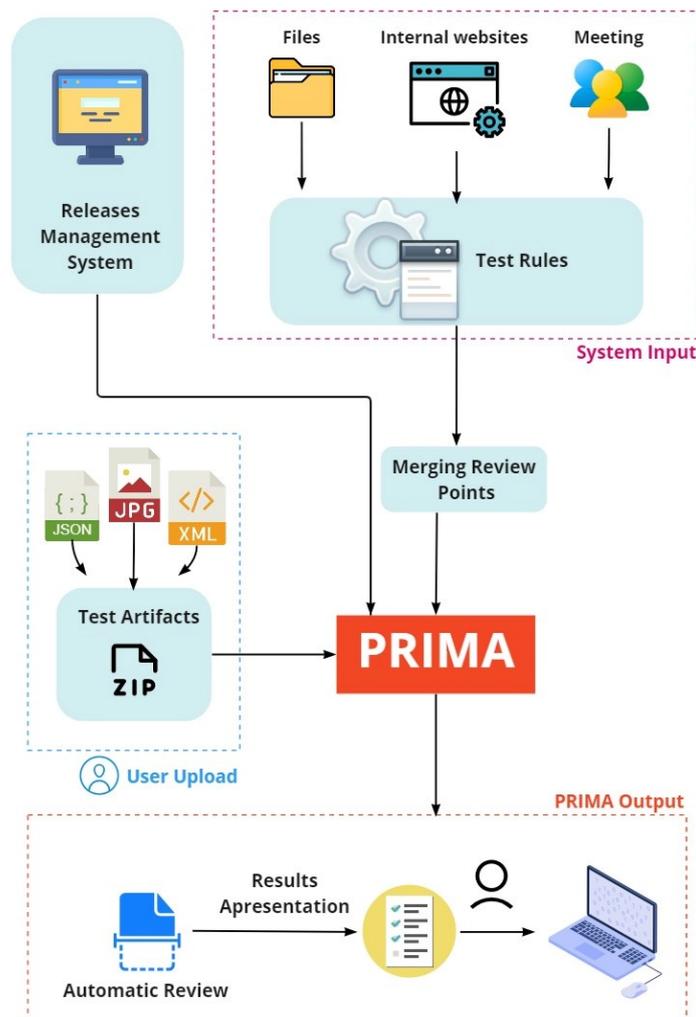
proporção de 15,18% de retrabalho, como mostra a Figura 3.

Já no primeiro semestre de 2022, temos 1941 resultados submetidos, sendo que 346 deles apresentaram inconsistências, sendo assim uma taxa de 17,82% de retrabalho, como mostra a Figura 4.

3 METODOLOGIA

O presente trabalho foi uma pesquisa aplicada a uma implementação, cujo objetivo foi a realização de uma pesquisa exploratória e explicativa sobre o material bibliográfico e laboratorial adquirido neste curso. Os procedimentos técnicos utilizados foram os de pesquisa bibliográfica e experimental. Como método de abordagem, utilizou-se o hipotético-dedutivo e a elaboração seguirá o método de procedimento monográfico. A coleta de dados foi feita através da observação direta intensiva e documentação indireta, sendo estes dados qualitativos e interpretados de forma global.

Figura 5 – Arquitetura da ferramenta de automação PRIMA.



Fonte: Klirssia Sahdo.

Em um primeiro momento, tem-se a ferramenta de implementação em estudo, ela

seguirá conformidade à arquitetura da figura 5, composta pelos seguintes campos:

Entrada do Sistema (*System Input*): para estabelecer os valores esperados, é necessário fazer uma pesquisa em fontes de dados heterogêneas. Esses dados vêm de várias fontes, por exemplo, documentações internas, sites com ferramentas de motores e atualizações de banco de dados que se referem a informações intrínsecas aos modelos dos fabricantes. Uma vez obtidas essas informações, um processo de transformação dos dados é executado, para padronização de dados. Por fim, organiza-se tudo em um banco de dados para consultoria de ferramentas.

Sistema de Gerenciamento de *Releases* (*Releases Management System*): este portal é um gerenciador de projetos que contém as informações de que o testador precisa a respeito das *releases* a serem testadas, a PRIMA faz uma consulta a este portal para validar informações do projeto que está relacionado aos artefatos de teste que foram inseridos pelo usuário na etapa de *User Input*.

Envios do Usuário (*User Upload*): os usuários enviam seus resultados compactados em um arquivo no formato ZIP para a ferramenta. Em seguida, a PRIMA revisa os arquivos dos resultados, comparando todos os resultados extraídos com os valores esperados previamente inseridos no banco de dados.

Resultado da PRIMA (*PRIMA Output*): como interface, os resultados serão exibidos em formato de lista, com a principal função de indicar qual arquivo apresenta inconsistência de regra, assim como sua descrição. Ao identificar a ocorrência, o testador deve aplicar a correção no arquivo correspondente e atualizá-lo na ferramenta para identificar se não houve algum efeito colateral na correção (*side effect*). Desta forma, a ferramenta se encaixa na etapa do processo para minimizar os erros que poderiam passar despercebidos para a revisão da Google.

Após a preparação do banco de dados com todas as informações necessárias, inclui-se a quantidade de regras que precisam ser validadas em cada tipo de teste, e quais são os arquivos de resultados específicos onde cada campo pode ser encontrado. Uma vez implementadas as regras e o banco de dados para comparação, os resultados estão corretos.

o sistema fica online para o usuário verificar os resultados dos seus testes. A ferramenta fica disponível no formato de Webservice, para facilitar o uso da intranet por todos os testadores pertencentes à empresa. Portanto, para verificar seus testes, o usuário faz um upload de arquivo de seus resultados na PRIMA e este por outro lado, passará por todos os arquivos e extrairá os campos necessários para verificação e os armazenará no formato do arquivo JSON, a fim de facilitar a integração com outros serviços que, porventura, necessitem utilizar informações de revisão. Ao final deste processo, os campos do arquivo JSON serão comparados com os resultados do banco de dados e serão validadas todas as regras inseridas. Portanto, resalta-se cada regra que não foi atendida como um erro para o usuário verificar.

A fim de otimizar os testes e amenizar a ocorrência de *Feedbacks*, a ferramenta automatizará a revisão de artefatos de teste durante os lançamentos de homologação para modelos que utilizam o Sistema Android. A ferramenta PRIMA, terá a capacidade de revisar os arquivos de teste gerados por meio de testes executados, interpretando as informações contidas e comparando-as com seus valores esperados, indicando os problemas encontrados.

Em um segundo momento, serão pesquisados métodos de implementação de experimento para grupos limitados de pessoas, no que diz respeito a método de conservação de dados obtidos, investigando-se posteriormente se houve comprometimento. Em seguida, será delimitado o escopo do experimento a ser aplicado, criando-se o roteiro do experimento e pontos de análise da pesquisa aos participantes.

Em um terceiro momento, ocorrerá a aplicação do experimento, que será realizado com um grupo de testadores de um ambiente real de testes, no caso deste trabalho, com uma empresa privada que tem seus serviços de testes com requisitos da Google para dispositivos mobile. E por fim, realizar-se-á a avaliação dos dados obtidos em experimento, respondendo-se as perguntas e hipóteses levantadas inicialmente como objetivo deste trabalho.

4 EXPERIMENTO

O experimento se encontrou em um contexto real de revisão de testes com releases android. Os dados pertinentes à execução deste experimento serão apresentados nas próximas seções.

No contexto de empresa escolhida, identificou-se um histórico recorrente de retrabalhos relacionados a *feedbacks*. Almejando-se identificar se a automação proposta, de fato auxilia na prevenção de inconsistências no processo de homologação de releases Android, assumiu-se que a hipótese é verdadeira, para que assim fosse possível presumir resultados esperados, ou seja, as respostas corretas. Para que então, com os resultados do experimento, seja possível responder as questões de pesquisa propostas.

4.1 DEFINIÇÃO DE PARTICIPANTES

Antes de tudo, é de suma importância, realizar uma avaliação equilibrada de participantes do experimento. Esta observação é indispensável, visto que, os resultados podem mudar dependendo de quem executa o teste, por isso a relevância deste tópico. Então, para execução do experimento, foram selecionados testadores da área de qualidade e teste de software que trabalham com foco na etapa de homologação de releases. Todos os participantes deste experimento não estiveram envolvidos na produção deste trabalho, evitando-se que eles tivessem quaisquer conhecimentos prévios sobre o protocolo de experimento ou das análises que foram levantadas, de forma a mitigar possíveis vieses. Visando distintas formas de trabalho, a seleção dos profissionais se deu conforme os vários níveis de experiência: desenvolvedores de testes acima de 3 anos de atuação no setor; de 2 a 3 anos; de 1 a 2 anos; e com menos de 1 ano.

No total, 12 profissionais foram selecionados dentro da empresa escolhida, uma vez que fosse possível contar com, no máximo, 20% do time disponível, para não impactar resultados na demanda da empresa. Os participantes envolvidos, necessitavam ter compreensão da experiência e consentimento de sua participação, portanto, um termo de consentimento foi disponibilizado. Com isto, cada colaborador do experimento rece-

beu detalhes pertinentes à sua participação, explicando os objetivos do experimento e destacando-se confidencialidade de informações com relação à identificação de cada um, assim como o sigilo referente aos dados individuais obtidos.

4.2 ESCOLHA DOS TESTES

Com o objetivo de identificar quais as inconsistências mais comuns encontradas nos artefatos de teste de uma release de homologação, realizou-se uma consulta aprofundada ao banco de dados da empresa cenário deste estudo para o ano de 2021. Dentre os dois anos de possível base para o experimento, escolheu-se o primeiro, pois no período de 2022 surgiram novos pontos de revisão. Assim será possível fazer uma breve comparação das mesmas inconsistências entres os dois anos. Observou-se os testes realizados para avaliar os erros que mais levaram ao *Feedback*. Para poder classificar melhor essas informações, dividiu-se as inconsistências encontradas em grupos relacionados ao tipo de ocorrência, eles podem ser observados conforme a Tabela 1 de descrição e a Figura 6 com a classificação em números. Com esta análise, tanto será possível atingir o primeiro objetivo específico deste trabalho, como direcionar quais erros inserir no teste experimental da ferramenta.

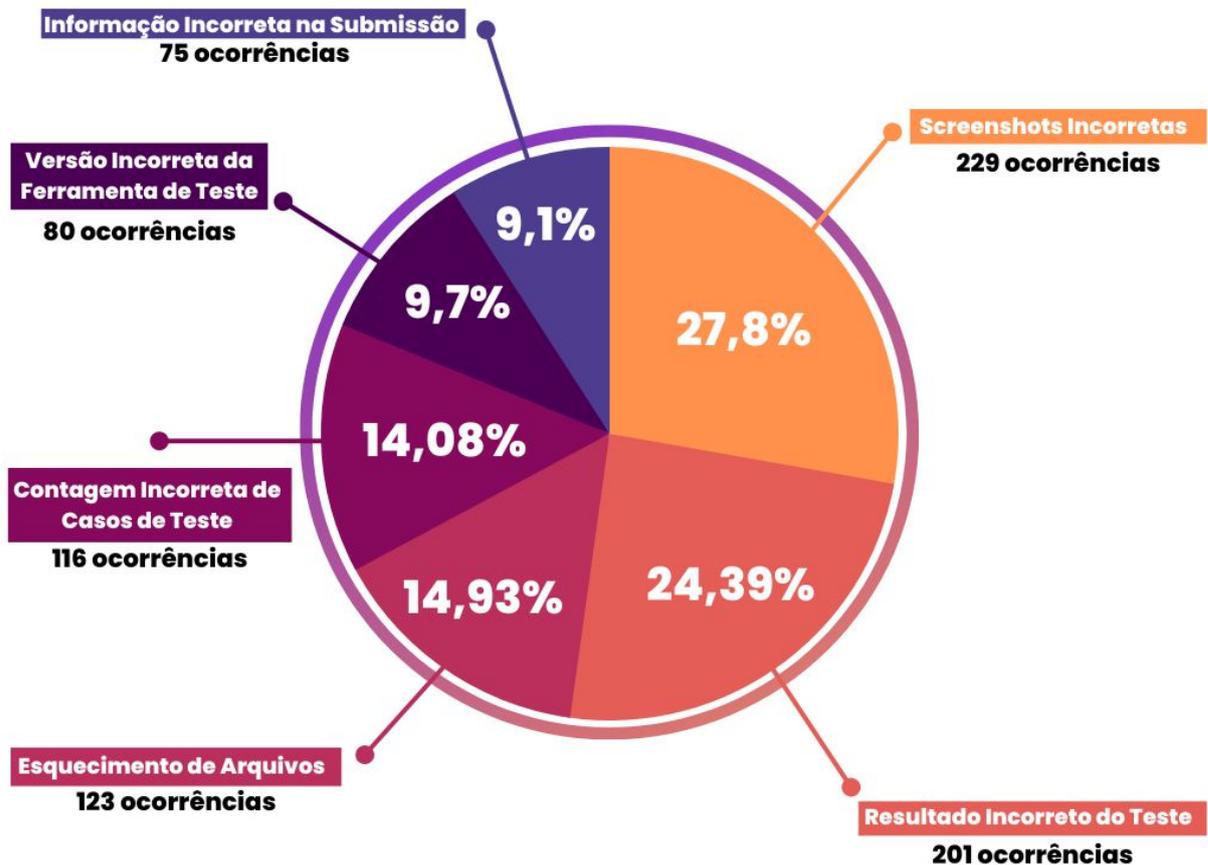
Tabela 1 – Ocorrências Mais Frequentes de 2021.

Inconsistências	Descrição
Screenshots Incorretas	Quaisquer erros em avaliação de imagens
Resultado Incorreto do Teste	Divergência de informação com a release
Esquecimento de Arquivos	Arquivo não encontrado dentro dos resultados
Contagem Incorreta	Contagem total de casos de testes fora do esperado
Versão Incorreta da Ferramenta	Ferramenta de teste fora da versão mais atual
Informação Incorreta	Erro na submissão dos resultados à Google

Fonte: Sidia, 2021.

Screenshots Incorretas: quaisquer erros em testes relacionados à avaliação de imagens, como o ícones de aplicativos, layout de aplicativos, coloração, ordem de apresentação e presença ou falta deles. A frequência deste tipo de falha se dá tanto pelo fato de que os dispositivos Android abrangem uma ampla contribuição para criação de diferentes tipos de produtos como também pela customização do fabricante do modelo (que altera

Figura 6 – Classificação de Inconsistências mais Recorrentes em 2021.



Fonte: Sidia, 2021.

conforme o modelo e versão android embarcada).

Resultado Incorreto do Teste: resultados onde há alguma divergência de informação com os dados esperados da release, estes por sua vez, dizem respeito a atenção e conhecimento do desenvolvedor do teste, acaba sendo comum pelo fato de que há modelos no mercados mobile que são muito parecidos, divergindo em pequenos detalhes.

Esquecimento de Arquivos: omissão de algum artefato de teste necessário para a homologação da release. Por muitas vezes, pontos de revisão mais simples são os que também correm o risco de faltar. Neste quesito se dá a situação em que o desenvolvedor, por vezes, tão acostumado com a atividade, vem a não dar ênfase aos pontos de revisão mais fácil ou de menor demanda (trabalho).

Contagem Incorreta: número total de casos de teste de qualquer ferramenta não foi executado em sua totalidade. Neste ponto, há a real e maior preocupação, conferir

todos os testes exigidos para teste pela Google. Como as ferramentas de teste utilizadas, em grande maioria, são da Google, em seus sistemas de código, há uma dificuldade, que não pode ser solucionada pelos desenvolvedores de teste, pois estes não são os ofertantes dela. Muitas vezes, mesmo executando todos os testes, as ferramentas deixam de executar alguns casos de teste. E outro fator para essa contagem, também ocorre quando o testador, pode não ter realizado todos os casos de teste exigidos.

Versão Incorreta: utilização das ferramentas na versão mais atual, ou seja, no escopo correto para os testes atuais são os pré-requisitos dos desenvolvedores de teste em um setor de Google Approval. Como ocorrem atualizações de software semanalmente e ainda de ferramentas, para abranger as novas aplicações, pode-se esquecer de atualizar o ambiente de teste com a ferramenta adequada.

Informação Incorreta: sobre qualquer erro durante a submissão dos resultados para o Google, isso se dá pela etapa final do desenvolvimento de teste, pois o testador pode de forma excelente verificar todos os testes, confirmar a correta funcionalidade dos dispositivos, mas ao enviar estas informações e arquivos para a Google, pode vir a errar tanto a forma de entrega dos arquivos como também o preenchimento de informações, fase final que demanda total atenção.

E como um breve comparativo, dentre as 346 submissões com feedback do primeiro semestre de 2022 (Figura 4), houveram 402 ocorrências visto que um feedback pode conter uma ou mais ocorrências, dentre elas cento e quarenta e seis delas, resultante a 36,32%, corresponderam a Resultado Incorreto do Teste, sendo a inconsistência que mais ocorreu durante esse período. Com oitenta casos, cerca 19,9%, corresponde a Screenshot Incorreta. Esquecimento de Arquivos, correspondeu a 17,91%. E para Contagem Incorreta de Casos de Testes, houveram 51 casos, representando 12,69% do todo. Além destes pontos, outros, como Versão Incorreta e Informação Incorreta, apresentaram 6,96% e 6,22% respectivamente.

Este estudo serviu de base para quais seriam os erros inseridos manualmente nos testes A e B que farão parte do experimento, conforme a Tabela 2. O teste A possui um total de 533 arquivos, 89 pastas e um total de 9 inconsistências inseridas manualmente.

Por sua vez, o teste B possui 554 arquivos, 101 pastas e um total de 6 inconsistências adicionadas manualmente. O resultado esperado, será melhor detalhado nos resultados.

Tabela 2 – Ocorrências Inseridas no Experimento.

Teste	Quantidade	Ocorrências
Teste A	4	Esquecimento de Arquivos
	2	Resultado Incorreto
	1	Screenshot Incorreta
	1	Informação Incorreta
	1	Versão Incorreta
Teste B	2	Esquecimento de Arquivos
	1	Screenshot Incorreta
	3	Resultado Incorreto

Fonte: Klirssia Sahdo.

4.3 ORGANIZAÇÃO DO EXPERIMENTO

Foram criados dois cenários para teste: cenário I - revisão de testes com uso da ferramenta PRIMA; e, cenário II - revisão de testes sem o uso da ferramenta PRIMA. A justificativa para a utilização destes cenários é a necessidade de responder nossa principal hipótese deste trabalho relacionada a eficácia da ferramenta de revisão de testes, identificando as principais diferenças entre um grupo que utilize a abordagem no contexto de revisão e outro grupo que não a utilize a fim de conflitar seus resultados. Em cada cenário, foram alocados seis participantes, com o total de 12 pessoas contabilizando ambos os cenários. Independente do cenário, os testadores revisaram dois testes, chamados de teste A e outro de teste B, sendo estes, provenientes de um contexto real de homologação. Para captar a resposta e interpretação dos testes aplicados, formulários de resposta foram disponibilizados, contendo campos para que o usuário preenchesse suas observações, padronizando e facilitando o processo de coleta de dados. Estes dados serão abordados nos resultados, quais foram tais descobertas baseadas nas respostas de todos os participantes.

Definiu-se o tempo de duas horas por pessoa em cada cenário, simulando prazos de um teste real. Conforme a formação de grupos planejada, os testadores foram organizados de acordo com a disposição de níveis de atuação na profissão, além de aplicar o mesmo

conjunto de testes nos dois cenários. Essas delimitações, colaboram para manter o ambiente controlado, acerca da análise de um mesmo conjunto de aplicação e, assim, realizar com melhor identificação a influência do uso da ferramenta durante a revisão de testes.

4.4 PASSOS DO EXPERIMENTO

Ao iniciar o processo, cada participante convidado dirigiu-se à sala de experimento portando consigo seu computador pessoal de trabalho, assim disponibilizou-se termo de consentimento para leitura. Ao finalizar todo o experimento, de acordo com o cenário disposto, os dois grupos de participantes avaliaram em um formulário suas experiências, opiniões com relação à usabilidade da ferramenta de automação PRIMA no processo de revisão de testes.

4.4.1 Cenário A - Revisão sem a ferramenta PRIMA

Inicialmente, cada participante com auxílio da leitura do documento de termo de consentimento, seguiu orientação à sala de experimento com seus itens pessoais de trabalho necessários. Disponibilizou-se consultas a alguns materiais, com exceção da ferramenta PRIMA, como por exemplo a documentação geral do time, onde estão indicados todos os pontos de revisão pertinentes aos artefatos de teste, conforme o normal cenário de trabalho. Depois, compartilhou-se o primeiro teste tipo A para revisão, solicitando-se o acesso ao formulário respectivo de resposta, que também estaria controlando o tempo que cada colaborador levaria para completar a atividade. Finalizando-se a primeira revisão, compartilhou-se o teste tipo B, repetindo-se o mesmo processo, tudo dentro do limite de duas horas.

4.4.2 Cenário B - Revisão com a ferramenta PRIMA

Neste segundo cenário, após a leitura do documento de termo de consentimento, o processo ocorreu de maneira similar ao cenário A, porém com a diferença de apresentação do funcionamento da ferramenta de revisão e não haver limite de acesso à ela como suporte. Também, foi definida a obrigatoriedade de uso da ferramenta PRIMA em qualquer etapa

do processo de revisão dos testes, neste caso, antes ou após a revisão de cada caso, o que ficava a critério do tester. Houve o compartilhamento do teste tipo B, e ao fim, os formulários foram organizados para análise e investigação das respostas.

5 RESULTADOS

Os resultados foram obtidos de acordo com os dados gerados no experimento, de forma que as análises deles serão abordadas segundo as ideias do que se esperava que ocorresse comparado com o que, de fato, definiu os cenários experimentais, além de comentar o que possa ter gerado tal conclusão. As questões explanadas dirão respeito às estatísticas alcançadas e ilustrações comportamentais do experimento, almejando-se concluir o desfecho da hipótese deste estudo e avaliar se os objetivos foram atingidos. Os resultados apresentados nesta seção são provenientes das respostas dos formulários utilizados pelos participantes no experimento.

5.1 ANÁLISE DOS RESULTADOS

As inconsistências encontradas por cada revisor (*tester*) variaram bastante, conforme a Tabela 3, dividida nos dois cenários, em que o cenário I (Sem PRIMA) é representado pelos *testers*: A, B, C, D, E e F, que não utilizaram a automação de revisão; e o cenário II (Com PRIMA), em que a ferramenta foi utilizada, apresentando os *testers*: G, H, I, J, K e L. Na penúltima linha, constam as falhas identificadas pela PRIMA; e, por último, o Resultado Esperado, que é o campo com o registro da totalidade de falhas que foram inseridas, ou seja, o resultado esperado. Seus respectivos resultados de acurácia em relação ao Ground Truth estão indicados em cada coluna do teste A, B e Total, onde fazemos o agrupamento dos resultado de ambos testes.

No resultado do tester E, que mesmo com menos de 1 ano de experiência, este obteve um desempenho melhor do que os participantes A e C, que possuem mais de 3 anos de experiência, nenhum destes três participantes citados fizeram uso da automação, revisando de forma totalmente manual. Abordando-se os testes separadamente, houveram casos em que no teste B, um participante fez uso da PRIMA e identificou todas as falhas no resultado, enquanto que outro participante, sem o uso da automação no mesmo tipo de teste, não conseguiu identificar nenhuma das falhas, estando ambos no mesmo nível de experiência. Notou-se que dentre os participantes com a automação, o participante

Tabela 3 – Comparação dos Resultados do Experimento da Ferramenta PRIMA.

Cenário	Participante	Resultado A	Resultado B	Total
Sem PRIMA	Tester A	3 (33,3%)	1 (16,6%)	4 (26,6%)
	Tester B	2 (22,2%)	1 (16,6%)	3 (20%)
	Tester C	3 (33,3%)	0 (0%)	3 (20%)
	Tester D	4 (44,4%)	2 (33,3%)	6 (40%)
	Tester E	5 (55,5%)	4 (66,6%)	9 (60%)
	Tester F	5 (55,5%)	4 (66,6%)	9 (60%)
Com PRIMA	Tester G	4 (44,4%)	4 (66,6%)	8 (53,3%)
	Tester H	6 (66,6%)	6 (100%)	12 (80%)
	Tester I	7 (77,7%)	4 (66,6%)	11 (73,3%)
	Tester J	3 (33,3%)	4 (66,6%)	7 (46,6%)
	Tester K	6 (66,6%)	5 (83,3%)	11 (73,3%)
	Tester L	3 (33,3%)	2 (33,3%)	5 (33,3%)
Base	PRIMA Eficácia	7 (77,7%)	5 (83,3%)	12 (80%)
	Resultado Esperado	9 (100%)	6 (100%)	15 (100%)

Fonte: Klirssia Sahdo.

L identificou apenas 5 das 15 falhas do resultado geral, que foi o pior resultado com automação, talvez por ser um tester de nível iniciante. Isso diminuiu o desempenho geral dos membros que trabalharam na revisão com a PRIMA, em contrapartida, mesmo este participante obtendo menor desempenho ao encontrar as falhas nos testes do seu respectivo cenário B, ele representou melhor apontamento das inconsistências do que metade dos participantes que não tiveram acesso à automação (cenário I). Seguindo esta ideia, ter mais experiência na profissão não é necessariamente um fator para realizar uma revisão mais eficaz, como também, mesmo com tal capacidade, uma revisão totalmente manual não manifesta tanta segurança no envio dos resultados.

Houve duas falhas de Resultados Incorretos que não foram identificadas por nenhum participante sem a aplicação da PRIMA, enquanto que 66% dos participantes que usaram a PRIMA relataram esses erros. Houve então um ganho de performance, já que ela pôde auxiliar na validação de resultados incorretos em um volume muito grande de informações. Percebe-se que a automação ajudou a prevenir problemas que passaram despercebidos em momentos de intensa revisão quando os participantes fizeram uso da PRIMA, eles relataram que a automação identificou alguma inconsistência da qual eles não haviam percebido durante sua própria revisão. Analisando o fator por falhas reportadas pela

automação e não reportadas pelos testers, surgiu um ponto da dificuldade de se entender o que a automação estava informando. No formulário de usabilidade, questionou-se se os participantes tiveram dificuldade em entender as informações passadas pela PRIMA, 4 dos 6 participantes responderam que tiveram alguma dificuldade em entender o que havia sido apontado na plataforma, logo compreende-se que poderiam haver melhorias de interface com relação a exposição de informação para resolver essa alternativa.

Com relação à quantidade de erros inseridos nos testes para experimento, nenhum dos 12 participantes conseguiu detectar todas as falhas, concluindo-se que utilizar a ferramenta PRIMA, não foi um fator decisivo para entregar um teste com 100% de acerto. A PRIMA é uma ferramenta de auxílio ao testador, mas não necessariamente pode garantir que as correções sejam, de fato, realizadas em totalidade. A avaliação, em média, dos participantes sem uso da PRIMA foi de 5,66 falhas encontradas das 15 existentes em ambos os testes. Enquanto que, a nota média dos participantes que fizeram uso da automação foi de 9 falhas encontradas, sendo um ganho aproximado de 59% para os participantes que usaram a automação auxiliar na revisão. Em relação a esta análise, o grupo utilizando a ferramenta encontrou, em média, 60% das falhas inseridas manualmente, ao mesmo tempo que o outro grupo que não fez uso da ferramenta conseguiu encontrar apenas 37.7%. A maior quantidade de falhas encontradas por membros sem a PRIMA foi observada pelos testers E e F com 9 falhas, ao passo que dos membros com a PRIMA, o participante H teve o melhor desempenho encontrando 12 falhas das 15 existentes.

É possível perceber, a partir desses resultados, que houve uma melhora no desempenho dos participantes que fizeram uso da automação, eles encontraram mais falhas do que aqueles que não a utilizaram. O tempo de realização do experimento em ambos os cenários foi parecido, então não foi visto um ganho imediato no tempo de revisão, possivelmente devido ao fato de a ferramenta apenas apontar os erros, não excluindo a necessidade de uma revisão manual em conjunto. Com a criação da ferramenta, o foco é de que um ganho em relação à qualidade dos resultados fosse possível, além de implicar diretamente no ganho em processos futuros, evitando-se o prolongado processo de aprovação das submissões e minimização de custos relacionados às novas manutenções sobre as inconsistências que

podem surgir posteriormente.

Portanto, baseado nas análises apresentadas nesta seção, há indícios de que a ferramenta PRIMA pode sim auxiliar no processo de revisão de artefatos de teste durante a homologação de releases Android. A hipótese do experimento é confirmada a partir dos dados observados, visto que a ferramenta proposta não conseguiu prevenir os erros em sua totalidade, porém o grupo que fez o uso da plataforma de revisão PRIMA, obteve maior eficácia em encontrar inconsistências nos resultados.

5.2 VALIDADE DOS RESULTADOS

Diante do experimento realizado, alguns pontos de validação foram considerados, tendo em vista que esta pesquisa engloba um tema muito específico que acompanha vieses e pontos de inconsistência, eles implicam diretamente na eficácia do resultado, como contribuição às falhas no processo de homologação de testes. Há quatro tipos de validade dos resultados: validade de conclusão, validade interna, validade externa e validade de construção (*constructo*), cada uma será contemplada conforme as situações em que se caracterizam.

5.2.1 Validade de Conclusão

A validade de conclusão é referente à capacidade de se chegar a uma conclusão correta proveniente dos relacionamentos entre tratamento e o resultado do experimento (MUSTAFA; LABICHE; TOWEY, 2019). Nesse quesito, encontram-se os conceitos de: escolha do teste estatístico; escolha do conjunto de participantes; confiabilidade dos parâmetros; e, da confiabilidade na implementação dos tratamentos. Isso foi observado no momento em que se aplicou um teste, que antes de ser modificado, já tinha passado por alguns colaboradores, dando prévio conhecimento de pontos que não deveriam estar com erros; e, também na quantidade de participantes, que não pôde ser mais expressiva. Como amenização destes efeitos, não se apresentou quaisquer dicas de erros que poderiam ser encontrados e se evitou de comentar sobre o experimento anteriormente, com quaisquer outros profissionais da empresa que não fossem os próprios responsáveis por este trabalho,

que inclusive não fizeram parte do conjunto de testes da ferramenta em experimento.

5.2.2 Validade Interna

A validade interna implica na verificação em que supostamente o resultado alcançado é uma consequência de manipulações e não de qualquer outro índice (KITE; WHITLEY, 2018), no que diz respeito ao modo da aplicação dos tratamentos e aspectos sociais (CLEMENS; IV; MOWEN, 2021). Durante o processo do experimento, observou-se uma mudança de atenção dos participantes na finalização de um teste para o outro, pois o exercício de revisão é algo exaustivo e demanda muita atenção de quem esteja realizando, alguns apresentaram cansaço ou desânimo. Este cenário é muito parecido com o que ocorre no cotidiano dos *testers*, porém não se proibiu que os participantes realizassem pausas, como ir ao banheiro ou tomar água.

5.2.3 Validade Externa

A validade externa é delineada pela generalização das conclusões para outros campos. Aqui, diz-se respeito à interação do tratamento com as pessoas, o lugar e o tempo (MUSTAFA; LABICHE; TOWEY, 2019), nem sempre são os mesmos. Em um ambiente não controlado esses dados podem variar um pouco, pois os testes e as pessoas delegadas a eles variam, além do tempo que cada *tester* pode desempenhar.

5.2.4 Validade de *Constructo*

Já a validade de construção, refere-se ao nível em que interferências podem ser feitas a partir da operação do seu estudo com construções teóricas em que se basearam, ou seja, considera-se os relacionamentos entre teoria e a observação (MUSTAFA; LABICHE; TOWEY, 2019). Algumas pessoas, podem ter se apresentado melhores no processo de experimento do que em um cenário real, por justamente, desta forma, ao serem avaliadas, trabalharem melhor, ou projetarem respostas no processo baseadas no que esperavam do experimento. Protegendo-se a validade dos dados, manteve-se um esforço significativo para que o experimento não parecesse de fato uma prova, ou avaliação que definiria os

participantes, até pelo fato de os dados obtidos serem apresentados nesta pesquisa de forma sigilosa.

6 CONCLUSÃO

Neste Trabalho de Conclusão de Curso foi testada uma abordagem para automatizar a revisão de artefatos de teste ao final do processo de homologação de releases Android. A empresa Sidia Instituto de Ciência e Tecnologia (SIDIA, 2022), componente e cenário de estudo neste material de pesquisa, juntamente à equipe de desenvolvedores de teste no ramo de Engenharia de Software, antes, não possuía mecanismos automatizados que poderiam auxiliar os testadores na revisão dos arquivos dos seus resultados de teste, desta forma a ferramenta testada atua como um agente facilitador, investigando as inconsistências observadas ao final do processo de homologação. A oportunidade de realizar um experimento com profissionais de teste de software em um ambiente real, trouxe resultados significativos: foi possível identificar uma melhoria de 59% em chance de detecção das inconsistências nos resultados finais de teste com o suporte da ferramenta PRIMA. Com a identificação prévia desses *Feedbacks*, um ganho de aproximadamente 1 dia na etapa de aprovação destes projetos pode ser obtida, devido a uma maior agilidade no processo de homologação.

Desta forma, sobre a hipótese inicial "Para reduzir ou prevenir inconsistências encontradas ao final do processo de homologação de *releases* Android, uma automação de revisão de artefatos de teste pode ser capaz de solucionar este problema, realizando um processo de leitura dos dados de cada artefato de teste e comparando com um sistema de informações que contém as condições ou regras de aprovação dos resultados", considera-se que ela é verdadeira. Felizmente, houve indícios de que aplicar a automação proposta na etapa de revisão dos artefatos de teste, de fato, preveniu mais inconsistências e tornou o processo de aprovação de releases mais ágil e assertivo. Consequentemente, dado o fato de ter se conquistado tal resultado, foi possível conduzir um experimento com o uso da ferramenta na participação de desenvolvedores de teste de software e então analisar o desempenho da ferramenta PRIMA comparando o tempo de revisão com o uso da ferramenta PRIMA e sem o uso da ferramenta com sucesso.

Além da principal contribuição deste trabalho que é a criação da ferramenta PRIMA,

outro objetivo específico foi alcançado. É oportuno evidenciar a disponibilização de informações para auxílio gerencial; padronização e classificação de inconsistências identificadas em releases; e, auxílio nas medidas estratégicas para os gestores da empresa reduzirem a ocorrência dos problemas pertinentes ao cotidiano de uma equipe de homologação. Isso graças ao segundo objetivo específico atendido, a apresentação de dados acerca das inconsistências mais comuns encontradas nos artefatos de teste no fim do processo de *releases* de homologação de uma empresa do ramo. Outros fabricantes podem ter problemas distintos aos apresentados pela consulta realizada neste material, porém são informações com baixo grau de divulgação, visto a estratégia adotada por cada fabricante. Neste sentido, torna-se complexo realizar um levantamento mais aprofundado sobre um cenário externo ao nosso. Essa análise pode ser respondida apenas aderente ao contexto da empresa estudada, não podendo estas conclusões serem estendidas sem antes realizar uma análise respectiva a outro cenário respectivo.

REFERÊNCIAS

- ALURE, S.; PURI, R. Firmware designing for android mobile. *INTERNATIONAL JOURNAL*, v. 5, n. 12, 2021.
- ATEŞOĞULLARI, D.; MISHRA, A. Automation testing tools: A comparative view. *International Journal on Information Technologies & Security*, v. 12, n. 4, 2020.
- BERNARDON, C. et al. Improving the software homologation process through peer review: An experience report on android development environment. *International Journal of Computer and Systems Engineering*, v. 16, n. 3, p. 54–60, 2022.
- CLEMENS, W. M.; IV, J. H. B.; MOWEN, T. J. Internal validity. *The Encyclopedia of Research Methods in Criminology and Criminal Justice*, Wiley Online Library, v. 2, p. 489–491, 2021.
- FERME, V.; PAUTASSO, C. A declarative approach for performance tests execution in continuous software development environments. In: *Proceedings of the 2018 ACM/SPEC International Conference on Performance Engineering*. [S.l.: s.n.], 2018. p. 261–272.
- FIGUEIREDO, L. P. de et al. An automatic approach to detect problems in android builds through screenshot analysis. In: *Proceedings of the 37th ACM/SIGAPP Symposium on Applied Computing*. [S.l.: s.n.], 2022. p. 926–932.
- GAROUSI, V. et al. Exploring the industry’s challenges in software testing: An empirical study. *Journal of Software: Evolution and Process*, Wiley Online Library, v. 32, n. 8, p. e2251, 2020.
- GOOGLE. *Trade Federation Overview - Android Source Documentation*. 2020. Disponível em: https://source.android.com/devices/tech/test/_infra/tradefed.
- GOOGLE. *Android Compatibility Program Overview - Android Source Documentation*. 2021. Disponível em: <https://source.android.com/compatibility/overview>.
- GOOGLE. *Flashing Devices - Android Source Documentation*. 2021. Disponível em: <https://source.android.com/setup/build/running>.
- GOOGLE. *Android Compatibility Definition Document - Android Source Documentation*. 2022. Disponível em: <https://source.android.com/compatibility/cdd>.
- KITE, M. E.; WHITLEY, B. E. The internal validity of research. In: *Principles of Research in Behavioral Science*. [S.l.]: Routledge, 2018. p. 277–309.
- LIN, J.-W.; SALEHNAMEADI, N.; MALEK, S. Test automation in open-source android apps: A large-scale empirical study. In: *Proceedings of the 35th IEEE/ACM International Conference on Automated Software Engineering*. [S.l.: s.n.], 2020. p. 1078–1089.
- MISHRA, A.; OTAIWI, Z. Devops and software quality: A systematic mapping. *Computer Science Review*, Elsevier, v. 38, p. 100308, 2020.

MUSTAFA, N.; LABICHE, Y.; TOWEY, D. Mitigating threats to validity in empirical software engineering: A traceability case study. In: IEEE. **2019 IEEE 43rd Annual Computer Software and Applications Conference (COMPSAC)**. [S.l.], 2019. v. 2, p. 324–329.

PIERCE, M. **Software Release Life Cycle (SRLC): Understand The 6 Main Stages**. 2022. Disponível em: <https://theproductmanager.com/topics/software-release-life-cycle/>.

POSSEMATO, A. et al. Trust, but verify: A longitudinal analysis of android oem compliance and customization. In: **2021 IEEE Symposium on Security and Privacy (SP)**. [S.l.: s.n.], 2021. p. 87–102.

SIDIA, S. **Sidia Instituto de Ciência e Tecnologia**. 2022. Disponível em: <https://sidia.com/>.