

UNIVERSIDADE DE PASSO FUNDO
Programa de Pós-Graduação em
Computação Aplicada

Dissertação de Mestrado

GUIA PARA DECOMPOR TAREFAS DE TESTES FUNCIONAIS EM SOFTWARE CROWDSOURCING

FERNANDO LUIZ COSTELLA



UNIVERSIDADE DE PASSO FUNDO
INSTITUTO DE CIÊNCIAS EXATAS E GEOCIÊNCIAS
PROGRAMA DE PÓS-GRADUAÇÃO EM COMPUTAÇÃO APLICADA

**GUIA PARA DECOMPOR TAREFAS DE
TESTES FUNCIONAIS EM SOFTWARE
CROWDSOURCING**

Fernando Luiz Costella

Dissertação apresentada como requisito parcial à obtenção do grau de Mestre em Computação Aplicada na Universidade de Passo Fundo.

Orientador: Prof. Dr. Cristiano Roberto Cervi
Coorientador: Prof. Dr. Alexandre Lazaretti Zanatta

Passo Fundo
2020

CIP – Catalogação na Publicação

C841g Costella, Fernando Luiz
Guia para decompor tarefas de testes funcionais em
software crowdsourcing [recurso eletrônico] / Fernando Luiz
Costella. – 2020.
5.3 MB ; PDF.

Orientador: Prof. Dr. Cristiano Roberto Cervi.
Coorientador: Prof. Dr. Alexandre Lazaretti Zanatta.
Dissertação (Mestrado em Computação Aplicada) –
Universidade de Passo Fundo, 2020.

1. Software - Testes. 2. Crowdsourcing. 3. Análise
hierárquica de tarefas. 4. Decomposição de tarefas. I. Cervi,
Cristiano Roberto, orientador. II. Zanatta, Alexandre
Lazaretti, coorientador. III. Título.

CDU: 004.415.53

Catalogação: Bibliotecária Jucelei Rodrigues Domingues - CRB 10/1569

ATA DE DEFESA DO TRABALHO DE CONCLUSÃO DE CURSO DO ACADÊMICO

FERNANDO LUIZ COSTELLA

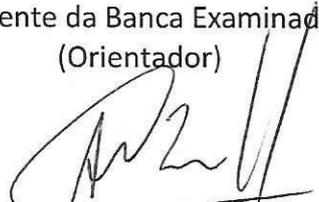
Aos vinte e cinco dias do mês de março do ano de dois mil e vinte, às quatorze horas, realizou-se, no prédio D1 sala 01, da Universidade de Passo Fundo (UPF), a sessão pública de defesa do Trabalho de Conclusão de Curso “Guia para decompor tarefas de testes funcionais em software crowdsourcing”, de autoria de **FERNANDO LUIZ COSTELLA**, acadêmico do Curso de Mestrado em Computação Aplicada do Programa de Pós-Graduação em Computação Aplicada – PPGCA. Segundo as informações prestadas pelo Conselho de Pós-Graduação e constantes nos arquivos da Secretaria do PPGCA, o aluno preencheu os requisitos necessários para submeter seu trabalho à avaliação. A banca examinadora foi composta pelos doutores Cristiano Roberto Cervi, Alexandre Lazaretti Zanatta, Ana Carolina Bertoletti de Marchi, Rafael Prikladnicki. Concluídos os trabalhos de apresentação e arguição, a banca examinadora considerou o candidato APROVADO. Foi concedido o prazo de até quarenta e cinco (45) dias, conforme Regimento do PPGCA, para a acadêmica apresentar ao Conselho de Pós-Graduação o trabalho em sua redação definitiva, a fim de que sejam feitos os encaminhamentos necessários à emissão do Diploma de Mestre em Computação Aplicada. Para constar, foi lavrada a presente ata, que vai assinada pelos membros da banca examinadora e pela Coordenação do PPGCA.



Prof. Dr. Cristiano Roberto Cervi – UPF

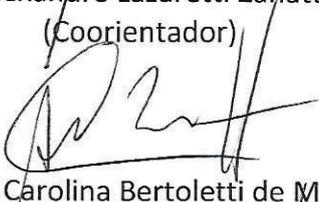
Presidente da Banca Examinadora

(Orientador)



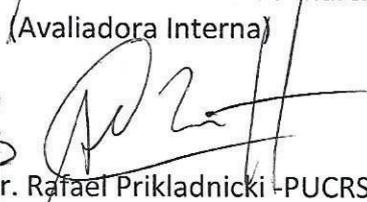
Prof. Dr. Alexandre Lazaretti Zanatta – UPF

(Coorientador)



Prof. Dra. Ana Carolina Bertoletti de Marchi – UPF

(Avaliadora Interna)



Prof. Dr. Rafael Prikladnicki - PUCRS

(Avaliador Externo)



Prof. Dr. Rafael Rieder
Coordenador do PPGCA

AGRADECIMENTOS

O desenvolvimento do trabalho, revelou pessoas que contribuíram de alguma forma no seu decorrer, com isso, gostaria de manifestar alguns agradecimentos.

A condução deste trabalho só foi possível graças a meus orientadores, professor **Alexandre Lazaretti Zanatta**, que dirigiu todo o trabalho, contribuindo de forma imensurável por meio de suas revisões, conhecimento, considerações e correções sempre pertinentes. Também a **Cristiano Roberto Cervi** que dispôs seu tempo com leituras e observações.

Aos professores **Ana Carolina Bertoletti De Marchi** e **Rafael Prikladnicki**, membros das bancas de qualificação e defesa, pela leitura e oportunas observações, críticas e sugestões.

Agradeço a **Matheus Bianchi Godinho** pela cooperação no início do trabalho. Também a professora **Tainara Bernardi**, que juntamente ao meu orientador, colegas **Angélica Meneghini Boff** e **Lucas Altmann**, me proporcionaram a primeira experiência na realização de um experimento, sendo muito importante para a sequência do trabalho.

A minha esposa **Laís Trentin**, pela sua paciência, contribuição e compreensão pelos momentos de ausência que foram dedicados no desenvolvimento do trabalho. Obrigado por permanecer ao meu lado, mesmo sem os carinhos, sem a atenção devida e depois de tantos momentos de lazer perdidos. Obrigado pelo presente diário, por seu sorriso e por me fazer feliz.

A meus pais **Dilse Teresinha Piva Costella** e **Luiz Caetano Costella** pelo apoio no decorrer do trabalho e pelos ensinamentos da importância da educação na vida.

Aos **professores, amigos, colegas** e **familiares** que direta ou indiretamente contribuíram. A Deus por ter me dado saúde e força para superar as dificuldades.

GUIA PARA DECOMPOR TAREFAS DE TESTES FUNCIONAIS EM SOFTWARE CROWDSOURCING

RESUMO

Os participantes de ambientes crowdsourcing enfrentam problemas, os quais são considerados barreiras para efetivação de sua contribuição. Uma das dificuldades apresentadas é a complexidade das tarefas disponibilizadas. O trabalho apresenta uma abordagem para reduzir esta complexidade, por meio da decomposição de tarefas. Ao contrário dos trabalhos relacionados que demonstram a realização da decomposição pela plataforma, a proposta prevê que seja o usuário a realizar tal procedimento. Para isso, foi desenvolvido um guia para decomposição de Macrotasks, incorporando em sua elaboração os padrões identificados nas tarefas pela análise de conteúdo, a técnica de Análise Hierárquica de Tarefas e o modelo de Melhoria do Processo de Teste. Para avaliação do guia, foi realizado um experimento pareado, com alunos selecionados por conveniência da disciplina de Teste de Software da Universidade de Passo Fundo. O experimento foi dividido em duas etapas: na primeira, foram realizadas tarefas de teste de software sem a utilização do guia e na segunda, com tarefas diferentes da Fase 1, utilizando o guia proposto. O experimento demonstrou resultados pouco significativos na análise quantitativa e significativos na análise qualitativa. Conclui-se que mesmo a análise quantitativa não tenha apresentado resultados relevantes, a análise qualitativa demonstrou uma boa aderência dos participantes, demonstrando a importância e relevância do guia.

Palavras-Chave: Crowdsourcing, Decomposição de tarefas, Guia para decomposição, HTA, MPT.

GUIDE TO DECOMPOSING FUNCTIONAL TESTING TASKS IN CROWDSOURCING SOFTWARE

ABSTRACT

Participants in crowdsourcing environments face problems, which are considered barriers to make their contribution effective. One of the difficulties presented is the complexity of the tasks provided. The work presents an approach to reduce this complexity, by decomposing tasks. Unlike related works that demonstrate the decomposition by the platform, the proposal foresees that it is the user who performs this procedure. For this, a guide for decomposing Macrotasks was developed, incorporating in its elaboration the patterns identified in the tasks by content analysis, the Hierarchical Task Analysis technique and the Test Process Improvement model. For the evaluation of the guide, a paired experiment was carried out, with students selected for convenience of the Software Testing discipline at the University of Passo Fundo. The experiment was divided into two stages: in the first, software testing tasks were performed without using the guide and in the second, with different tasks from Phase 1, using the proposed guide. The experiment showed insignificant results in the quantitative analysis and significant results in the qualitative analysis. It is concluded that even though the quantitative analysis did not present relevant results, the qualitative analysis demonstrated a good adherence of the participants, demonstrating the importance and relevance of the guide.

Keywords: Crowdsourcing, Task decomposition, Decomposition guide, HTA, MPT.

LISTA DE FIGURAS

Figura 1.	Atores em Crowdsourcing	17
Figura 2.	Barreiras na colaboração	20
Figura 3.	Exemplo de Diagrama Sequencial	21
Figura 4.	Etapas da Análise Hierárquica da Tarefa	23
Figura 5.	Exemplo de Decomposição Hierárquica	24
Figura 6.	Fases do desenho de pesquisa	31
Figura 7.	Teste para distribuição normal	42
Figura 8.	Passos para realização de decomposição de uma tarefa	46
Figura 9.	Estrutura após realizar decomposição	48
Figura 10.	Decomposição tarefa validar arquivos	53
Figura 11.	Decomposição tarefa tipos de ambiente	54
Figura 12.	Decomposição tarefa tipos de ambiente	54
Figura 13.	Distribuição t student	57
Figura 14.	Respostas referente dificuldade nas execuções das tarefas	59
Figura 15.	Respostas da consideração dos participantes a respeito da complexidade das tarefas	59
Figura 16.	Respostas sobre apresentação das tarefas	59
Figura 17.	Respostas sobre a documentação das tarefas	60
Figura 18.	Respostas sobre o auxílio do guia na decomposição	60
Figura 19.	Respostas sobre o auxílio do guia na compreensão da tarefa	61
Figura 20.	Respostas sobre o auxílio do guia e a decomposição na resolução das tarefas	61
Figura 21.	Respostas sobre o motivo da desistência na resolução das tarefas	62

LISTA DE TABELAS

Tabela 1.	Representações gráficas das fases da análise de tarefas	21
Tabela 2.	Análise hierárquica de tarefas em formato tabular	22
Tabela 3.	Níveis da MPT	26
Tabela 4.	Comparação entre os trabalhos relacionados	29
Tabela 5.	Exemplo de tratamento pareado para resolução das tarefas	39
Tabela 6.	Detalhamento da estrutura da tarefa	45
Tabela 7.	Número de tarefas submetidas na Rodada 1	51
Tabela 8.	Número de tarefas submetidas na Rodada 2	53
Tabela 9.	Tarefas submetidas fases 1 e 2	55
Tabela 10.	Recorte da tabela da distribuição t-Student	57

LISTA DE SIGLAS

FBI – Federal Bureau of Investigation

UPF – Universidade de Passo Fundo

HTML – HyperText Markup Language

CSS – Cascading Style Sheets

SGBD – Sistema de gerenciamento de banco de dados

MPT – Melhoria do Processo de Teste Brasileiro

PET – Projeto e Execução de Teste

HTA – Análise Hierárquica de Tarefas

MPT – Melhoria do Processo de Teste Brasileiro

PET – Projeto e Execução de Teste

SUMÁRIO

1	INTRODUÇÃO	13
1.1	MOTIVAÇÃO	13
1.2	OBJETIVOS	14
1.3	ORGANIZAÇÃO DO TEXTO	15
2	FUNDAMENTAÇÃO TEÓRICA	16
2.1	CROWDSOURCING	16
2.2	SOFTWARE CROWDSOURCING	17
2.2.1	Crowd testing	18
2.2.2	Plataformas	18
2.2.3	Barreiras para contribuição dos usuários	20
2.3	TÉCNICAS DE DECOMPOSIÇÃO DE TAREFAS	20
2.4	MELHORIA DO PROCESSO DE TESTE BRASILEIRO	24
2.4.1	Nível de Maturidade	25
2.4.2	Áreas do processo do modelo MPT	25
3	TRABALHOS RELACIONADOS	28
4	MATERIAIS E MÉTODOS	31
4.1	ANÁLISE DE CONTEÚDO	32
4.1.1	Pré-análise	32
4.1.2	Exploração do material	34
4.1.2.1	Codificação	34
4.1.2.2	Categorização	35
4.1.3	Tratamento dos resultados e interpretação	35
4.1.3.1	Inferência	35
4.2	AVALIAÇÃO DA TÉCNICA DE DECOMPOSIÇÃO E MELHORIA DO PROCESSO DE TESTES	35
4.3	EXPERIMENTO	36
4.3.1	Organização do experimento	36
4.3.1.1	Seleção das amostras	37
4.3.1.2	Design do experimento	38

4.3.1.3	Instrumentação	39
4.3.2	Metodologia e preparação para o experimento	39
4.4	ANÁLISE DO EXPERIMENTO	41
4.4.1	Análise quantitativa	41
4.4.2	Análise Qualitativa	42
5	RESULTADOS	44
5.1	ANÁLISE DE CONTEÚDO	44
5.2	GUIA PARA DECOMPOSIÇÃO DE TAREFAS	44
5.2.1	Análise da estrutura da tarefa	45
5.2.2	Validação e identificação individual dos itens da estrutura	45
5.2.3	Passos para realização da decomposição	46
5.2.4	Execução e acompanhamento	48
5.2.4.1	Execução dos cenários mapeados	48
5.2.4.2	Reportar incidentes	48
5.2.4.3	Acompanhamento de bugs	49
5.3	EXPERIMENTO - FASE 1	50
5.3.1	Resolução das tarefas - Fase 1	51
5.4	EXPERIMENTO - FASE 2	51
5.4.1	Resolução das tarefas - Fase 2	52
5.5	ANÁLISE DO EXPERIMENTO	55
5.5.1	Análise quantitativa do experimento	55
5.5.2	Análise qualitativa do experimento	58
5.5.3	Questionário Fase 1	58
5.5.4	Questionário Fase 2	60
6	DISCUSSÃO DOS RESULTADOS	63
7	CONCLUSÕES	66
7.1	CONTRIBUIÇÕES	66
7.2	LIMITAÇÕES DO TRABALHO	67
7.3	TRABALHOS FUTUROS	67
	REFERÊNCIAS	68
	APÊNDICE A – Questionário - Rodada 1	72

APÊNDICE B – Questionário - Rodada 2	78
APÊNDICE C – Documento de livre consentimento	83
APÊNDICE D – Protocolo do experimento	85
APÊNDICE E – Tarefas do experimento - Rodada 1	89
APÊNDICE F – Tarefas do experimento - Rodada 2	104
APÊNDICE G – Guia para decomposição de Macrotasks	115

1. INTRODUÇÃO

A inteligência coletiva tem sido explorada nas mais diversas áreas do conhecimento, tendo sua proposta ligada à informação e às tecnologias da informação e comunicação, de modo que tal discussão se mostra pertinente [1]. Com o surgimento da digitalização¹, nos últimos anos testemunhou-se um surto de empreendimento coletivo do processo de produção diferente das formas tradicionais da organização. Movimentos de softwares *open source* demonstraram que a produção pode ser realizada fora da organização, sendo possível aproveitar talentos espalhados de forma global [2].

Um dos eventos notáveis da chamada Web 2.0 não são as ferramentas em si, mas as maneiras pelas quais as novas tecnologias redesenham os relacionamentos entre as pessoas e as organizações. Conforme Brabham [3], a internet vem sendo um local de cultura participativa e no início dos anos 2000, viu-se pela primeira vez o interesse das organizações para alavancar a inteligência coletiva por parte das comunidades on-line. Isso foi realizado com o propósito de servir às metas de negócio, melhorar a participação do público em governança, desenvolver produtos e resolver problemas. Empresas, agências governamentais e organizações sem fins lucrativos, integraram as energias criativas das comunidades on-line no seu dia-a-dia e, também, muitas organizações foram inteiramente construídas dessa forma. Essa mistura de processos criativos com metas organizacionais é chamada de *crowdsourcing* [3].

1.1 MOTIVAÇÃO

Os atributos que a internet proporcionou ocasionaram mudanças fundamentais nos modelos de negócio. O *crowdsourcing* se destaca por velocidade, alcance e barreiras enfrentadas por seus membros, diferindo dos demais processos na solução de problemas e produção coletiva [4]. Seus participantes precisam estar motivados, pois é de fundamental importância a participação da multidão para realização de projetos em *crowdsourcing* [3].

A conclusão das tarefas propostas, tem influência na continuação dos participantes nas plataformas. Uma pesquisa com usuários do Topcoder² verificou que, de modo geral, em uma amostra de 3403 usuários, os participantes não ganharam nenhuma competição disponibilizada [5], ocasionando a desistência do trabalhador em buscar novas tarefas nas plataformas [4].

Mao et al. [6] pontuam que as tarefas disponibilizadas nas plataformas de *crowdsourcing*, por vezes, são complexas, exigindo uma carga de trabalho considerável e foco

¹Ato de transformar um documento físico em um documento digital, tornando dinâmica sua disponibilidade.

²Plataforma que faz o elo entre o solicitante das tarefas e a multidão.

na realização das atividades. Além disso, um nível de habilidade elevado é exigido para realização das tarefas, podendo limitar o número de trabalhadores em potencial. Latoza et al. [7] definem que uma tarefa é uma solicitação pequena e independente para que um trabalho seja concluído, exigindo um mínimo de tempo e esforço cognitivo, sendo que essas tarefas podem resultar em grandes projetos quando combinadas.

A fim de aumentar o paralelismo dessas atividades, é fundamental a realização de decomposição das tarefas em software *crowdsourcing*. Em seu estudo, Latoza et al. [7] verificam que quando decompostas em tarefas menores, os sistemas de *crowdsourcing* permitem que trabalhadores com diferentes habilidades concluam as tarefas com maior agilidade por meio da paralelização e da larga escala das atividades. Ao ser aplicada essa abordagem ao desenvolvimento de software, a participação tende a aumentar e, com isso, a reduzir as barreiras e o tempo de realização da tarefa. Como estratégia de qualidade, a mesma tarefa pode ser disponibilizada a vários trabalhadores para obter um melhor resultado, sendo a escolha por votação ou pela utilização de critério definido pelo solicitante [8].

Morris et al. [9] e Tajedin e Nevo [10], da mesma forma, verificaram que a alternativa para transformar a complexidade das tarefas é a realização da decomposição, aumentando a perspectiva de bons resultados, porém essa atividade não é trivial [6]. Dessa forma, utiliza-se a decomposição para permitir que a multidão conclua as tarefas com agilidade [7]. Um estudo da Topcoder indicou que a decomposição de tarefas está entre os principais desafios e preocupações nos ambientes *crowdsourcing* [6] [11] [12] [13].

Na tentativa de reduzir as barreiras relacionadas às tarefas (Ver seção 2.2.3) e com o intuito de auxiliar as pessoas que participam das plataformas de *crowdsourcing* a submeterem um número maior de tarefas, foi desenvolvido um estudo com objetivo de criar um guia para realização da decomposição de tarefas de testes funcionais.

1.2 OBJETIVOS

Tendo em vista a complexidade das tarefas disponibilizadas em *crowdsourcing* [6] e o desafio de encontrar uma decomposição adequada [8] [11], o objetivo deste trabalho foi auxiliar os participantes de plataformas de *crowdsourcing* a submeterem um maior número de tarefas. Para isso, foi desenvolvido um estudo destinado à criação de um guia para decomposição de tarefas de testes funcionais, tendo sido analisadas as lacunas e determinados os seguintes objetivos específicos:

- *Identificar técnicas*: identificação e mapeamento das técnicas de decomposição de tarefas utilizadas no software tradicional e no software *crowdsourcing*.
- *Comparar entre as técnicas*: realização de comparativo entre as técnicas de decomposição estudadas, avaliando suas características.

- *Adaptar técnica*: adaptação da técnica de decomposição de tarefas para auxiliar os trabalhadores na submissão de tarefas de testes funcionais.
- *Criar guia*: criação de guia, incorporando a técnica adaptada para auxílio na decomposição de tarefas.
- *Realizar experimento*: Realização de experimento utilizando o guia proposto.
- *Avaliar experimento*: avaliação dos resultados do experimento para identificar se o guia auxiliou na realização da decomposição e na submissão das tarefas.

1.3 ORGANIZAÇÃO DO TEXTO

Os próximos capítulos deste trabalho buscam elencar apontamentos para a diminuição das barreiras de usuários que contribuem em cenários *crowdsourcing*, tratando da decomposição de tarefas. Para isso, o trabalho está organizado com a seguinte estrutura:

- O Capítulo 2 apresenta a revisão dos temas abordados, sendo iniciado pela fundamentação de *crowdsourcing* e *software crowdsourcing*, seguido por técnicas de decomposição de tarefas e abordando, ainda, o MPT.
- O Capítulo 3 traz uma exposição dos trabalhos relacionados.
- O Capítulo 4 expõe os métodos e os procedimentos empregados para condução da investigação do trabalho, descrevendo os protocolos e as operações.
- O Capítulo 5 descreve os resultados obtidos na execução do trabalho. São descritos os procedimentos para criação do guia e o experimento para avaliá-lo, sendo realizada a análise dos resultados.
- O Capítulo 6 discute os resultados obtidos.
- O Capítulo 7 conclui o trabalho com uma síntese geral das contribuições e limitações, com apontamentos para trabalhos futuros.

2. FUNDAMENTAÇÃO TEÓRICA

Neste capítulo será apresentada a fundamentação teórica utilizada para desenvolvimento do trabalho.

2.1 CROWDSOURCING

Comunidades online são fontes férteis de criatividade e talento. De acordo com Brabham [3], pesquisas acadêmicas para verificar como e porque o *crowdsourcing* tem evoluído cresceram consideravelmente. O termo foi utilizado por Howe [2] o qual verificou em sua pesquisa que as empresas estavam utilizando o modelo para criar uma rede de trabalhadores anônimos para o desenvolvimento de atividades. Essas atividades eram pre-estabelecidas e disponibilizavam uma remuneração menor em comparação a empregados tradicionais. Empresas assumiram funções que antes eram desempenhadas por funcionários, realizando a distribuição para outros trabalhadores em uma chamada aberta para a comunidade on-line [3] [14]. O *crowdsourcing* foi um encaixe adequado, pois transformou dois conceitos: a terceirização e uma multidão de trabalhadores on-line [3].

Conforme Brabham [3] e Mao et al. [6], as primeiras tentativas acadêmicas para definir o *crowdsourcing* iniciaram em 2008. Essas definições eram conflitantes e explicavam o *crowdsourcing* de acordo com os participantes, pelas razões para participação, ferramentas utilizadas, características organizacionais, pelo grau de complexidade e pela participação do usuário. *Crowdsourcing* é uma história de cooperação, agregação e trabalho em equipe, consenso e criatividade. É uma forma de fazer um trabalho, mas também pode ser considerado um fenômeno no qual, nas condições corretas, os grupos de pessoas podem superar até mesmo especialistas individuais [3].

Com o passar do tempo, foram observadas diversas situações envolvendo a interação da multidão em prol de um objetivo. Um exemplo é a empresa Coca Cola que lançou um desafio ao redor do mundo para que as pessoas enviassem seus métodos para adoçar alimentos e bebidas, baseados em suas culturas. Foram escolhidos cinco vencedores individuais ou em equipes, que competiram por um prêmio total de até 100,000 dólares [15]. Outro exemplo é datado em 2013 quando ocorreu um atentado na maratona de Boston, sendo que ninguém assumiu a responsabilidade pelos ataques. Com isso, o FBI (*Federal Bureau of Investigation*) precisou realizar buscas pelos culpados e pediu a todas as pessoas que estiveram presentes divulgassem suas fotos ou filmagens da maratona, para contribuir com a investigação. As fotos enviadas foram utilizadas na identificação dos responsáveis pelo ocorrido na maratona [16].

Esse cenário também se apresenta na construção do software, estando presente em todas as fases do desenvolvimento. Dessa forma, torna-se necessário pontuar como o desenvolvimento de software *crowdsourcing*.

2.2 SOFTWARE CROWDSOURCING

Com a disseminação da tecnologia uma nova forma de produção tem surgido, diferente das formas tradicionais. Os movimentos de software *open source* demonstraram que a construção do software pode ser realizada fora do ambiente empresarial, por pessoas com diversos talentos e dispostas globalmente. Esse modelo de software *crowdsourcing* atraiu a atenção de empresas, tendo em vista o aproveitamento dos talentos da multidão em seu benefício [10]. Para Hinchcliffe [17], desde que a terceirização se tornou difundida na década de 1990, tornou-se facilitada a externalização das tarefas que antes eram realizadas apenas por um indivíduo, podendo ser realizado algo melhor e mais barato.

Conforme Tajedin e Nevo [10], o software *crowdsourcing* está crescendo ao longo dos anos, de acordo estimativas que apontam um crescimento que pode ultrapassar 100% ao ano. Com esse crescimento, uma comunidade global tem voltado suas atenções para esses fenômenos interconectados de software, hardware, código aberto, inovação, inteligência coletiva e a multidão (sabedoria, poder e “riqueza”). Esse fenômeno rompe com o modo de colaboração global como era baseado. Acredita-se que isso tenha levado às inovações mais importantes das tecnologias que são utilizadas [18].

Hosseini [19] ilustrou conforme pode ser observado na Figura 1, os quatro pilares do processo do *crowdsourcing*, quais sejam: a Multidão que são as pessoas responsáveis pela realização das tarefas propostas; o Solicitante que é o responsável por oferecer uma tarefa, buscando o conhecimento da Multidão; a Tarefa que é a representação da atividade solicitada; e a Plataforma que é a responsável pela comunicação entre o Solicitante e a Multidão. É na plataforma que estão contidas as tarefas disponíveis para realização.

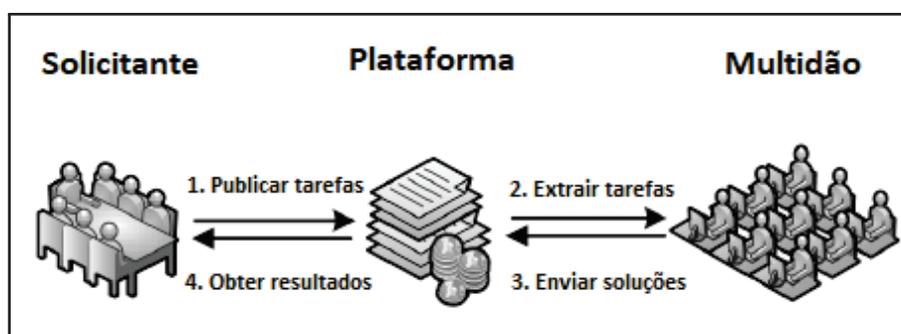


Figura 1. Atores em Crowdsourcing
Fonte: Adaptado de Ke Mao [6] e Hosseini [19]

Inicialmente, a organização que possui um desafio ou problema que necessita ser solucionado realiza a divulgação desse a um público on-line. Esse público trabalha e, após, retorna com a solução do problema. Juntamente com o público, a empresa finaliza e remunera os responsáveis pela solução, gerando um ganho, tanto para a empresa quanto para o público. Nesse cenário, encontram-se empresas que solicitam apenas a validação do software, tendo as plataformas de *crowd testing* que fazem a conexão entre os trabalhadores e a empresa.

2.2.1 Crowd testing

O desenvolvimento de software contempla a especificação de requisitos, projeto, implementação e testes [20]. Nesse cenário o teste funcional, também conhecido como caixa preta, busca verificar as funções do sistema sem se preocupar com sua implementação, tendo em vista os dados de entrada e as respostas produzidas [21]. A qualidade da experiência geral dos usuários tem surgido além da usabilidade, aceitabilidade e do desempenho das tarefas, sendo um foco frequente na realização dos testes. Testes são necessários, porém, custosos tanto no ponto de vista organizacional quanto financeiro [22].

Como alternativa a esse cenário, surge o *crowd testing*, visando à redução dos custos na realização dos testes, utilizando para isso o *crowdsourcing* [22]. Esse modelo está presente nas plataformas, nas quais testadores participam para oferecer benefícios como a força de trabalho com velocidade, processo mediado pela plataforma que realiza a triagem e o pagamento dos trabalhadores envolvidos no projeto [8].

O *crowd testing* utiliza uma comunidade de pessoas distribuídas em diversos locais para realização do teste de software de um determinado produto ou serviço. Esse software é apresentado utilizando cenários realistas como se fossem usuários do próprio sistema. Os testadores que encontram defeitos e cumprem os critérios de aceitação, são remunerados pelas plataformas (Ver seção 2.2.2) [23]. O *crowd testing* utiliza o conceito de *crowdsourcing*, pois os usuários realizam testes subjetivos de forma remota em seu ambiente preferido [22].

2.2.2 Plataformas

Diante desse modelo de distribuição com solução de problemas como o *crowdsourcing*, tornou-se necessária a criação de um elo entre o solicitante e a multidão. Ou seja, a criação de plataformas para auxílio no gerenciamento, não apenas de pequenas tarefas, de projetos inteiros, partindo desde sua documentação até sua entrega final. Dessa forma, possibilitou-se ao solicitante encontrar talentos além de seus limites físicos, tendo também outras vantagens como: redução nos custos, visibilidade de sua marca/empresa,

soluções diversificadas, melhor qualidade, acesso à criatividade, além é claro, da resolução do seu problema [24]. Para Morris [9], as plataformas de *crowdsourcing* são desejadas pelos empregadores também para questões de custo, embora o desempenho nas atividades realizadas possa variar.

Algumas plataformas buscam o aprimoramento ou oferecem serviços em uma determinada fase do processo, seja na coleta dos requisitos, no designer, no desenvolvimento ou na validação do escopo de uma demanda. Por exemplo, a UTest é uma plataforma de *crowd testing*, na qual os usuários participam para encontrar defeitos no sistema, sendo remunerados conforme gravidade do defeito reportado. Conforme Mao et al. [6] esses usuários não precisam, necessariamente, ter algum conhecimento em programação ou análise de requisitos. Os participantes necessitam seguir padrões para reportar os defeitos encontrados e, principalmente, saber o que faz parte do escopo. Outras plataformas, como por exemplo o Topcoder, atua em todas as fases do desenvolvimento de software.

As plataformas de teste de software via *crowdsourcing* podem oferecer flexibilidade e instantaneidade como diferenciais sobre os modelos empresariais tradicionais de validação de software, pois as conexões dos participantes ocorrem diariamente. Outro diferencial se refere aos participantes, pois as validações em teste de software não são realizadas apenas por profissionais, ou seja, a impressão sobre os programas e as aplicações serão próximas das do usuário ao qual o sistema é destinado.

As plataformas de *crowdsourcing* tomam por base as necessidades do solicitante e utilizam a inteligência coletiva para supri-las. São perceptíveis as semelhanças entre as plataformas, por exemplo em relação ao processo de realização dos testes de software, considerando que a maior parte dos testes são exploratórios ou na forma de relatar os defeitos, visto que os passos para criação desses são similares. Outro ponto importante é que elas utilizam o inglês como seu idioma padrão. Essas semelhanças entre as plataformas facilitam a compreensão do novato, servindo de estímulo para a participação em mais de uma plataforma.

A Laafttask foi desenvolvida por Altmann [25] em seu trabalho de conclusão de curso. A plataforma possui similaridades com as existentes, tratando dos pilares que estruturam o *crowdsourcing*. Para o desenvolvimento dessa ferramenta, foi utilizada a linguagem de programação PHP, tendo um módulo interpretado no servidor, gerando uma página web ao usuário. A plataforma teve seu servidor hospedado em ambiente disponibilizado pela Universidade de Passo Fundo (UPF). As tecnologias para a estrutura do *front-end* foram HTML5, *Javascript*, CSS e *Bootstrap*, possibilitando uma interface web para o usuário. Foi utilizado o PostgreSQL como sistema gerenciador de banco de dados (SGBD) pois apresenta licença *open source*. Para hospedagem do banco de dados, foi utilizado um serviço também disponibilizado pela UPF.

2.2.3 Barreiras para contribuição dos usuários

Conforme Brabham [3], estudos sugerem que os indivíduos estão mais propensos a contribuir quando verificam que seu conteúdo está sendo consumido. Outros fatores de motivação se referem à realização das atividades por *hobbies*, desenvolver suas habilidades, construir um portfólio para futuros empregos e o desafio na realização de um problema complexo. Outras pessoas utilizam o ganho financeiro como motivação para a participação.

As tarefas disponibilizadas nas plataformas de *crowdsourcing* muitas vezes são de alta complexidade, exigindo uma carga de trabalho custosa e trabalhadores focados na realização das atividades. Essa é uma barreira que exige do trabalhador um alto nível de habilidade, podendo limitar os trabalhadores em potencial na realização da tarefa [6]. Rohrmeier [26] relata que a participação dos usuários nas plataformas diminuiu, conforme esses se relacionam fora das plataformas.

Em uma avaliação com usuários do TopCoder, Zanatta et al. [4] verificaram que em uma amostra de 3403 usuários predominavam os que nunca haviam ganho uma competição de algoritmos disponibilizada. Esse foi um dos indicativos da presença de barreiras para vencer as competições disponíveis nas plataformas. A Figura 2 apresenta as barreiras identificadas em Zanatta et al. [4].

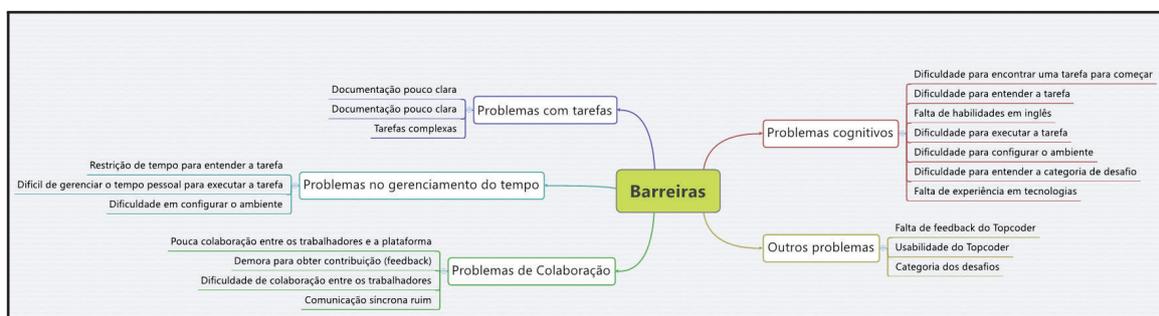


Figura 2. Barreiras na colaboração
Fonte: Adaptado de Zanatta et al. [4]

Observa-se que as barreiras foram agrupadas em cinco categorias: gestão de tempo, problemas com a complexidade das tarefas (tendo como alternativa a decomposição para redução da complexidade), barreiras referentes à colaboração, competências e demais problemas.

2.3 TÉCNICAS DE DECOMPOSIÇÃO DE TAREFAS

Durante o processo de design em um software, é necessária a representação para que os aspectos conceituais possam se tornar tangíveis. Nesse contexto, é preciso considerar as representações gráficas que podem ser um elo entre os membros do grupo, tanto

nas atividades de criação quanto na avaliação e implementação. As representações são consideradas importantes na solução de problemas, funcionando como artefatos cognitivos e fazendo a externalização do problema [27].

Na análise de tarefas, uma série de representações podem ser utilizadas. A Tabela 1 apresenta uma síntese das principais representações utilizadas em análise de tarefas.

Fase do método	Função de representação	Exemplo de representação
Descrição da tarefa	Sintetizar informações organizacionais e operacionais da tarefa	Ilustrações, tabelas e gráficos
Decomposição da tarefa	Fornecer uma visão da estrutura da tarefa com os objetivos, atividades e ações que a compõem	Diagramas hierárquicos, sequenciais e tabelas
Análise da tarefa	Estimar ou verificar relações entre variáveis específicas da tarefa	Redes de intensidade de fluxo, gráficos de demandas, cartas de-para, matrizes de erro
Síntese e aplicação dos dados da tarefa	Resumir os requisitos gerados pela análise da tarefa a serem aplicados no projeto	Listas textuais, tabelas, mapas conceituais

Tabela 1. Representações gráficas das fases da análise de tarefas

Fonte: Padovani [27]

Conforme detalhado na Tabela 1, a representação gráfica condizente com a fase de decomposição de tarefas pode ser a de diagramas hierárquicos, sequenciais e tabelas.

Para a decomposição sequencial, as tarefas podem ser desmembradas em subtarefas globais ou em atividades específicas, sendo posicionadas na ordem de ocorrência. Podem ser utilizados elementos gráficos informais ou notações rígidas, empregando codificações para diferenciar as atividades relatadas [27]. Em *Applied Cognitive Task Analysis* (ACTA), os autores propõem a decomposição em sub tarefas gerais, gerando um diagrama de seqüência com o objetivo de fornecer uma visão geral da tarefa. As subtarefas estão dispostas em formas elípticas conectadas por setas para indicar a direção [28]. A Figura 3 apresenta um exemplo de diagrama sequencial de envio de mensagens por e-mail.

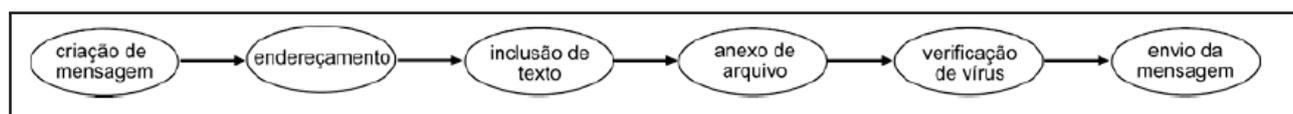


Figura 3. Exemplo de Diagrama Sequencial

Fonte: Padovani [27]

Conforme Padovani [27], a decomposição tabular representa as informações da análise hierárquica, porém em formato tabular, sendo que a notação possui o benefício de não restringir o espaço de expor os planos, operações e notas, conforme Tabela 2.

Operações e planos	Descrição continua?	Notas
0 Enviar e-mail Plano 0: 1 – 3 – 2 – 4. Se quiser enviar arquivo - 5 1. Criar nova mensagem 2. Verificar vírus 3. Endereçar mensagem 4. Enviar mensagem 5. Anexar arquivo	Sim Não Sim Não Sim	Informar limite
1 Criar nova mensagem Plano 1: 1.1 ou 1.2 1.1 Digitar texto na hora 1.2 Copiar texto de outro software	Não Não	Permitir formatação
3 Endereçar mensagem Plano 3: 3.1 ou 3.2 3.1 Digitar endereço 3.2 Buscar na lista de endereços	Não Não	Usar auto-completar Abrir em janela separada
5 Anexar arquivo Plano 5: 5.1 - 5.2 5.1 Buscar arquivo 5.2 Confirmar arquivo	Não Não	Integrar com ferramenta do sistema

Tabela 2. Análise hierárquica de tarefas em formato tabular

Fonte: Padovani [27]

Na primeira coluna da tabela, são incluídos os planos e as operações. A segunda coluna opera como um navegador, sendo que se a descrição for indicada positivamente, significa que a subtarefa precisa ser decomposta. A terceira coluna reserva um espaço livre para observações [27].

Na decomposição hierárquica, primeiramente identificam-se as metas das tarefas do usuário, realizando a decomposição dessas em subtarefas e atividades específicas.

Com isso, podem-se utilizar os planos para mostrar de que forma as atividades estão relacionadas. Uma das técnicas para decomposição hierárquica é a *Hierarchical Task Analysis* (HTA) que envolve a divisão de tarefas e subtarefas, sendo que, em seguida, é realizada uma nova divisão e, assim, sucessivamente. Em seguida, elas são agrupadas em planos que especificam de que forma as tarefas podem ser executadas. Dessa forma, a HTA procura caracterizar as práticas do trabalho, realizando a identificação, a categorização e a decomposição de tarefas, sendo realizados refinamentos ao modelo. A Figura 4 apresenta as etapas da Análise Hierárquica da Tarefa.

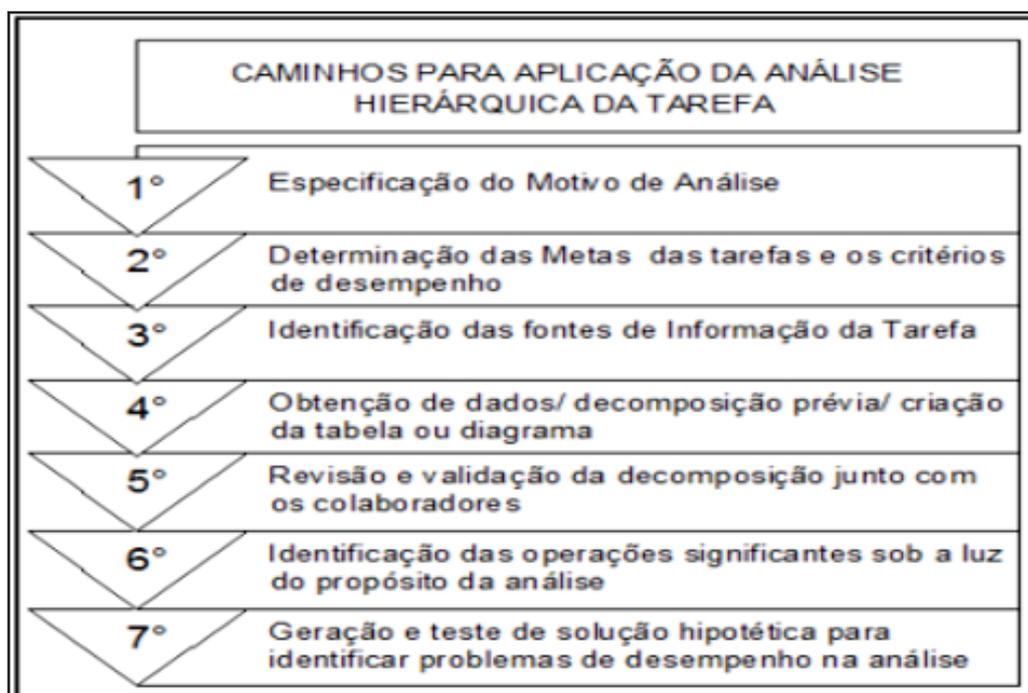


Figura 4. Etapas da Análise Hierárquica da Tarefa

Fonte: De Menezes [29]

A Figura 5 apresenta um exemplo da estrutura da decomposição hierárquica.

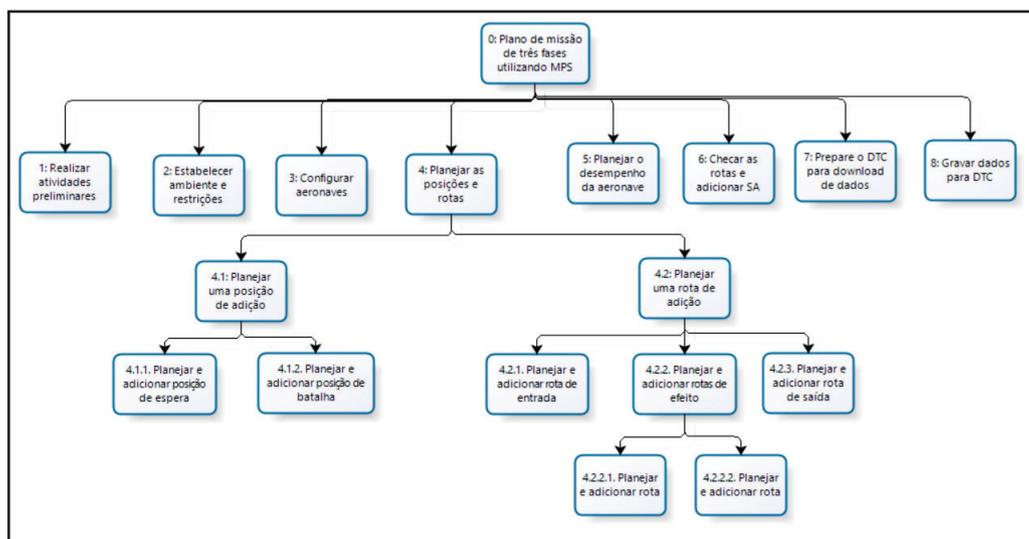


Figura 5. Exemplo de Decomposição Hierárquica

Fonte: Adaptado de Salmon [30]

O ponto de partida ao iniciar a decomposição é a identificação do objetivo do usuário, focando nas ações físicas e observáveis que serão executadas. Com isso, as tarefas principais a serem executadas são identificadas, sendo que, quando necessário, essas são subdivididas continuamente [31]. Após esse processo, é criada a estrutura da decomposição de tarefas hierárquicas.

2.4 MELHORIA DO PROCESSO DE TESTE BRASILEIRO

A qualidade do sistema ou produto é extremamente influenciada pela qualidade do processo utilizado. A busca por modelos é vinculada à demanda organizacional, sendo que a gestão dos ativos da organização é crítica para o sucesso do negócio. Desse modo, os modelos de maturidade têm por objetivo auxiliar as organizações a alcançarem os resultados desejados, realizando a melhor execução das atividades planejadas além de minimizar os impactos [32].

O modelo MPT trata da melhoria do processo por meio de práticas relativas a atividades desenvolvidas no decorrer do ciclo de vida do teste do software e tem como principais objetivos [32]:

- Tornar-se referência para definição, implantação e melhoria do processo de teste.
- Tratar da melhoria contínua dos processos de teste de acordo com os objetivos das organizações e nível de maturidade desejado.
- Reunir boas práticas e estruturá-las, conforme o grau de complexidade e o nível de maturidade que está relacionada.

A versão 2.0 do modelo foi aperfeiçoada tendo como referência as versões anteriores, a aplicação em projetos piloto e as contribuições de comunidades de teste de software.

2.4.1 Nível de Maturidade

O guia apresenta cinco níveis de maturidade representando patamares para evolução do processo de teste de software, conforme descritos a seguir:

1. *Parcialmente gerenciado*: é o primeiro nível do patamar, representando o mínimo que a organização precisa para demonstrar que a disciplina é aplicada nos projetos.
2. *Gerenciado*: no segundo nível, o processo de teste possui maior visibilidade, sendo controlado por processo de gestão de mudanças com padrões definidos e monitoramento dos processos.
3. *Definido*: nesse nível, o processo do teste se torna organizacional. Os processos são adotados, a garantia de qualidade é instituída, são definidas responsabilidades na organização do teste e programa de medição.
4. *Prevenção de defeitos*: o quarto nível foca na prevenção de defeitos e na melhoria da qualidade do produto. O processo de gestão é apresentado na organização, na qual os defeitos encontrados com antecedência são acompanhados e, ações são tomadas para evitar o surgimento de novos.
5. *Automação e Otimização*: o quinto nível tem por objetivo estabelecer o processo de melhoria contínua e automação de testes. Existe uma abordagem sistemática para automatizar a execução do teste sendo que o processo é controlado para realizar a melhoria contínua.

Cada nível de maturidade é composto de um conjunto de áreas e associado a um conjunto de práticas genéricas que devem ser aplicadas em cada uma das áreas do processo. Para que a organização atinja o nível de maturidade, ela deve demonstrar que o processo é aplicado em seus projetos.

2.4.2 Áreas do processo do modelo MPT

A organização das áreas do processo é definida, conforme apresentado na Tabela 3. Com os níveis detalhados, cabe à organização escolher um nível a ser almejado e optar pelas práticas e áreas do processo a ser atingido.

Nível de Maturidade	Áreas do processo
Nível 1	GPT - Gerência de Projetos de Teste (práticas GPT1 a GPT20) PET - Projeto de Execução de Teste (práticas PET1 a PET4)
Nível 2	GRT – Gerência de Requisitos de Teste (práticas GRT1 a GRT5) GPT – Gerência de Projetos de Teste (práticas GPT21 a GPT25) PET – Projeto e Execução de Teste (práticas PET5 e PET6)
Nível 3	FDT – Fechamento do Teste (práticas FDT1 a FDT4) GDQ – Garantia da Qualidade (práticas GDQ1 a GDQ3) MAT – Medição e Análise de Teste (práticas MAT1 a MAT5) OGT – Organização do Teste (práticas OGT1 a OGT10) TDA – Teste de Aceitação (práticas TDA1 a TDA7) TES – Teste Estático (práticas TES1 a TES5) TRE – Treinamento (práticas TRE1 a TRE4) GPT – Gerência de Projetos de Teste (práticas GPT26 a GPT28) PET – Projeto e Execução de Teste (prática PET7)
Nível 4	AQP – Avaliação da Qualidade do Produto (práticas AQP1 a AQP5) GDD – Gestão de Defeitos (práticas GDD1 a GDD3) TNF – Teste Não-Funcional (práticas TNF1 a TNF3) OGT – Organização do Teste (práticas OGT11 e OGT12)
Nível 5	AET – Automação da Execução do Teste (práticas AET1 a AET6) CEP – Controle Estatístico do Processo (práticas CEP1 a CEP5) GDF – Gestão de Ferramentas (práticas GDF1 a GDF6)

Tabela 3. Níveis da MPT
Fonte: Adaptado de MTP [9]

Para o Projeto e Execução de Teste (PET), os objetivos são identificar, elaborar e executar os casos de teste, realizando o registro da execução do teste e as divergências entre os resultados atuais e os esperados. Essa área do processo envolve:

- *PET1 – Identificar casos de teste:* o objetivo da prática é a identificação, a priorização e a documentação dos cenários de teste do sistema. Os cenários podem ser estruturados em alto ou baixo nível, sendo que em alto nível não são apresentados valores específicos para a entrada de dados. Essa atividade produz o caso de teste como produto típico.
- *PET2 – Executar casos de teste:* o objetivo é a execução dos casos de teste identificados e o registro das informações da execução. A atividade gera o registro da execução de teste.

- *PET3 – Reportar incidentes:* seu objetivo é garantir que as divergências de comportamento sejam reportadas na forma de incidente. A atividade produz o registro de incidente como produto típico.
- *PET4 – Acompanhar incidentes:* o objetivo dessa prática é realizar a análise e o acompanhamento de todos os incidentes até seu fechamento. Essa atividade produz o relatório de acompanhamento de incidentes como produto típico.

3. TRABALHOS RELACIONADOS

Neste capítulo serão apresentados os trabalhos relacionados, sendo discutidos pontos que se assemelham e se diferenciam do trabalho proposto.

Em seu estudo, Latoza et al. [7] implementam uma abordagem que realiza a decomposição de tarefas. No trabalho é realizado um rastreamento das alterações de um gráfico de artefatos, gerando as micro tarefas de maneira apropriada. Essa abordagem é implementada no IDE *Crowdcode*, o qual apresenta um paradigma (*map-reduce*) que divide o problema em vários pequenos problemas menores (subproblemas). Após a resolução desses subproblemas (*map*), eles são unidos (*reduce*). Para a avaliação, foi realizado um experimento. O resultado do estudo, verificou que os participantes estariam mais propensos a contribuir com esses projetos utilizando o *Crowdcode*. O trabalho se relaciona ao proposto na tentativa de diminuir barreiras para a contribuição dos trabalhadores. Para isso, foi proposto uma solução para realizar a decomposição das tarefas.

Kittur et al. [33] apresentam estrutura de propósito geral para realização de tarefas complexas utilizando a abordagem de micro tarefas. Para realização da decomposição das tarefas, foi utilizado o *CrowdForge* com uma estrutura de *MapReduce* na realização de processamento distribuído. Isso permite que as tarefas complexas sejam divididas em tarefas menores. No estudo foi gerado um protótipo e estudos de caso para comprovação da abordagem proposta. Como conclusão dessa investigação, os estudos de caso mostraram que as tarefas produzidas pelo *CrowdForge* eram mais bem classificadas que as produzidas individualmente. A relação do trabalho ao que foi proposto expressa-se na tentativa de diminuir as barreiras para a contribuição dos trabalhadores, propondo como solução decomposição das tarefas.

Shu et al. [34] apresentaram uma abordagem de decomposição de tarefas e sua especificação funcional. Foi utilizado um meta modelo que adota a linguagem formal orientada por objetos e com isso foi possível auxiliar na precisão do modelo. A abordagem possibilita a verificação da inconsistência da decomposição e mostra se a tarefa decomposta é consistente com a original. Como resultado foi descrito um meta modelo e, para verificação, foi realizado estudo de caso, comprovando a eficiência da abordagem. A relação do trabalho ao que foi proposto expressa-se na tentativa de diminuir as barreiras para a contribuição dos trabalhadores, propondo como solução a decomposição das tarefas.

O estudo realizado por Jiang e Shigeo [35] apresenta uma análise entre dois métodos de decomposição de tarefas, apresentando a decomposição de tarefas verticais e horizontais. O objetivo desse estudo esteve focado na avaliação da eficiência na comparação entre os dois métodos. O resultado apresentado foi que a decomposição de tarefas vertical demonstrou superioridade em relação à horizontal no que tange à melhoria da quali-

dade da solução final. A relação ao trabalho está na proposta de uma solução para realizar a decomposição das tarefas e, com isso, buscar a diminuição das barreiras na contribuição dos usuários.

O trabalho de Dubey et al. [36] identificou barreiras impeditivas para a realização de contribuição pela multidão. Após a identificação dessas, foi sugerido um conjunto de recomendações para que os usuários superem-nas. Os autores apresentaram como limitação do trabalho o fato de que a abordagem pode não ter a mesma eficácia em outras plataformas, visto que o estudo foi aplicado apenas no TopCoder. Como conclusão relatam que os trabalhadores devem possuir competência e disponibilizar um grande esforço na gestão do tempo para que possam ter sucesso nas plataformas *crowdsourcing*. O trabalho está relacionado ao proposto na tentativa de diminuição nas barreiras na contribuição dos trabalhadores.

Os trabalhos relacionados que foram identificados são apresentados na Tabela 4. Na tabela são apresentadas diferenças entre o trabalho proposto e os relacionados.

Abordagem	Autor	Diferença ao trabalho proposto	Realização da decomposição
IDE Crowdcode na realização da decomposição.	Latoza et al. [7]	Decomposição realizada após da disponibilização das tarefas a multidão.	Pela plataforma.
Utilização do CrowdForge para realizar a decomposição das tarefas.	Kittur et al. [33]	Decomposição realizada após da disponibilização das tarefas a multidão.	Pela plataforma.
Adoção de meta modelo para realização da decomposição das tarefas.	Shu et al. [34]	Decomposição realizada após da disponibilização das tarefas a multidão.	Pela plataforma.
Comparativo entre métodos de decomposição de tarefas.	Jiang e Shigeo [35]	Adaptação de técnica de decomposição para auxiliar na realização da tarefa.	N/A
Identificação de barreiras na contribuição da multidão nas plataformas.	Dubey et al. [36]	Adaptação de técnica de decomposição para auxiliar na realização da tarefa e superar as barreiras.	N/A

Tabela 4. Comparação entre os trabalhos relacionados
Fonte: Autoria própria

Pode-se observar que a principal diferença entre a literatura e o trabalho proposto é a realização da decomposição das tarefas. Na literatura apresentada nos trabalhos relacionados, a decomposição é tratada como uma atividade praticada pela plataforma e o

que está sendo proposto destina-se à realização da decomposição pelo usuário antes da realização da tarefa.

4. MATERIAIS E MÉTODOS

Para realização da pesquisa, foi abordada uma forma híbrida, sendo realizadas as avaliações quantitativas e qualitativas. Dessa forma, o capítulo exhibe os métodos e procedimentos adotados na pesquisa, conforme apresentado na Figura 6.

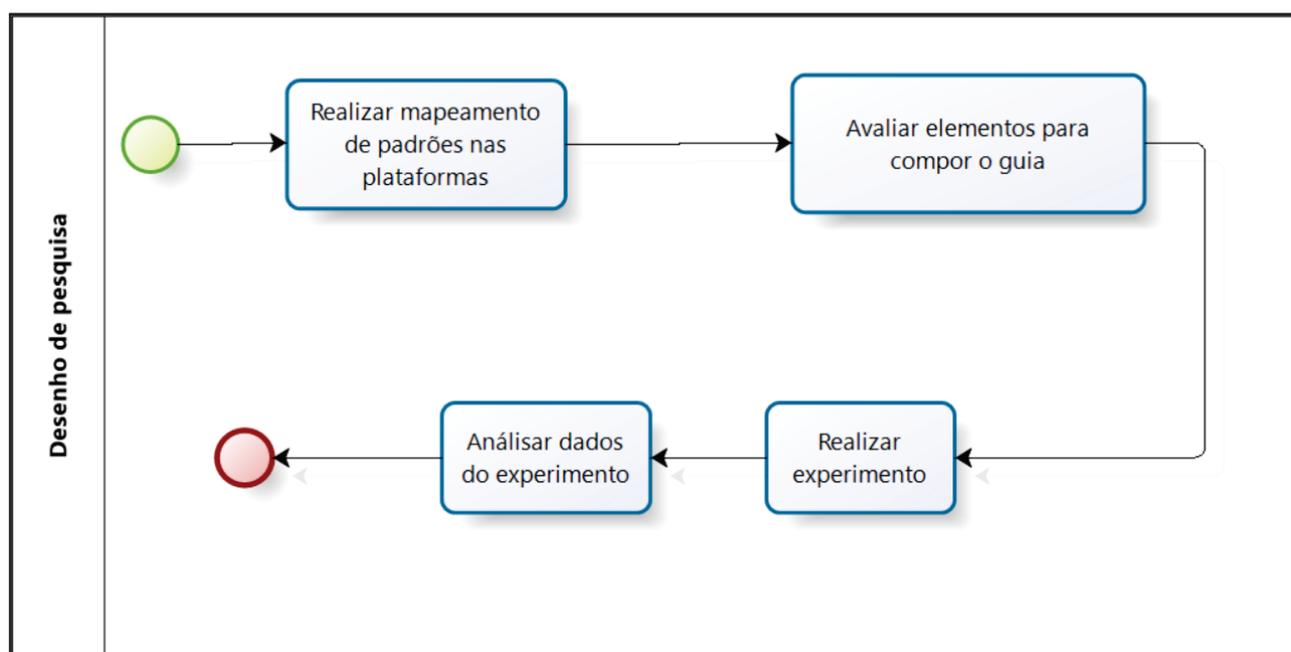


Figura 6. Fases do desenho de pesquisa
Fonte: Autoria própria

Primeiramente, as plataformas de *crowd testing* foram acessadas com o objetivo de mapear padrões na estrutura das tarefas. Para isso, foi realizada uma análise de conteúdo, visto na Seção 4.1. O primeiro passo foi importante para entender a estrutura das tarefas, servindo de base para a criação do guia. Na sequência, foram analisados elementos para compor o guia, sendo avaliadas técnicas de decomposição de tarefas e os padrões apresentados na Melhoria do Processo de Teste Brasileiro (MPT), assimilando as boas práticas para incorporá-las ao guia, vistos na Seção 4.2.

Baseado na técnica de decomposição hierárquica HTA, na Melhoria do Processo de Teste Brasileiro (MPT) e na estrutura das tarefas identificada pela análise de conteúdo (Seção 4.1), foi criado o guia³ para decomposição de tarefas, visto na Seção 5.2. A finalidade da criação de um guia foi propor um documento para auxiliar o trabalhador na realização da decomposição de tarefas em *crowdsourcing*, sendo validado por meio de um experimento pareado, tratado na Seção 4.3. Posteriormente, com dados obtidos no experi-

³Um guia é definido como um manual que contém informações, instruções e conselhos de diversas naturezas.

mento, foi realizada uma análise quantitativa (ver seção 4.4.1) e uma análise qualitativa (ver seção 4.4.2).

4.1 ANÁLISE DE CONTEÚDO

Nesta seção, serão apresentados os passos efetuados para a realização da análise de conteúdo das tarefas disponíveis nas plataformas de *crowd testing*, com a finalidade de mapear padrões na estrutura das tarefas. A metodologia e a organização da análise foram baseadas em Bardin [37] e sua aplicação realizada através do software Nvivo⁴.

As diferentes fases da análise de conteúdo organizam-se em três polos:

- Pré-análise
- Exploração do material
- Tratamento dos resultados e interpretação

A pré-análise é demonstrada na Seção 4.1.1, a exploração do material é tratada Seção 4.1.2 e o tratamento dos resultados são tratados na seção 4.1.3.

4.1.1 Pré-análise

Na fase de pré-análise, foi realizado o estudo em 7 plataformas de *crowdsourcing*: TopCoder, uTest, Testbirds, 99Test, Passbrains, Bugfinders e Crowdtest. A escolha das seis primeiras plataformas foi baseada em estudos de Ke Mao [6] e Zanatta et al. [38]. O TopCoder selecionado pois é, atualmente, a plataforma com o maior número de usuários inscritos e conforme Li et al. [39], uma plataforma que obteve êxito em *crowdsourcing*. As demais plataforma escolhidas por se tratarem de plataformas para realização de testes, sendo a Crowdtest uma plataforma de *crowd testing* brasileira. Após a realização do cadastro, iniciou-se um estudo sobre tarefas de cada plataforma, que buscou compreender a seleção de tarefas que possuíam abordagem em testes funcionais de software.

As atividades realizadas na fase de pré-análise estão listadas a seguir:

1. *Leitura flutuante*: a atividade inicial consistiu na leitura “flutuante” dos documentos para estabelecer o primeiro contato, conhecendo o texto para abstrair impressões e percepções. O contato com os documentos foi realizado nas plataformas já citadas, avaliando as tarefas apresentadas.

⁴O NVivo é um software para análise de dados produzido pela QSR International.

2. *Escolha dos documentos*: a segunda atividade realizada, foi a escolha das tarefas para análise, para escolha dos documentos suscetíveis, fornecendo informações sobre o problema levantado. Após a demarcação do universo, foi necessária a demarcação do *corpus*. O *corpus* é definido como o conjunto de documentos vistos para serem submetidos aos procedimentos analíticos e sua escolha implica em obedecer às regras.

A regra de exaustividade define que os campos do *corpus* não devem ser deixados de fora da análise por nenhuma razão. A regra de representatividade determina se a amostra tem parte representativa no universo homogêneo. Em cada uma das plataformas, o critério de escolha foi que as tarefas deviam ser relacionadas a Testes Funcionais. A regra de homogeneidade deve apresentar documentos que obedecem aos critérios de escolha, conforme definidos anteriormente. A regra de pertinência define que os documentos retidos devem ser adequados, de modo a corresponder ao objetivo da análise.

Em cada plataforma as tarefas foram escolhidas conforme filtros e disponibilidade. No TopCoder foram analisadas as tarefas disponíveis e acessadas por meio da realização do filtro por palavra-chave na plataforma, sendo que não possuía agrupamento específico para buscar tarefas de testes funcionais. Com isso, o filtro utilizado foi o de “QA”, retornando quatro, sendo que dessas apenas uma tarefa se referia a testes funcionais. No Utest, foi utilizada uma tarefa, a qual foi recebido o convite para realização de testes. Para ser possível a análise de outras tarefas, é necessário aguardar a disponibilização de novos projetos para realização dos testes. No Passbrains foi extraída para avaliação uma tarefa que se refere ao ciclo de teste inicial da plataforma. É necessário aguardar o recebimento de novos testes para se obter acesso a outras tarefas. No 99test estavam disponíveis duas tarefas para a avaliação. Essas foram extraídas para a ferramenta de análise. No Testbirds a tarefa que foi extraída é o teste de entrada da ferramenta. Após reportar essa atividade foi necessário aguardar o recebimento de novas tarefas. Na plataforma Crowdtest, foram utilizadas as tarefas disponibilizadas com data de encerramento posterior ao ano de 2018, totalizando 42 tarefas disponíveis na plataforma que foram utilizadas na análise. Para o Bugfinders, não foram recebidas tarefas para realização da análise.

3. *Formulação das hipóteses*: a próxima atividade desenvolvida se refere à formulação da hipótese. Trata-se de uma afirmação provisória que o pesquisador se propõe a verificar ou confirmar, recorrendo a procedimentos de análise. A hipótese levantada é a seguinte: as tarefas disponibilizadas nas plataformas de *crowdsourcing* possuem padrões em sua estrutura.
4. *Referenciação dos índices e elaboração dos indicadores*: os textos podem ser considerados como uma manifestação de índices, sendo que essa fase consiste na escolha

de índices em função da hipótese. Os indicadores levantados foram citações específicas dos temas analisados ou termos que remetem ao tema.

5. *Preparação do material*: antes da realização da análise, faz-se necessária a organização do material que será analisado. Foi realizado o *download* das tarefas verificadas, dividindo-as por plataforma.

4.1.2 Exploração do material

4.1.2.1 Codificação

Há a necessidade de saber porquê os dados são analisados e demonstrar a forma de análise. A codificação é definida pelo processo pelo qual os dados brutos são transformados e agregados em unidades que permitem uma descrição exata das características do conteúdo. A codificação corresponde a uma transformação, sendo efetuada segundo regras precisas, permitindo atingir uma representação de conteúdo.

A etapa de codificação é organizada compreendendo três escolhas:

- O recorte: escolha das unidades.
- A enumeração: escolha das regras de contagem.
- A classificação e a agregação: escolha das categorias.

A **unidade de registro** se refere à unidade de significação a ser codificada e corresponde ao segmento do conteúdo, considerada a unidade de base, pretendendo categorizar e realizar a contagem por frequência. A escolha das unidades de registro se deu em nível semântico, ou seja, o recorte realizado teve por base o tema. O tema é uma unidade de significação que se liberta naturalmente do texto analisado, podendo ser recortado em ideias, enunciados ou em proposições portadoras de significado.

A **unidade de contexto** serve como unidade de compreensão na codificação da unidade de registro, correspondendo ao segmento da mensagem, sendo que as dimensões são ótimas para que se possa compreender o significado exato da unidade de registro. A unidade de contexto a ser observada se refere às tarefas de testes funcionais das plataformas de *crowdsourcing*.

Para a **regra de enumeração** se faz necessária a distinção da unidade de registro, do que se conta e a regra de enumeração. Foi utilizada como modo de contagem a frequência, sendo previsto o aumento da importância da unidade de registro, conforme a frequência de sua aparição. Essa medida propõem que todos os elementos têm a mesma importância.

4.1.2.2 Categorização

A categorização é a operação de classificação de elementos que constituem um conjunto por diferenciação e, na sequência, por agrupamentos segundo o gênero, com critérios previamente definidos. As categorias são rubricas ou classes que reúnem um grupo de elementos, sendo que esse agrupamento é realizado em razão dos caracteres comuns desses elementos. O critério utilizado foi o tema, sendo de categorização semântica. O processo é dividido em duas etapas: o inventário que é utilizado para isolar os elementos e a classificação que reparte os elementos para procurar ou impor uma organização às mensagens.

4.1.3 Tratamento dos resultados e interpretação

4.1.3.1 Inferência

A análise de conteúdo fornece informações adicionais ao leitor crítico de uma mensagem, quando esse se distancia de sua leitura para saber mais sobre o texto.

A inferência se trata de obter conclusões sobre a população a partir de uma amostra que temos da população (efetuar generalizações a partir de amostras). Pode ser entendida como uma operação intelectual que afirma a autenticidade de uma proposição decorrente de ligações com outras proposições reconhecidas como verdadeiras.

Com base nas hipóteses levantadas anteriormente, deduz-se que a análise dos resultados das plataformas apresenta padrões e uma estrutura padrão existente entre as tarefas das plataformas. Considera-se essa informação devido ao propósito das plataformas no *crowdsourcing* e às tarefas propostas, tendo o objetivo de realizar a validação de um software.

4.2 AVALIAÇÃO DA TÉCNICA DE DECOMPOSIÇÃO E MELHORIA DO PROCESSO DE TESTES

As técnicas mapeadas na Seção 2.3, foram analisadas para eleger a que iria compor o guia de decomposição. No decorrer do trabalho, foi realizada a escolha pela modelagem hierárquica pois essa realiza um agrupamento de funções similares do sistema, mostrando as informações necessárias de cada grupo, facilitando a estruturação do sistema [27]. Dentre as técnicas de modelagem hierárquica, a técnica HTA foi utilizada, pois pode ser usada para descrever como as tarefas podem ser realizadas pelo usuário. Para utilização no guia, a HTA foi adaptada, sendo que os passos 5 e 7 não foram utilizados. O propósito da HTA é dividir as operações a serem realizadas hierarquicamente, relacionando-

se com o padrão de decomposição gerado, apontando os passos que o usuário deve seguir para realização da tarefa em questão na plataforma [30].

Além da HTA, foi analisado o processo da Melhoria do Processo de Teste (MPT), tendo por objetivo de abstrair parte das práticas e incorporá-las ao guia. Em virtude dos usuários possuírem experiências, optou-se por agregar esse processo para auxiliar esses usuários a terem os resultados desejados. Após a análise das práticas do MPT, foi optado por utilizar uma de suas práticas, o Projeto e Execução de teste (PET), visando a identificação, elaboração e execução dos cenários de teste do sistema.

Tendo por base o HTA e MPT, foi o guia para decomposição de tarefas, que foi avaliado através de um experimento pareado.

4.3 EXPERIMENTO

O experimento foi realizado com dois grandes objetivos: o primeiro avaliou se o guia auxiliou o usuário a submeter as tarefas de testes funcionais, considerando as tarefas para a decomposição. O segundo objetivo avaliou se o guia auxiliou na realização da decomposição da tarefa. Para validação do guia foi realizado um experimento pareado com 11 alunos da disciplina de Teste de Software do sétimo nível do curso de Análise e Desenvolvimento de Sistemas da Universidade de Passo Fundo (UPF). Para isso, foi utilizada uma sala do laboratório de informática da UPF. Os computadores da sala estavam disponíveis para os testes e, porém, os computadores pessoais na rede interna também poderiam ser utilizados, tendo sido essa a opção de alguns alunos. Antes de iniciar cada fase do experimento, foi disposto em torno de 25 minutos para explicação do seu funcionamento e assinatura do termo de livre consentimento.

O teste foi feito em duas etapas: na primeira, os alunos realizaram as tarefas disponibilizadas sem a utilização do guia; e, na segunda, eles utilizaram o guia de decomposição de tarefas. Para eliminar o conhecimento prévio, as tarefas da fase 1 foram diferentes da fase 2. Foram gerados os documentos de planejamento, servindo de base para a execução e análise, conforme descrito por Wohlin et al. [40].

4.3.1 Organização do experimento

Na organização do experimento, a primeira fase é a **seleção do contexto** que tem por objetivo obter elementos gerais para o experimento. Nessa fase, foi realizado um projeto com um grupo de 11 estudantes, utilizando tarefas reais criadas pelo autor do trabalho, com o objetivo de realizar testes no site de Eventos da UPF⁵.

⁵<https://secure.upf.br/eventos/eventos>

Posteriormente, foi definida a hipótese, delineada como uma teoria ou suposição que pode vir a explicar um determinado comportamento útil à pesquisa. O estudo experimental tem por objetivo a coleta de dados em ambientes controlados, confirmando ou negando a hipótese.

As hipóteses levantadas foram:

- H0: A decomposição das tarefas não aumenta o número de submissões das tarefas.
- H1: A decomposição das tarefas aumenta o número de submissões das tarefas.
- H0: O guia proposto não auxilia na realização da decomposição da tarefa pelo usuário.
- H1: O guia proposto auxilia na realização da decomposição da tarefa pelo usuário.

A hipótese nula (H0) indica que não há padrões no experimento e são coincidências, sendo que o analista deve buscar sua rejeição. A hipótese alternativa (H1) é a inversa à hipótese nula, sendo que será aceita caso a H0 seja rejeitada.

Na seleção das variáveis do estudo, foram definidas as dependentes e independentes. As variáveis independentes são as que podem ser controladas e alteradas no experimento. O efeito do tratamento é medido por meio das variáveis dependentes. Para o experimento, foi definida a variável independente como a complexidade das tarefas, afetando a desistência dos usuários de *crowdsourcing*, vista como variável dependente.

4.3.1.1 Seleção das amostras

No âmbito dos cenários do software *crowdsourcing*, a população se dá pela totalidade de usuários cadastrados nas plataformas estudadas. Desse modo, foram acessadas as informações de cada uma das plataformas com o intuito de verificar os usuários cadastrados. Em cada uma das plataformas foram identificados:

- Topcoder: 1.479.596⁶
- Crowdtest: 13.762⁷
- UTest: 693.734⁸
- Testbirds: 400.000⁹

⁶<https://www.topcoder.com/community/statistics/>

⁷<https://www.crowdtest.me/>

⁸<https://platform.utest.com/account/my-profile/statistics>

⁹<https://www.testbirds.com/company/about-us/company-history/>

- 99Test: 48.524¹⁰
- Passbrains: 45.000¹¹
- Bugfinders: 50.000¹²

Considerando as plataformas, a população é de 2.730.616 milhões de usuários cadastrados. Buscou-se, então, o número de usuários ativos, porém, em virtude da restrição de informações das plataformas, não se tem conhecimento da quantidade para obter a amostra.

A pesquisa de Mao et al. [41] utilizou dados históricos de tarefas da plataforma Topcoder e demonstrou que apenas 13,3% dos usuários já venceram uma tarefa. Outros 21,8% nunca venceram uma tarefa, porém permaneceram ativos na participação por 180 dias. Considerando apenas uma estimativa de usuários ativos, a amostra torna-se um número inviável para a pesquisa. Dessa forma, utilizou-se a amostragem não probabilística por conveniência, utilizando pessoas próximas para realização do experimento.

Os participantes foram representados por 11 alunos de ambos os sexos, graduandos da disciplina de Teste de Software do curso de Análise e Desenvolvimento de Sistemas da Universidade de Passo Fundo (UPF), sendo que todos desenvolvem uma atividade profissional relacionada a software. As atividades foram desempenhadas por alunos distribuídos por experiência da seguinte forma: um aluno (9.1%) desempenha funções na área de software há menos de um ano; outro (9.1%) ficou na faixa entre um ano e cinco anos; e nove alunos (81.8%) não possuem experiência na área de teste de software. Os participantes possuíam idades variadas, sendo que 90.9% de 20 a 25 anos e 9.1% de 26 a 30 anos.

4.3.1.2 Design do experimento

No design do experimento foi utilizado um fator com dois tratamentos, no qual o fator é o guia para decomposição de tarefas e os tratamentos são: a utilização do guia e a não utilização do guia. Como o número e os sujeitos que iniciaram o primeiro tratamento são os mesmos que no segundo tratamento, há um balanceamento no número de sujeitos, conforme demonstrado na Tabela 5.

¹⁰<http://99tests.com/>

¹¹<https://www.passbrains.com/>

¹²<https://join.digivante.com/as-functional-tester>

Sujeitos	Tratamento 1 (Sem utilizar o guia)	Tratamento 2 (Utilizando o guia)
1	2	3
2	0	1
3	2	3
4	3	1

Tabela 5. Exemplo de tratamento pareado para resolução das tarefas

Fonte: Autoria própria

No exemplo apresentado na Tabela 5, é exibido no design do experimento, contendo a listagem dos sujeitos envolvidos. Na primeira coluna apresenta a listagem dos participantes, na segunda coluna as tarefas submetidas sem a utilização do guia e na terceira coluna as tarefas submetidas utilizando o guia.

4.3.1.3 Instrumentação

Na fase de instrumentação foram selecionados os mecanismos de coleta de dados. Os documentos utilizados na fase foram:

- **Documento de consentimento:** esse documento (ver Apêndice C) descreve as condições para a realização do experimento, declarando que concorda com a participação no experimento, sendo de própria vontade, não tendo qualquer ônus e com a finalidade exclusiva de colaboração com a pesquisa. Todos os participantes assinaram o termo antes do início do experimento.
- **Caracterização dos participantes:** foi acessada antecipadamente, a lista de todos os alunos que participaram do experimento por meio da lista da disciplina.
- **Material de treinamento:** foi disponibilizado aos participantes documento com instruções para relatar possíveis *bugs* encontrados durante os testes, tomando por base o formulário de defeitos.
- **Formulário de defeito:** também foi disponibilizado modelo para utilização na abertura dos *bugs*.
- **Questionário:** o questionário foi elaborado e disponibilizado pelo Google Forms e foi elaborado conforme seção 4.4.2 (ver Apêndice A e Apêndice B).

4.3.2 Metodologia e preparação para o experimento

O experimento foi realizado para verificar se o guia para decomposição das tarefas auxiliou os usuários na submissão das tarefas. Para registro do processo de experimenta-

ção, foi elaborado um protocolo (ver Apêndice D) com os procedimentos relatados nesta seção.

Inicialmente, foi necessário definir onde os testes seriam realizados. Depois de avaliar possibilidades, definiu-se que o site de Eventos¹³ da UPF seria utilizado para os testes da Fase 1. Para a Fase 2, definiu-se a utilização da área administrativa¹⁴ do site de eventos. Para isso, se fez necessário o contato com os responsáveis para disponibilização de um ambiente de homologação.

Com a definição do site, iniciou-se a definição das tarefas que seriam utilizadas no experimento. Para a Fase 1 foram disponibilizadas sete tarefas (ver Apêndice E), contemplando a página principal do site, página de login, cadastro de usuários e funcionalidades acessadas após login (meus dados, meus pagamentos, inscrição para evento). Na Fase 2, foram disponibilizadas cinco tarefas (ver Apêndice F) da área administrativa, contemplando funcionalidades de cadastro de eventos, cadastro de ambientes para eventos, etc.

No dia da realização da primeira fase, os alunos foram visitados pelo autor do trabalho no local do experimento para explicação dos procedimentos e funcionamento, sendo que o mesmo ocorreu em um laboratório de informática da UPF. Inicialmente foi contextualizado o poder da multidão, a proposta de trabalho praticada no *crowdsourcing* e como é o envolvimento das plataformas nesse cenário, enfatizando que as pessoas trabalham em prol de um objetivo em comum. Após a visão geral do experimento, foram apresentados seus objetivos, informando que esse seria dividido em duas fases, sendo que no dia seria iniciada a primeira fase.

A realização dos testes ocorreu com tarefas pré definidas com o objetivo de testar o site de eventos da Universidade de Passo Fundo. Para acesso dos participantes às tarefas, essas foram disponibilizadas dentro da plataforma Laaftask. Optou-se pela sua utilização ao invés de uma plataforma já existente para que com base nessa fosse possível obter o controle necessário para realização do trabalho, tanto na disponibilização das tarefas quanto na avaliação dos resultados.

Além das tarefas, para simular o ambiente *crowdsourcing*, foi necessário disponibilizar um fórum para debate entre os participantes em interação com a plataforma. Como a Laaftask está em desenvolvimento beta, não foi implementada essa funcionalidade. Com isso, foi utilizado o Moodle¹⁵ da UPF para utilizar o fórum durante o experimento. O Moodle é um software livre de apoio à aprendizagem, desenvolvido por uma comunidade colaborativa de usuários distribuídos pelo mundo todo, podendo ser executados em diversas plataformas como Linux, Windows e MAC OS. Também foi utilizado o Moodle para disponibilizar os documentos e manuais utilizados no decorrer do experimento, além dos dados para acesso à plataforma, link dos questionários, guia para decomposição das tarefas e

¹³<http://eventos-test.upf.br>

¹⁴<http://eventosadm-test.upf.br/eventosadm-test>

¹⁵<https://ead.upf.br/>

uma área para anexo de todos os documentos gerados nos testes. Os acessos às ferramentas puderam ser realizados tanto pelo computador pessoal dos alunos quanto pelos disponibilizados no laboratório, contudo o acesso à plataforma e ao site de Eventos estava restrito ao ambiente da UPF.

Para iniciar o experimento, os participantes foram orientados a realizar a assinatura do termo de livre consentimento, que pode ser observado no Apêndice C. Em seguida o experimento se iniciou e o autor do trabalho se dirigiu a um dos locais disponíveis no laboratório. Durante a realização desse, houve algumas intervenções pontuais para sanar dúvidas, contudo, a orientação foi para que todas as dúvidas fossem postadas no fórum do Moodle. A primeira fase do experimento se estendeu durante a noite em sala de aula e nos próximos seis dias, sendo que as tarefas ficaram disponíveis na plataforma.

Na semana seguinte, iniciou-se a segunda fase, mantendo os procedimentos e a contextualização do experimento. Nessa fase, a realização dos testes ocorreu na área administrativa do site de Eventos da universidade, com suas tarefas sendo apresentadas na plataforma Laaftask. A principal diferença para a segunda fase é a utilização do guia para decomposição de tarefa e os alunos foram orientados para utilização dessa durante o processo de resolução das tarefas. A segunda fase do experimento também se estendeu no decorrer da noite em sala de aula e nos próximos seis dias, tendo as tarefas disponíveis na plataforma para realização. O autor do trabalho ficou disponível para sanar qualquer dúvida postada no Moodle no decorrer das duas semanas do experimento.

Ao final de cada uma das fases do experimento os participantes do experimento foram orientados para responderem um questionário (ver Apêndice A e Apêndice B), construído conforme Seção 4.4.2.

4.4 ANÁLISE DO EXPERIMENTO

Nesta seção, serão apresentados os métodos de análise do experimento realizado.

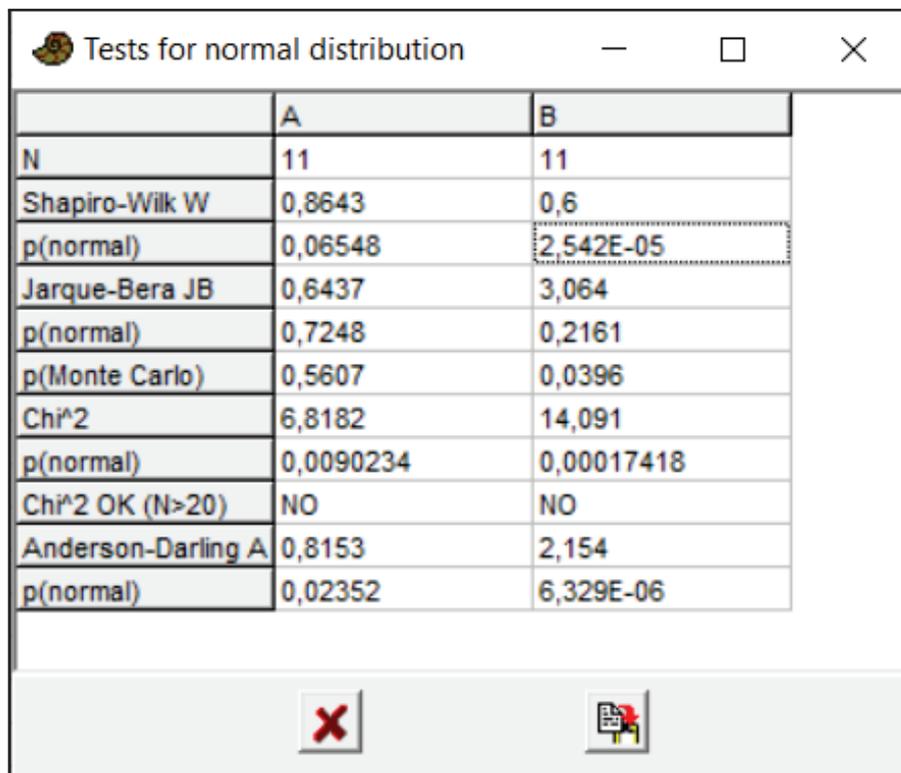
4.4.1 Análise quantitativa

Na pesquisa quantitativa, os resultados podem ser quantificados, sendo centrada na objetividade e recorrendo à matemática para descrever as causas dos fenômenos. Tem suas raízes no pensamento positivista lógico, enfatizando o raciocínio dedutivo, as regras lógicas e os atributos da experiência humana que são mensuráveis [42].

Para iniciar a análise, primeiramente foi avaliada a distribuição das variáveis do experimento, para o que foi utilizado o software estatístico Past¹⁶. O Past é um software livre utilizado na análise de dados estatísticos, com funções para manipular dados, plotagem,

¹⁶<https://folk.uio.no/ohammer/past/>

análise ecológica, etc. Os dados coletados do experimento foram inseridos no software e submetidos ao teste de normalidade, utilizando o teste Shapiro-Wilk, sendo que o resultado calculado para P foram superiores a 0.05, significando que os dados são normais [43]. É demonstrado, na Figura 7, o resultado do teste para normalidade dos dados.



	A	B
N	11	11
Shapiro-Wilk W	0,8643	0,6
p(normal)	0,06548	2,542E-05
Jarque-Bera JB	0,6437	3,064
p(normal)	0,7248	0,2161
p(Monte Carlo)	0,5607	0,0396
Chi ²	6,8182	14,091
p(normal)	0,0090234	0,00017418
Chi ² OK (N>20)	NO	NO
Anderson-Darling A	0,8153	2,154
p(normal)	0,02352	6,329E-06

Figura 7. Teste para distribuição normal

Fonte: Autoria própria

A distribuição normal caracteriza-se por uma função em que o gráfico descreve uma forma de sino. Essa distribuição depende da média (μ) e do desvio padrão (σ), sendo que sua curva é perfeitamente simétrica em torno da média [44].

O próximo item analisado objetivou esclarecer qual técnica a ser utilizada na análise dos dados. Conforme validado, o experimento gera dados pareados pois cada usuário da primeira fase do experimento é seu próprio par na segunda fase. Assim, a técnica utilizada para análise dos dados foi o teste t para dados pareados, sendo esse o apropriado para comparar dois conjuntos de dados quantitativos [44]. O teste t avalia os dados antes e depois de receberem um tratamento específico, sendo que o trabalho em questão se trata do guia para decomposição de tarefas.

4.4.2 Análise Qualitativa

A pesquisa qualitativa apresenta possibilidades de se estudar fenômenos envolvendo os seres humanos e suas relações sociais. Como um fenômeno pode ser melhor

compreendido no local onde ele ocorre, o pesquisador busca captar o fenômeno a partir da perspectiva das pessoas envolvidas. No decorrer da investigação o estudo qualitativo pode ser conduzido por caminhos diferentes [45].

Para conduzir a pesquisa qualitativa neste estudo, foi criado um questionário (ver Apêndice A e Apêndice B) para cada fase do experimento, sendo respondido ao final, contendo questões abertas e fechadas. Para o questionário da fase 1, um dos objetivos foi conhecer o perfil dos participantes e se esses conheciam ou participaram de projetos de *crowdsourcing*, além de informações profissionais. Outra finalidade foi para entender pontos da execução das tarefas como a complexidade e a dificuldade na realização. Para o questionário da fase 2, além da identificação dos participantes, foi avaliada a utilização do guia de decomposição, validando se esse auxiliou na realização das tarefas e na criação da decomposição.

Para identificação de possíveis erros antes da aplicação, foi realizado o pré-teste do questionário com cinco usuários. Para isso, foi encaminhado um e-mail a cada usuário, informando tratar-se de um questionário pré teste, solicitando que esse fosse avaliado conforme itens a seguir:

- Se a ordenação das perguntas está adequada.
- Se as perguntas possuem erros ortográficos.
- Se existem perguntas redundantes.
- Se existem perguntas constrangedoras.
- Itens em geral do questionário.

Os questionários da Fase 1 e da Fase 2 foram avaliados e respondidos com as considerações por meio do mesmo e-mail. Em linhas gerais, os itens identificados foram erros ortográficos, ordenação das perguntas e a forma de colocação das respostas das questões. As respostas foram avaliadas e realizados os ajustes pertinentes, gerando uma nova versão para cada um dos questionários.

5. RESULTADOS

As informações neste capítulo apresentam os resultados da criação e a aplicação do guia de decomposição de tarefas.

5.1 ANÁLISE DE CONTEÚDO

Para identificar os padrões na estrutura das tarefas de *crowd testing*, foi utilizando o software NVivo para organização e análise, as tarefas foram importadas no software. Foram criados agrupamentos, identificados como nós pela ferramenta. Ao final o conjunto de categorias mapeadas foi:

- Situação
- Deadline
- Instruções
- Requisitos
- Ajuda
- Pagamento

Essas categorias estavam presentes nas tarefas analisadas, demonstrando possuir padrões em sua estrutura. Com isso, verifica-se que a hipótese levantada na Seção 4.1.1 é verdadeira em relação às tarefas possuírem padrões em sua estrutura.

5.2 GUIA PARA DECOMPOSIÇÃO DE TAREFAS

O guia foi construído baseado na estrutura das tarefas identificadas na análise de conteúdo, na técnica de decomposição HTA e no MPT. Foi estruturado em quatro itens principais:

- Análise da estrutura da tarefa;
- Validação e identificação individual dos itens da estrutura
- Passos para realização da decomposição
- Execução e acompanhamento

5.2.1 Análise da estrutura da tarefa

O primeiro passo da análise a ser realizado é a avaliação da estrutura que compõem a tarefa a ser executada. O usuário deverá avaliar e entender quais itens fazem parte da estrutura da tarefa a ser executada. Conforme itens identificados na análise de conteúdo, a estrutura básica de uma tarefa é composta por: situação, deadline, instruções, requisitos, objetivos, ajuda e pagamento.

5.2.2 Validação e identificação individual dos itens da estrutura

Com a estrutura identificada, uma avaliação de cada um dos itens deverá ser realizada individualmente. Os quatro primeiros se tornam impeditivos para a sequência da execução da tarefa, conforme Tabela 6, a seguir.

Estrutura da tarefa	Detalhamento	Impeditivo?
Situação	Estado atual da tarefa	Sim
Deadline	Período de realização dos testes	Sim
Instruções	Orientações para os testes	Sim
Requisitos	Itens a serem completados na tarefa	Sim
Objetivos	Finalidade da tarefa	Não
Ajuda	Tópicos de auxílio ao testador	Não
Pagamento	Remuneração pela conclusão da tarefa	Não

Tabela 6. Detalhamento da estrutura da tarefa

Fonte: Autoria própria

A **Situação** corresponde ao estado atual da tarefa, podendo estar ativa ou inativa. Caso a tarefa esteja ativa significa que está disponível para realização dos testes e o testador pode iniciar a resolução. Caso esteja inativa, suas atividades foram interrompidas e impossibilitam a realização dos testes. Isso faz com que a situação da tarefa seja determinante na realização da tarefa pelo testador.

O **Deadline** identifica o período para realização dos testes, contendo data de início e fim para realizar a atividade. Esse item deve ser analisado cuidadosamente, pois o testador deverá avaliar se o tempo disponível é suficiente para realização da tarefa. Outro ponto importante é que quanto mais próxima a data final, mais *bugs* serão relatados na tarefa, o que diminui a possibilidade de identificar um problema que ainda não foi relatado.

As **Instruções** correspondem a orientações a serem seguidas na realização da tarefa. Podem ser apresentadas como pré-requisito para realização dos testes ou como orientação durante a realização desses. Uma instrução que não é seguida pode acarretar na reprovação do *bug* relatado ou na impossibilidade da realização de um teste.

Os **Requisitos** são um conjunto de itens a serem completados para que a tarefa seja iniciada e possa ser concluída. O não atendimento de algum item é impeditivo para iniciar ou finalizar corretamente uma tarefa.

Para auxiliar o testador, tópicos de **ajuda** são apresentados nas tarefas, os quais devem ser observados, pois contém guias e determinadas dúvidas frequentes entre outros testadores da plataforma. O **pagamento** é apresentado, indicando a remuneração que o testador receberá caso obtenha êxito na tarefa. Para orientar o testador, podem ser observados os tipos de *bug* mais bem remunerados na resolução da tarefa.

5.2.3 Passos para realização da decomposição

Neste item do guia são apresentados os passos para realização da decomposição de uma tarefa em software *crowdsourcing*. Como a estruturação do guia foi voltada à utilização em ambientes *crowdsourcing*, os passos 5 e 7 da técnica não foram utilizados. O HTA reconhece a responsabilidade do usuário em planejar o uso de recursos disponíveis para atingir o objetivo. Utilizam-se submetas e planos, sendo que o diagrama resultante apresenta uma hierarquia que está inter-relacionada, visto que seu foco está no sistema e suas propriedades [46]. A Figura 8 apresenta os passos necessários para realização da decomposição de uma tarefa.

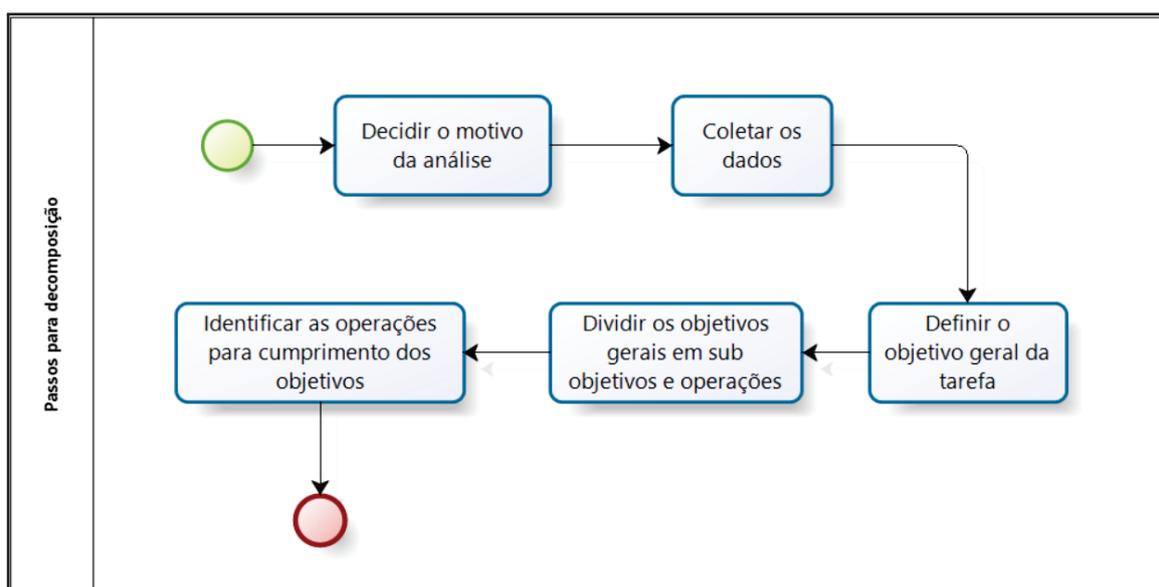


Figura 8. Passos para realização de decomposição de uma tarefa

Fonte: Autoria própria

No primeiro passo, a *decisão do motivo da análise*, é delimitado qual o objetivo da análise que será realizada, definindo a tarefa sob investigação e identificando o propósito desta. Esse ponto pode ser descrito como: "Realização dos testes do menu de simulação de contrato de renegociação da fatura de cartão de crédito e efetivação de pagamento da parcela do boleto".

Na *coleta de dados*, é necessária a obtenção das informações da forma como a tarefa será executada. Para isso, deve-se identificar onde o teste deverá ser realizado e quais os tipos de *bugs* aceitos. Por exemplo: para determinada tarefa, está descrito que os testes deverão ser realizados no mobile e não serão aceitos *bugs* referentes a erros de ortografia.

No próximo passo, deve-se *definir o objetivo geral da tarefa* e apresentá-lo no nível superior da árvore. Nesse passo também deverão ser identificadas as funcionalidades que deverão ser testadas, definindo o escopo dos testes e apresentando-os no próximo nível da hierarquia. Além do escopo, deverão ser definidos os itens que não fazem parte do escopo dos testes. Por exemplo: o objetivo da tarefa é a realização dos testes no menu de simulação de renegociação da fatura do cartão de crédito, tendo como escopo a utilização desse menu. Os itens definidos como fora de escopo são os que não são acessados por meio desse menu. Assim, o testador deverá ignorar qualquer *bug* encontrado nas funcionalidades não definidas no escopo do teste.

No passo *dividir os objetivos gerais em sub objetivos e operações*, devem ser mapeados os sub objetivos e as operações para cada objetivo e, dessa forma, realizar o mapeamento dos cenários de teste. O objetivo é realizar a identificação, a elaboração e a documentação dos cenários em alto nível. A estrutura da árvore seguirá sendo montada, tendo os subobjetivos um nível abaixo dos objetivos; e as operações, um nível abaixo dos sub objetivos. Como exemplo pode ser apresentado para o objetivo "Realização de testes na tela de login", um dos subobjetivos mapeados para testes poderia ser "Validar a impossibilidade de realização do login com dados inválidos"; ou "Validar a impossibilidade de login com dados não cadastrados". Quanto às operações para o sub objetivo "Validar a impossibilidade de realização do login com dados inválidos" poderia ser indicada como "informar os dados inválidos nos campos" e posteriormente "clicar no botão para realização do login".

Na *identificação das operações para cumprimento dos objetivos* deverão ser identificados os sistemas operacionais, as versões e os dispositivos necessários para a realização dos testes. Esse item deve ser observado, pois a realização dos testes em um dispositivo ou sistema incorreto acarreta em rejeição do *bug* aberto. Por exemplo: a determinação da plataforma é a realização do teste em computadores com sistema operacional Windows, no navegador Chrome e com sua versão superior à 35. Ao final da aplicação da técnica, os itens da tarefa ficarão estruturados em forma de árvore, conforme apresentado na Figura 9.

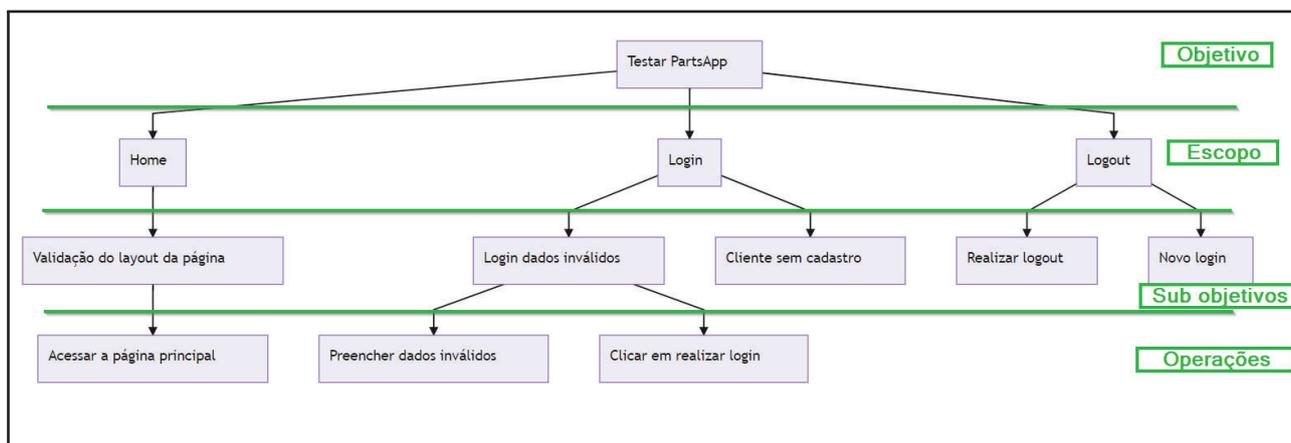


Figura 9. Estrutura após realizar decomposição

Fonte: Autoria própria

A Figura 9 apresenta um exemplo de estrutura após a realização da decomposição, tendo sido divididos em objetivos, escopo das tarefas, subobjetivos e operações.

5.2.4 Execução e acompanhamento

Neste item serão descritos os elementos para realização da execução e acompanhamento dos cenários mapeados. Os itens mapeados a seguir estão embasados na MPT, especificamente no Projeto e Execução de teste (PET) de 2 a 4, que consiste em PET2 – Executar casos de teste, PET3 – Reportar incidentes e PET4 – Acompanhar incidentes.

Os objetivos do PET são a identificação, a elaboração e a execução dos testes, realizando o registro da execução e as divergências destes. Isso implica que os casos de teste sejam identificados e elaborados contemplando as divergências reportadas [47].

5.2.4.1 Execução dos cenários mapeados

Nesse item devem ser executados os casos de teste. Devem ser identificadas as inconsistências, baseando-se nos cenários mapeados anteriormente, observando as saídas apresentadas pelo sistema. O objetivo dessa atividade é executar e registrar os cenários de teste mapeados anteriormente, observando os resultados fornecidos pela aplicação com o intuito de não deixar passar falhas [47].

5.2.4.2 Reportar incidentes

O objetivo desse item é garantir que as divergências entre o resultado esperado e o resultado obtido sejam reportadas por meio de incidentes. Um incidente é um evento

significante que não foi planejado durante o teste, sendo que pode não ser um defeito¹⁷. Pode-se chegar à conclusão de que o sistema se comportou indevidamente em virtude da configuração do ambiente de testes [47]. As situações identificadas devem ser reportadas, sendo que para cada inconsistência, deve ser criado um registro. Devem ser evidenciados os *bugs* encontrados, conforme instruções da plataforma. Caso não contenham instruções, devem ser considerados os itens básicos para reportar um incidente. A seguir, uma estrutura básica com as informações para reportar um *bug* [47]:

- Identificador do incidente
- Sumário
- Descrição do incidente
 - Entradas
 - Resultados esperados
 - Resultados Atuais
 - Anomalias
 - Data e hora
 - Passo do procedimento
 - Ambiente
 - Tentativas de repetição
 - Testador
 - Observadores
- Impacto

Há um ponto importante a ser observado antes do report do *bug*: o testador deve analisar os *bugs* já abertos e não reportar um problema já existente.

5.2.4.3 Acompanhamento de bugs

Neste item deve ser realizado o acompanhamento dos *bugs* enviados para verificar se esses serão aceitos ou rejeitados. Seu objetivo é garantir que os incidentes reportados sejam acompanhados até seu fechamento [47].

¹⁷É uma incoerência no software, algo que não foi implementado de maneira correta.

5.3 EXPERIMENTO - FASE 1

Na Fase 1 do experimento, realizada no dia 31/10/2019, os alunos executaram as tarefas pré-definidas sem a utilização do guia. Como a participação era opcional, um dos alunos optou por não participar. Dessa forma, foi dado início ao experimento, sendo que o autor se direcionou ao final da sala de aula para acompanhamento. A orientação foi para que as dúvidas fossem sanadas por meio do fórum do Moodle.

No decorrer do experimento foram pontuadas percepções pelo autor do trabalho. Esse acompanhamento foi realizado até o horário do intervalo da aula pois o restante da noite foi utilizado para atividades da disciplina. Seguem apontamentos:

- Os alunos conversavam entre si para tirar certas dúvidas em relação ao experimento, às tarefas e sobre se o comportamento verificado se tratava de um *bug*. Em contrapartida outros alunos utilizaram fones de ouvido para ficarem centrados na atividade.
- Mesmo com a explicação inicial, surgiu o questionamento da forma que o fórum deveria ser acessado.
- Após 10 minutos do início do experimento, foi verificado que alguns alunos iniciaram o preenchimento do documento para reportar os *bugs*.
- A funcionalidade observada com mais aderência para os testes foi a área de Meus Dados, que se trata da área com as informações do usuário do site.
- Observou-se que alguns alunos utilizaram o emulador de dispositivos mobile para realização dos testes ao invés do acesso com aparelho pessoal.
- Ocorreu indisponibilidade da internet durante a realização do experimento.
- Observou-se que a navegação dos participantes não se concentrava apenas em uma funcionalidade específica, mas em todas as funcionalidades.
- Após 30 minutos do início do experimento, o primeiro questionamento foi postado no fórum.

No fórum do Moodle foram postadas duas dúvidas. A primeira se refere ao procedimento que deveria ser realizado para relatar os *bugs* identificados. A orientação foi passada para utilizar imagens da tela e direcionar de forma textual para auxiliar na compreensão do *bug*. O segundo questionamento realizado no fórum foi referente à plataforma Laafttask, questionando a exibição das tarefas concluídas na área de minhas tarefas. Nesse caso, foi informado que a área não mantinha as tarefas dadas como concluídas. As respostas das dúvidas foram respondidas também por meio do fórum.

5.3.1 Resolução das tarefas - Fase 1

Após a finalização do prazo, as tarefas não foram mais exibidas na plataforma. Com isso, os dados publicados na ferramenta foram extraídos e avaliados. A Tabela 7 apresenta os participantes e o número de tarefas submetidas.

Participante	Tarefas submetidas
Participante 1	2
Participante 2	1
Participante 3	2
Participante 4	1
Participante 5	1
Participante 6	0
Participante 7	1
Participante 8	3
Participante 9	1
Participante 10	1
Participante 11	0

Tabela 7. Número de tarefas submetidas na Rodada 1

Fonte: Autoria própria

Conforme apresentado, um participante submeteu três tarefas; dois participantes, duas tarefas; seis participantes, uma tarefa; e apenas dois participantes não submeteram nenhuma das tarefas.

5.4 EXPERIMENTO - FASE 2

No dia 07/11/2019 foi executada a Fase 2 do experimento, realizando as tarefas utilizando o guia de decomposição. O local do experimento foi o mesmo da Fase 1, sendo utilizados os computadores do laboratório de informática e os pessoais de cada participante.

Os moldes da Fase 1 foram mantidos, com as tarefas disponibilizadas na plataforma Laaftask, porém, referentes à área administrativa do site de Eventos da UPF. As orientações da primeira fase foram revisadas rapidamente, tendo foco na principal diferença que foi a utilização do guia de decomposição para realização das tarefas. Três alunos que não haviam participado da Fase 1 do experimento estiveram presentes, tendo sido orientados separadamente acerca dos procedimentos e solicitados a assinar o termo de livre consentimento. Os demais participantes utilizaram esse período para responderem o questionário elaborado para a Fase 1. Com isso, iniciou-se a Fase 2 do experimento.

Tal como na Fase 1, foram levantadas percepções e ocorrências no decorrer do experimento, conforme descritas a seguir:

- Após a explicação inicial dos procedimentos, não houve dúvidas dos participantes.
- Os participantes interagiam com menos frequência em relação a Fase 1 do experimento.
- Uma dúvida pontual foi levantada verbalmente por um dos participantes, sendo respondida. Para registro, a pergunta e resposta foram publicada no fórum.
- Ocorreram problemas de acesso ao fórum da rodada 2 no Moodle. Após 10 minutos o problema foi solucionado.

No fórum do Moodle da rodada 2, foram enviadas três dúvidas. A primeira pergunta era referente ao local em que deveriam ser publicados os documentos de *bugs*, na plataforma ou na área do Moodle. Como a plataforma possui apenas uma área para anexar arquivos e o armazenamento limitado, foram orientados para que o documento de *bugs* fosse publicado nas respectivas tarefas, porém, ao final, os arquivos deveriam ser reunidos e anexados na área do Moodle. A segunda pergunta se referiu a um comportamento durante os testes da aplicação, sendo que a orientação foi para que o comportamento fosse avaliado pelo participante e documentado, caso esse considerasse um *bug*. O terceiro questionamento foi referente à forma de verificação dos demais *bugs*. Foi informado que esse item não estava contemplado e poderiam ser reportados sem o acompanhamento dos *bugs* já abertos.

5.4.1 Resolução das tarefas - Fase 2

Conforme realizado na Fase 1, ao final do período dos testes, as tarefas ficaram indisponíveis na plataforma, iniciando-se, então, a avaliação dos resultados. A Tabela 8 apresenta o número de tarefas submetidas pelos participantes.

Participante	Tarefas submetidas
Participante 1	2
Participante 2	0
Participante 3	2
Participante 4	0
Participante 5	0
Participante 6	0
Participante 7	0
Participante 8	1
Participante 9	0
Participante 10	0
Participante 11	0

Tabela 8. Número de tarefas submetidas na Rodada 2

Fonte: A autoria própria

Diferentemente da Fase 1 na qual apenas dois não concluíram nenhuma tarefa, nessa fase foram oito participantes com nenhuma tarefa concluída, dois com duas tarefas e apenas um com uma tarefa. Em relação à decomposição de tarefas, dois dos participantes que concluíram tarefas também realizaram a entrega das decomposições criadas. A Figura 10 apresenta a estrutura criada pelo participante 3 na resolução da tarefa referente à funcionalidade de validação de arquivos.

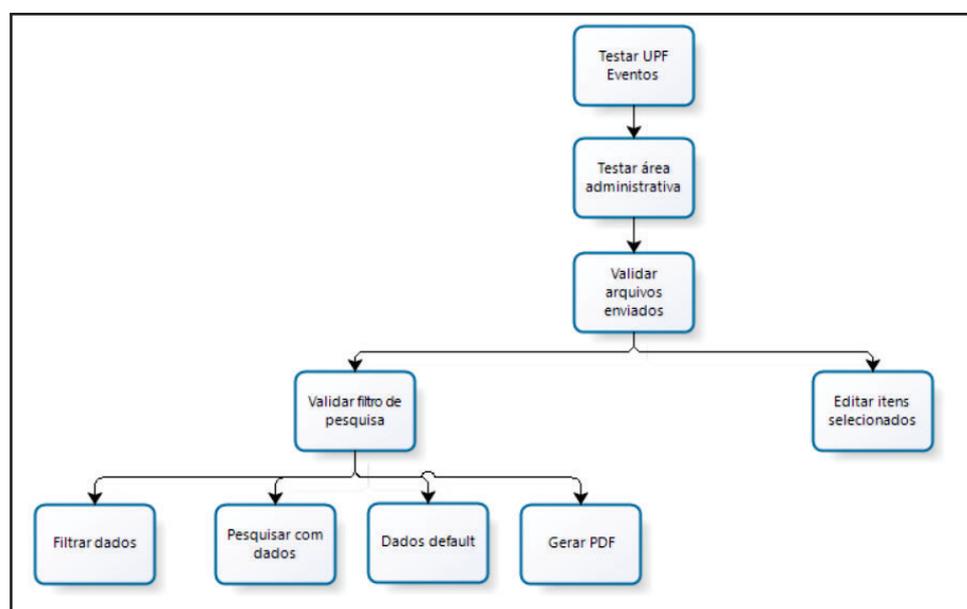


Figura 10. Decomposição tarefa validar arquivos

Fonte: Adaptado do entregue pelo participante 3

A Figura 11 apresenta a decomposição realizada para a tarefa de tipos de ambiente.

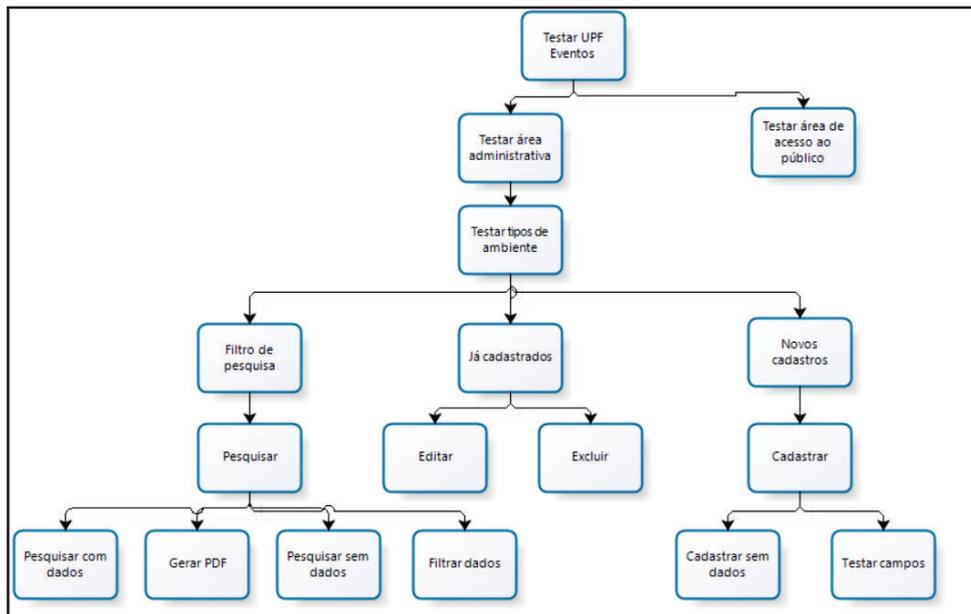


Figura 11. Decomposição tarefa tipos de ambiente

Fonte: Adaptado do entregue pelo participante 3

Em seguida a Figura 12 apresenta a decomposição realizada pelo participante 8.

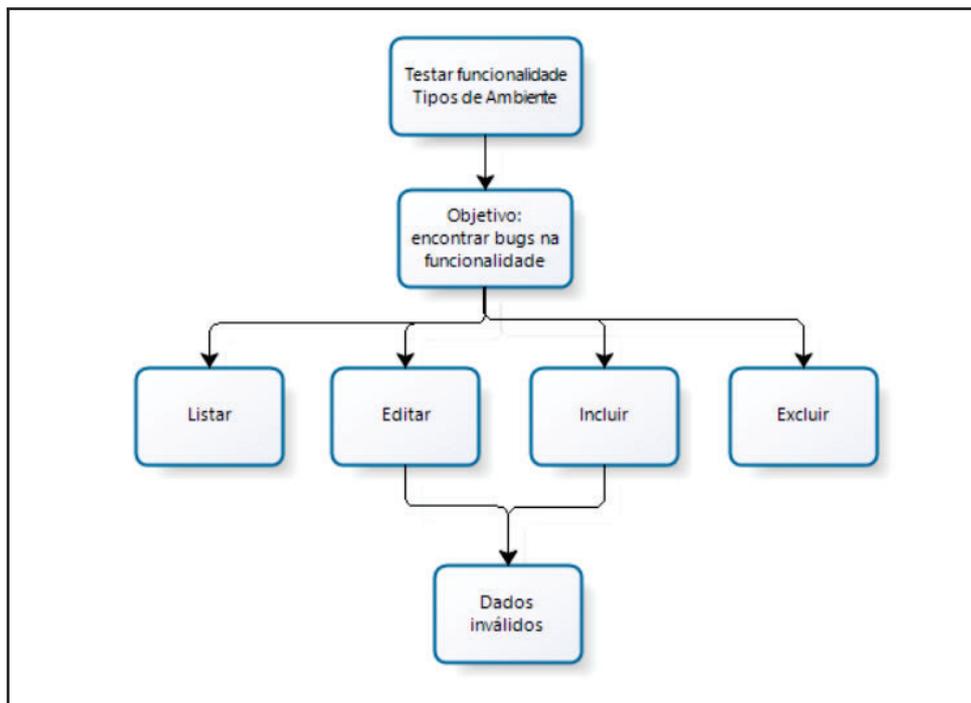


Figura 12. Decomposição tarefa tipos de ambiente

Fonte: Adaptado do entregue pelo participante 8

Observa-se que a decomposição entregue pelos dois participantes atende à solução proposta para a tarefa. Igualmente, a estrutura apresentada nas decomposições entregues respeita a estrutura proposta pela análise hierárquica de tarefas.

5.5 ANÁLISE DO EXPERIMENTO

O experimento coletou dados da realização das tarefas de quinze participantes. Foram desconsiderados da análise quatro participantes utilizando o critério de participação nas duas fases do experimento, sendo que um aluno participou apenas da Fase 1 e outros três apenas da Fase 2.

Nesta seção, serão apresentadas as análises quantitativa e qualitativa dos dados gerados pelo experimento.

5.5.1 Análise quantitativa do experimento

Para realização da análise quantitativa, foi utilizado o *teste t* para dados pareados, conforme relatado na Seção 4.4.1. Para iniciar a aplicação do método, a Tabela 9 apresenta as tarefas submetidas nas duas fases do experimento.

Participante	Fase 1	Fase 2	d	d^2
Participante 1	2	2	0	0
Participante 2	1	0	1	1
Participante 3	2	2	0	0
Participante 4	1	0	1	1
Participante 5	1	0	1	1
Participante 6	0	0	0	0
Participante 7	1	0	1	1
Participante 8	3	1	2	4
Participante 9	1	0	1	1
Participante 10	1	0	1	1
Participante 11	0	0	0	0
			$\Sigma = 8$	$\Sigma = 10$

Tabela 9. Tarefas submetidas fases 1 e 2

Fonte: Autoria própria

A coluna intitulada como d apresenta as diferenças, considerando as tarefas da Fase 1 subtraídas das tarefas da Fase 2 e a coluna do d^2 apresenta as diferenças elevadas ao quadrado. Ao final são apresentados os valores do somatório (Σ) de cada coluna.

Na sequência houve a definição das hipóteses, conforme descrito a seguir:

- H_0 : A utilização do guia não aumenta a submissão de tarefas.
- H_a : A utilização do guia aumenta a submissão de tarefas.

Ou,

$$H_0 : \mu_{fase\ 2} \leq \mu_{fase\ 1} \quad e \quad H_a : \mu_{fase\ 2} > \mu_{fase\ 1} \quad (1)$$

Onde,

μ_{fase1} : submissão de tarefas antes da utilização do guia.

μ_{fase2} : submissão de tarefas com a utilização do guia.

A partir da definição das hipóteses, define-se que na H_0 , $\mu_d \geq 0$ e na H_a , $\mu_d < 0$.

Para cálculo de t para dados pareados, é necessário realizar o cálculo, utilizando a forma definida por:

$$t = \frac{\bar{d} - \mu_d}{\frac{S_d}{\sqrt{n}}} \quad (2)$$

Sendo que,

\bar{d} : corresponde a média das diferenças observadas

n : ao tamanho da amostra

S_d : ao desvio padrão das diferenças

Para obtenção dos elementos necessários para sequência do teste, obtém-se o valor de \bar{d} por meio da fórmula:

$$\bar{d} = \frac{\sum d_i}{n} = \frac{8}{11} = 0,727 \quad (3)$$

Onde,

$\sum d_i$: corresponde ao somatório das diferenças (d), definido na Tabela 9.

n : corresponde ao tamanho da amostra, definida como $n = 11$ conforme demonstrado na Tabela 9.

Na sequência, obtém-se o valor de S_d .

$$S_d = \sqrt{\frac{\sum d_i^2 - \left[\frac{(\sum d_i)^2}{n}\right]}{n-1}} = \sqrt{\frac{10 - \left[\frac{(8)^2}{11}\right]}{11-1}} = \sqrt{\frac{10 - 5,818}{10}} = \sqrt{\frac{4,182}{10}} = \sqrt{0,4182} = 0,646 \quad (4)$$

Com os valores definidos, aplica-se a Fórmula 2 para obter o valor de t .

$$t = \frac{\bar{d} - \mu_d}{\frac{S_d}{\sqrt{n}}} = \frac{0,0727 - 0}{\frac{0,646}{\sqrt{11}}} = \frac{0,727}{\frac{0,646}{0,316}} = \frac{0,727}{0,194} = 3,747 \quad (5)$$

Após a observação dos dados amostrais e a realização do cálculo do valor de t , obteve-se a probabilidade de significância a partir da tabela da distribuição de t de Student. Para teste da H_0 versus H_a , observa-se uma amostra de 10 participantes, sendo calculado na Fórmula 5 o valor de $t = 3,747$ para 10 graus de liberdade ($n - 1$). Por meio da tabela do t de Student, demonstrada na Tabela 10, obteve-se o valor para o $t = 1,372$, sendo que a área corresponde ao nível de significância descrita pela amostra.

Graus de liberdade	Probabilidade para um teste bicaudal						
	0,30	0,20	0,10	0,05	0,02	0,01	0,001
-							
1	1,9626	3,0777	6,3138	12,7062	31,8205	63,657	636,619
2	1,3862	1,8856	2,9200	4,3027	6,9646	9,9248	31,5991
3	1,2498	1,6377	2,3534	3,1824	4,5407	5,8409	12,9240
4	1,1896	1,5332	2,1318	2,7764	3,7469	4,6041	8,6103
5	1,1558	1,4759	2,0150	2,5706	3,3649	4,0321	6,8688
6	1,1342	1,4398	1,9432	2,4469	3,1427	3,7074	5,9588
7	1,1192	1,4149	1,8946	2,3646	2,9980	3,4995	5,4079
8	1,1081	1,3968	1,8595	2,3060	2,8965	3,3554	5,0413
9	1,0997	1,3830	1,8331	2,2622	2,8214	3,2498	4,7809
10	1,0931	1,3722	1,8125	2,2281	2,7638	3,1693	4,5869
11	1,0877	1,3634	1,7959	2,2010	2,7181	3,1058	4,4370

Tabela 10. Recorte da tabela da distribuição t-Student
Fonte: Adaptado de [48]

Como o teste é unilateral, a coluna da tabela a ser considerada é o dobro do valor dos graus de liberdade, sendo neste caso de 20%. Com isso, o valor obtido é referente à linha de 10 graus de liberdade, e na coluna, de 20%.

Considerando um nível de significância de 10%, verifica-se na Figura 13 que o valor de t não faz parte da região crítica apresentada em destaque no gráfico.

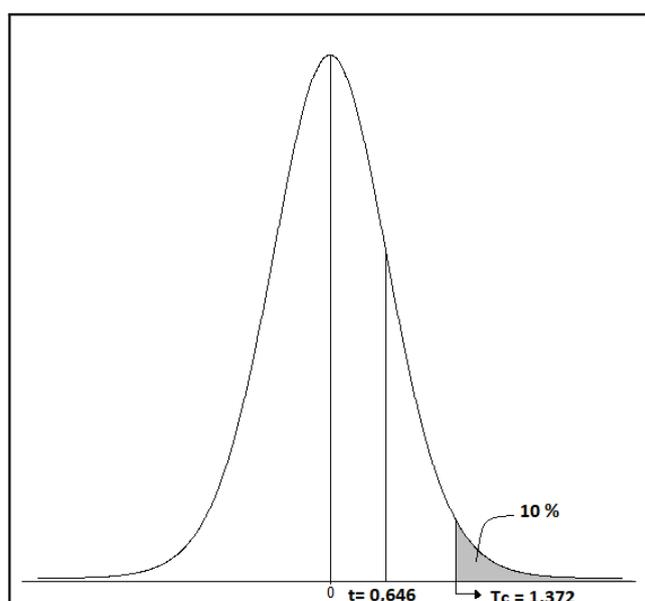


Figura 13. Distribuição t student
Fonte: Autoria própria

Com isso, a hipótese H_a é rejeitada e a H_0 é aceita, indicando que o número de submissões da fase 1 foi maior do que na fase 2. Observa-se, também, que as decomposições enviadas foram apenas duas, representando 18,18% dos participantes.

5.5.2 Análise qualitativa do experimento

Nesta seção é apresentada a análise qualitativa do experimento. Para obtenção dos dados foi elaborado um questionário para a Fase 1 do experimento e um para a Fase 2. Foram analisados os dados dos onze participantes, tanto para a Fase 1 quanto para a Fase 2.

5.5.3 Questionário Fase 1

No questionário da Fase 1, foram analisadas, inicialmente, as respostas das seguintes questões:

1. Qual é sua idade?
2. Além de estudante, você desenvolve alguma atividade profissional?
3. Quanto tempo de experiência em atividades relacionadas a teste de software você possui?
4. Você conhecia o desenvolvimento de software *crowdsourcing*?
5. Já participou em projetos de desenvolvimento de software via *crowdsourcing*?

Para a primeira questão, 90,0% dos participantes responderam que têm entre 20 anos e 25 anos de idade e 9,1%, entre 26 anos e 39 anos. Na segunda pergunta, 100% responderam que desenvolvem atividades profissionais. Na terceira pergunta, 81,8% responderam que não possuem experiência em teste de software, 9,1% menos de um ano e 9,1% entre 1 e 5 anos. A próxima pergunta indicou que 81,8% não conheciam desenvolvimento de software *crowdsourcing* e 18,2% tinham conhecimento. A próxima pergunta indicou que 100% dos participantes não haviam participado anteriormente de projetos em *crowdsourcing*.

Na sequência foram analisadas questões relacionadas às tarefas, conforme a seguir:

1. Sobre a realização das tarefas, possui dificuldade na execução.
2. Sobre as tarefas, considero a tarefa realizada complexa.
3. A forma de apresentação/descrição da tarefa foi suficiente para sua realização.
4. A documentação foi suficiente para o entendimento e realização da tarefa.

Foi utilizada a escala *likert* de 5 pontos, sendo que 1 representa discordo totalmente e 5 representa concordo totalmente. As respostas à primeira pergunta apresentaram

que a maior parte dos participantes demonstra-se neutra em relação à dificuldade na execução, conforme demonstrado na Figura 14.

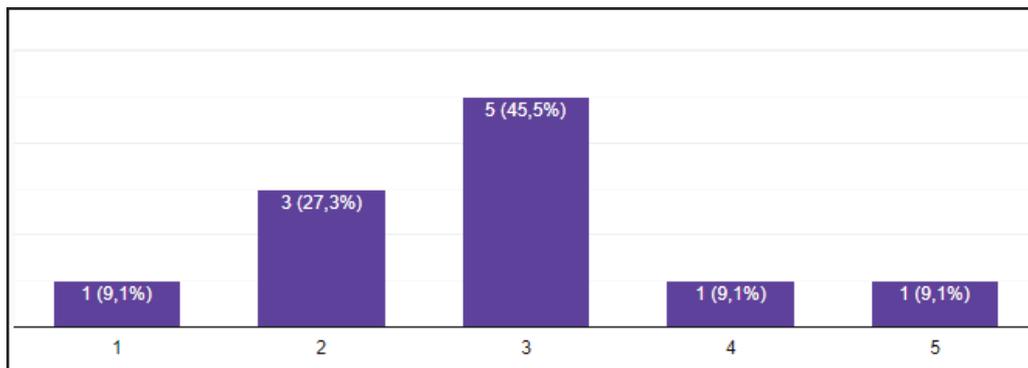


Figura 14. Respostas referente dificuldade nas execuções das tarefas
Fonte: Autoria própria

As respostas à segunda pergunta identificaram que, em linhas gerais, os participantes divergiram ao considerar as tarefas complexas, conforme apresentado na Figura 15.

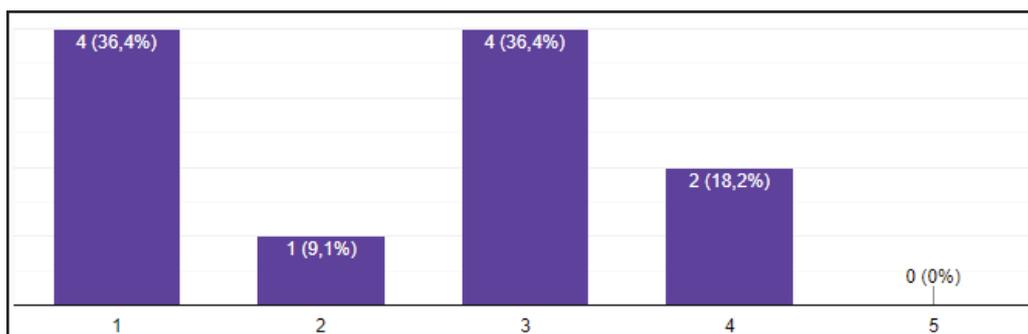


Figura 15. Respostas da consideração dos participantes a respeito da complexidade das tarefas
Fonte: Autoria própria

As respostas à terceira pergunta identificaram que as informações apresentadas na tarefa foram suficientes para sua realização, conforme apresentado na Figura 16.

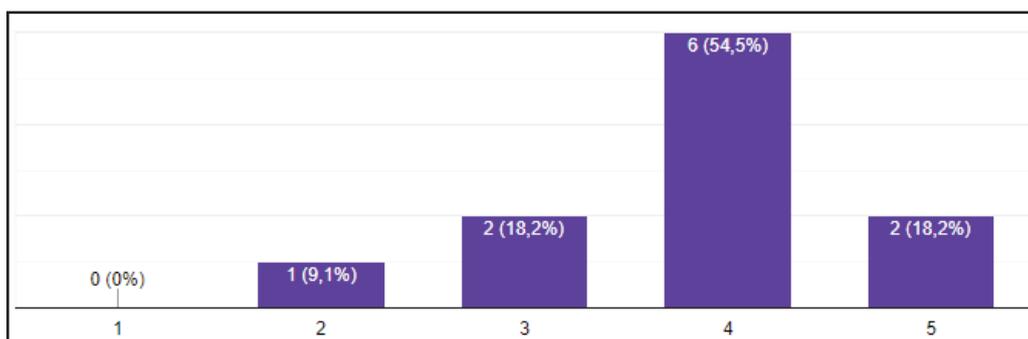


Figura 16. Respostas sobre apresentação das tarefas
Fonte: Autoria própria

As respostas à quarta pergunta identificaram que a documentação fornecida para realização das tarefas foi suficiente para realização dessas, conforme Figura 17.

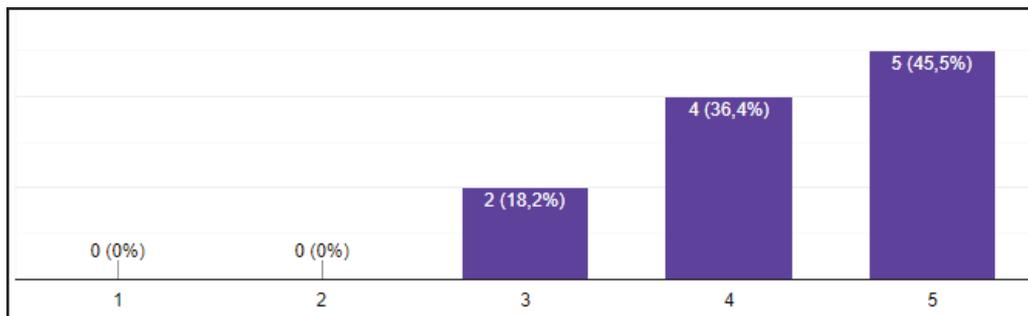


Figura 17. Respostas sobre a documentação das tarefas
Fonte: Autoria própria

5.5.4 Questionário Fase 2

No questionário da Fase 2 foram analisadas questões referentes ao guia de decomposição. As respostas avaliadas foram:

1. Você realizou a decomposição de tarefa utilizando o guia?
2. O guia auxiliou na realização da decomposição da tarefa.
3. O guia auxiliou para a compreensão das tarefas.
4. O guia e a decomposição criada auxiliaram na resolução das tarefas.

Na primeira questão, foi identificado que 90,9% dos participantes utilizou o guia para realizar a decomposição das tarefas e apenas 9,1% deixou de utilizar. A segunda questão identificou que, em linhas gerais, o guia auxiliou na realização da decomposição das tarefas, conforme apresentado na Figura 18.

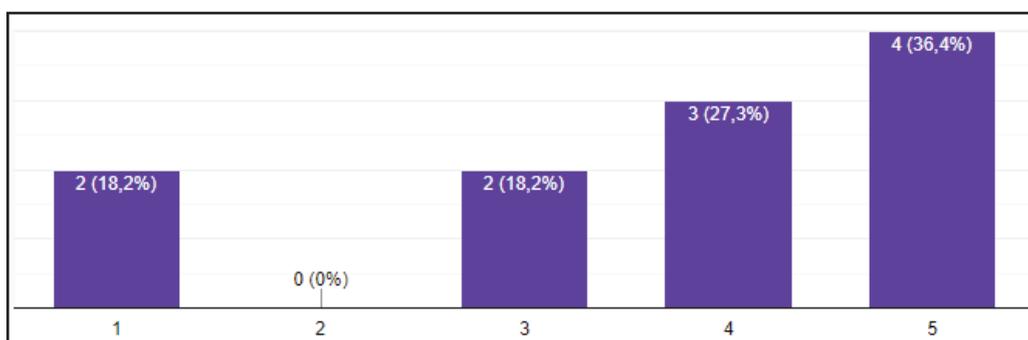


Figura 18. Respostas sobre o auxílio do guia na decomposição
Fonte: Autoria própria

A terceira questão identificou que, em geral, o guia auxiliou na compreensão das tarefas, conforme apresentado na Figura 19.

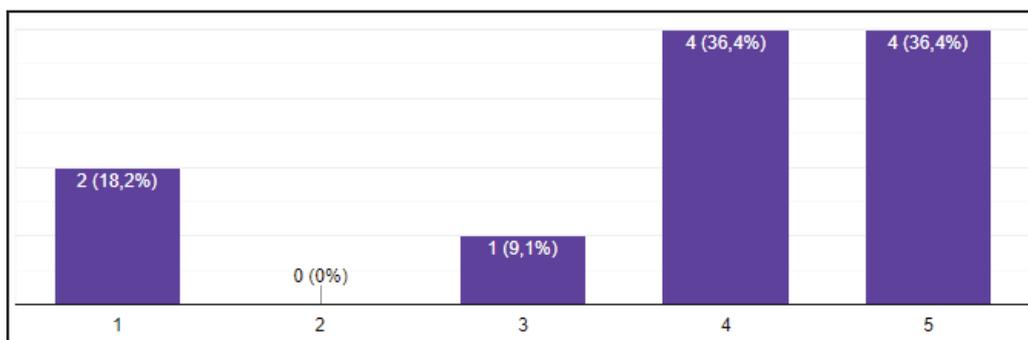


Figura 19. Respostas sobre o auxílio do guia na compreensão da tarefa
Fonte: Autoria própria

A quarta pergunta apresentou que 8 participantes demonstraram que o guia e a decomposição auxiliaram na resolução das tarefas propostas, conforme Figura 20.

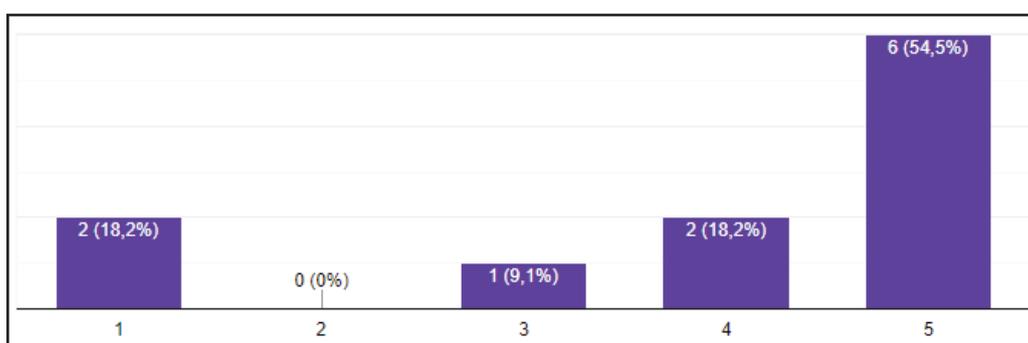


Figura 20. Respostas sobre o auxílio do guia e a decomposição na resolução das tarefas
Fonte: Autoria própria

As próximas questões, identificadas a seguir, se relacionavam com a utilização do guia em plataformas comerciais de *crowdsourcing*.

1. Você considera que em uma plataforma comercial o guia auxiliaria na resolução da tarefa?
2. Você utilizaria o guia para resolver tarefas em uma plataforma comercial?

Para as duas perguntas, 100% dos participantes responderam de forma positiva, considerando relevante a utilização do guia em plataformas comerciais.

As questões a seguir foram referentes à desistência na finalização das tarefas.

1. Você desistiu de finalizar alguma tarefa?
2. Você realizou a decomposição da tarefa e após desistiu de executá-la?
3. Por que desistiu da tarefa após realizar a decomposição?

A terceira pergunta apresenta os motivos da desistência das tarefas indicados pelos participantes, conforme apresentado na Figura 21.

Por que desistiu da tarefa após realizar a decomposição?

As tarefas que foram feitas decomposição foram realizadas.

Não desisti.

Não compreendi

Não

Provas e trabalhos

Dificuldade

.

Não desisti de nenhuma tarefa

Falta de tempo

Não desisti, eu não comecei na verdade

Tempo

Figura 21. Respostas sobre o motivo da desistência na resolução das tarefas
Fonte: Autoria própria

A primeira pergunta obteve uma resposta positiva por parte de 63,3% dos participantes e a segunda resposta foi sinalizada positivamente por 54,5%.

6. DISCUSSÃO DOS RESULTADOS

O objetivo do trabalho foi de propor um guia para auxiliar os trabalhadores na submissão de tarefas complexas em software *crowdsourcing*. A seguir serão discutidos os pilares na construção do guia e os resultados qualitativos:

- **Melhoria do Processo de testes (MPT):** Conforme previamente destacado, o *crowd testing* trata a realização dos testes de forma apartada ao restante do projeto. Com isso, foi observada a necessidade de incorporar um padrão para auxiliar na execução das atividades de teste, visto que há diversidade de pessoas e rotatividade das campanhas de teste, conforme apontado por Hossfeld et al. [22]. O MPT por ser um modelo de maturidade que trata da melhoria do processo, foi incorporado ao guia com o intuito de utilizar as boas práticas aplicadas na realização do teste, pois geralmente são fornecidos mecanismos limitados de garantia de qualidade [22]. O foco esteve na abordagem prevista em um dos níveis iniciais de maturidade processo, o PET. Assim, atentou-se para a identificação, a elaboração e a execução dos casos de teste, tendo em vista também um padrão para reportar defeitos encontrados e a realização de seu acompanhamento, conforme detalhado no MPT [32].

Outro ponto a ser observado é a diversidade apresentada nos projetos de *crowdsourcing*, apresentando diferentes níveis e experiências, conforme descrito por Tajedin e Nevo [10] e Mao et al. [6], o que reitera a utilização de um processo para auxiliar na realização dos testes. Essa variedade nos níveis de experiência foi verificada no experimento e mesmo apresentando um número limitado de participantes, demonstrou alunos sem experiência, menos de um ano e alunos com 1 a 5 anos de experiência.

- **Análise hierárquica e documentação:** Complementando o MPT, observa-se a utilização da análise hierárquica de tarefas. O HTA foi incorporado ao guia pois pode descrever como as tarefas do usuário podem ser realizadas. Tendo em vista esse ponto, a utilização da técnica contribui indiretamente para que as barreiras de documentação pouco claras e o não entendimento da tarefa sejam reduzidas, visto que estimula o usuário a descrever as validações a serem realizadas. Conforme Mao et al. [6], a documentação desempenha um papel crucial e causa defeitos quando é imprecisa. Essas anotações propostas pelo HTA oferecem uma base inicial para organização da documentação e para posterior execução dos testes.
- **Realização da decomposição e utilização do guia:** Em contrapartida aos resultados da análise quantitativa, observa-se as respostas ao questionário. A utilização do guia foi realizada por 90,9% dos participantes do experimento. Observa-se que esse cenário vai ao encontro a necessidade da realização de atividades da disciplina, pois os dois pontos em conjunto levaram os participantes a realizar a decomposição e a não submeterem a tarefa. Embora sem resultados na submissão das tarefas, o percentual demonstra uma aderência satisfatória na utilização do guia.

O questionário demonstrou que além de utiliza-lo no experimento, os participantes consideraram que o guia os auxiliou na realização da decomposição, tendo respostas positivas. Além disso, o guia foi considerado importante na compreensão das tarefas disponibilizadas no experimento e, posteriormente, agregado a decomposição gerada, importante para realizar a resolução da tarefa.

Outro item que demonstrou resultado satisfatório foi a estrutura da decomposição criada e entregue pelos participantes. Apenas dois realizaram a entrega, porém ficaram dentro do esperado, respeitando a estrutura em formato de árvore, mapeando as validações das tarefas, conforme demonstrado na Seção 5.4.1.

- **Impacto de atividades externas:** Tendo em vista os resultados, foi demonstrado na análise quantitativa, que o guia proposto não apresentou o efeito previsto, inicialmente, pois o número de submissões de tarefas na Fase 1 foi maior que na Fase 2. Um dos fatores que influenciaram esse cenário foi de que, na Fase 2, os participantes possuíam atividades da disciplina com prazo próximo para entrega, sendo essa uma das percepções do autor do trabalho. Conforme descrito por Mao et al. [6], as tarefas complexas exigem cargas de trabalho pesadas e os recursos dedicados, tendo impacto na resolução caso esse procedimento não seja seguido.

Outro ponto que corrobora essa ideia é o fato de que, na Fase 1, apenas dois participantes não submeteram tarefas e, na Fase 2, foram oito participantes sem nenhuma tarefa submetida, conforme demonstrado na Tabela 9. As respostas do questionário enfatizam a influência de atividades externas na participação no experimento. Observa-se que, mesmo o questionário apontando que os participantes realizaram a decomposição utilizando o guia, a questão "Você desistiu de finalizar alguma tarefa?" indicou que mais de 50% dos participantes desistiram e a resposta à questão "Você realizou a decomposição da tarefa e após desistiu de executá-la?" também indicou uma desistência acima de 50%. Com isso nota-se que a realização do experimento em outro momento do semestre, possivelmente poderia apresentar resultados diferentes, corroborando com a ideia de que os participantes precisam ter foco nas atividades, conforme estudo de Mao et al. [6].

- **Decomposição tratada como atividade extra:** Um item a ser observado, conforme limitação apresentada na Seção 7.2, é de que a decomposição poderia ser tratada como uma atividade extra e onerar tempo na solução da tarefa. Com isso, ao iniciar a solução de uma tarefa, o usuário precisa considerar a criação da decomposição, sendo que essa atividade será agregada ao tempo total da tarefa. Dessa forma, é reforçada a importância de os usuários administrarem seu tempo e superar essa barreira, conforme descrito no estudo de Zanatta et al. [4]. Previamente a esse ponto, foi observado na questão aberta "Caso não tenha utilizado o guia, informe o motivo da não utilização do mesmo.". Na resposta dada nesse questionamento pelo participante que não utilizou o guia foi: "Acabei não utilizando por falta de tempo". Observa-se que essa resposta é um indicativo de que a utilização do guia tornaria maior o tempo de resolução da tarefa.
- **Problemas relacionados ao gerenciamento do tempo:** Percebe-se que o experimento expôs outra barreira: a dificuldade de gerenciamento do tempo pessoal. O prazo dos alunos estava curto e como a participação no experimento não era obrigatória, optaram por utilizar o período indicado para realização do experimento para se dedicarem as atividades da disciplina. A atividade em questão foi disponibilizada pelo professor no decorrer do semestre, tendo prazo até o final do semestre para ser concluída. Isso enfatiza a dificuldade de organizar o tempo para desenvolvimento da atividade, necessitando utilizar um período destinado a outras atividades para concluir o trabalho. Essa barreira foi apontada por Zanatta et al. [4], tendo impacto na finalização das tarefas.

Esse ponto se confirmou na análise dos questionários da Fase 2, apresentado na Seção 5.5.4. Em uma pergunta aberta, os participantes foram questionados do motivo da desistência na realização da tarefa após a realização da decomposição, houve as seguintes respostas:

- Tempo
 - Falta de tempo
 - Provas e trabalhos
 - Não desisti, eu não comecei na verdade
- **Problemas relacionados a colaboração e comunicação:** Outra barreira demonstrada no experimento se relacionou aos problemas de colaboração e comunicação, conforme relatado no experimento de Zanatta et al. [4]. No experimento, foi utilizado o Moodle para realizar a comunicação, tendo o autor do trabalho para centralizar as dúvidas dos participantes. Nota-se a importância do moderador e do impacto na sequência das atividades caso o mesmo não participe da comunidade, dando suporte as dúvidas e problemas. As mesmas foram publicadas no fórum e já respondidas em seguida. Durante o experimento, também ocorreu um problema no acesso ao fórum, sendo resolvido na sequência.

Observa-se também que a comunicação entre os alunos na Fase 1, foi maior que a da Fase 2, conforme verificado pelo autor do trabalho durante o experimento. As atividades externas realizadas pelos alunos, influenciaram na sua comunicação durante o experimento. Por mais que em *crowdsourcing* os participantes interagem a distância, nada os impede de reunir um grupo para realizar as atividades.

7. CONCLUSÕES

Este capítulo encerra o presente trabalho que apresentou uma proposta de criação e utilização do guia para decomposição de tarefas, suas contribuições, limitações e uma perspectiva para trabalhos futuros.

O avanço da tecnologia fez com que o desenvolvimento de software fosse aprimorado. Essa atividade envolve um conjunto de papéis para adicionar singularidades a cada projeto. Foi em 2006 que o termo *crowdsourcing* foi definido como um novo modelo de negócio que apresenta certas similaridades com os modelos existentes, porém com adaptações em alguns pontos. Contudo, esse modelo apresenta algumas barreiras na participação dos usuários, tendo como uma das razões a complexidade das tarefas disponibilizadas. A literatura apresenta tentativas de diminuir essas barreiras utilizando mecanismos para decomposição de uma tarefa.

O trabalho forneceu uma forma de realização da decomposição de tarefas de testes funcionais pelo usuário da plataforma de *crowdsourcing*. Tendo como ponto de partida a realização da análise de conteúdo nas ferramentas, a técnica de decomposição de tarefas HTA e o MPT, foi desenvolvido um guia para realização da decomposição de tarefas de testes funcionais. A utilização do guia foi destinada a ser realizada após a seleção da tarefa na plataforma, entendendo que esse processo poderia aumentar o tempo de realização da tarefa.

Para avaliação do guia, foi realizado um experimento pareado utilizando uma amostra selecionada por conveniência. Foi feita uma avaliação quantitativa na qual o guia não apresentou os resultados esperados. A avaliação qualitativa realizada por meio de um questionário, demonstrou que os participantes utilizaram o guia para realização da decomposição e que auxiliou na realização da decomposição das tarefas.

7.1 CONTRIBUIÇÕES

O desenvolvimento do trabalho proporcionou a compreensão sobre a realização da decomposição de tarefas, sendo as principais elencadas a seguir:

- **Estrutura das tarefas:** por meio da análise de conteúdo, verificou-se um padrão na estrutura das tarefas de teste de software, sendo apresentada nas plataformas estudadas.
- **O guia:** embora não tenha apresentado os resultados esperados na avaliação quantitativa, o guia para decomposição de tarefas apresentou resultados satisfatórios na visão dos usuários observado na análise qualitativa. Observa-se a aderência do guia na visão dos usuários, demonstrando a importância da utilização do guia na decomposição das tarefas. Com isso possibilita-se o avanço no estudo e a aplicabilidade do guia, desde a aplicação com uma amostra maior de participantes como a utilização com outros tipos de tarefas.
- **Realização da decomposição:** abre-se uma possibilidade de novos estudos da realização da decomposição pelo usuário. Isso vai de encontro às propostas de decompo-

sição apresentadas na literatura, pois se apresenta como sendo de responsabilidade da plataforma a realização da decomposição.

7.2 LIMITAÇÕES DO TRABALHO

As restrições deste trabalho se relacionam aos métodos e aos procedimentos bem como aos resultados apresentados.

Inicialmente, destaca-se a limitação em relação ao período da realização do experimento. Como os alunos possuíam atividades da disciplina próximas da entrega, utilizaram o período em sala de aula para realização dos trabalhos. Como a participação era optativa, houve influência na participação dos alunos na realização das tarefas. Além do período em sala de aula, esse ponto influenciou no decorrer da semana, quando as tarefas do experimento seguiam disponíveis, porém, limitando-se à rede interna da UPF.

Em relação ao experimento, outra limitação que se apresenta é a realização de apenas uma fase do experimento. Com isso não foi possível realizar avaliações na primeira fase para serem aplicadas melhorias em uma nova fase. Além disso, observa-se que o tamanho da amostra é outro fator limitante do experimento.

Outro ponto a ser observado é o relatório dos *bugs*, o qual foi realizado individualmente, sendo que não havia controle de *bugs* reportados por outro participante. Considera-se uma limitação, pois esse controle é realizado no *crowdsourcing*. Outra questão a ser observada é de que o guia pode ser tratado como mais uma tarefa a ser realizada durante o processo, sendo que isso pode onerar tempo do testador.

Em resumo, percebe-se um conjunto de fatores limitantes para o trabalho. Contudo, essas restrições podem representar a condução de novas investigações apresentadas a seguir.

7.3 TRABALHOS FUTUROS

Dentre os pontos observados durante a realização do trabalho, evidencia-se a possibilidade da realização do experimento em um período no qual atividades externas não influenciem nos resultados do experimento. Também verifica-se a possibilidade da realização de novas fases do experimento, realizando avaliações com intuito de melhorar o guia proposto.

Existem também oportunidades de disseminar a pesquisa e ampliar a aplicação do guia de decomposição de tarefas. Nesse ponto, destaca-se a utilização do guia em plataformas comerciais. Além disso, torna-se relevante realizar uma adaptação no guia para aplicação em outras tarefas além das que se referem a testes funcionais.

Sendo assim, tem-se em vista um cenário produtivo de software e a demanda para o uso do *crowdsourcing*, observa-se que há uma área favorável para realização de novas pesquisas.

REFERÊNCIAS BIBLIOGRÁFICAS

- [1] BEMBEM, A. H. C.; SANTOS, P. L. V. A. d. et al. Inteligência coletiva: um olhar sobre a produção de pierre lévy. *Perspectivas em ciência da informação*, Escola de Ciência da Informação da UFMG, p. 139–151, 2013.
- [2] HOWE, J. The rise of crowdsourcing. *Wired magazine*, v. 14, n. 6, p. 1–4, 2006.
- [3] BRABHAM, D. C. *Crowdsourcing*. [S.l.]: Mit Press, 2013.
- [4] ZANATTA, A. L.; MACHADO, L.; STEINMACHER, I. Competence, collaboration, and time management: barriers and recommendations for crowdworkers. In: ACM. *Proceedings of the 5th International Workshop on Crowd Sourcing in Software Engineering*. [S.l.], 2018. p. 9–16.
- [5] ZANATTA, A. L. et al. Barriers faced by newcomers to software-crowdsourcing projects. *IEEE Software*, IEEE, v. 34, n. 2, p. 37–43, 2017.
- [6] MAO, K. et al. A survey of the use of crowdsourcing in software engineering. *Rn*, v. 15, n. 01, 2015.
- [7] LATOZA, T. D. et al. Microtask programming: Building software with a crowd. In: ACM. *Proceedings of the 27th annual ACM symposium on User interface software and technology*. [S.l.], 2014. p. 43–54.
- [8] LATOZA, T. D.; HOEK, A. V. D. Crowdsourcing in software engineering: Models, motivations, and challenges. *IEEE software*, IEEE, v. 33, n. 1, p. 74–80, 2015.
- [9] MORRIS, R. R.; DONTCHEVA, M.; GERBER, E. M. Priming for better performance in microtask crowdsourcing environments. *IEEE Internet Computing*, IEEE, v. 16, n. 5, p. 13–19, 2012.
- [10] TAJEDIN, H.; NEVO, D. Determinants of success in crowdsourcing software development. In: ACM. *Proceedings of the 2013 annual conference on Computers and people research*. [S.l.], 2013. p. 173–178.
- [11] STOL, K.-J.; FITZGERALD, B. Two's company, three's a crowd: a case study of crowdsourcing software development. In: ACM. *Proceedings of the 36th International Conference on Software Engineering*. [S.l.], 2014. p. 187–198.
- [12] VAZ, L.; STEINMACHER, I.; MARCZAK, S. An empirical study on task documentation in software crowdsourcing on topcoder. In: IEEE. *2019 ACM/IEEE 14th International Conference on Global Software Engineering (ICGSE)*. [S.l.], 2019. p. 48–57.
- [13] STOL, K.-J.; CAGLAYAN, B.; FITZGERALD, B. Competition-based crowdsourcing software development: A multi-method study from a customer perspective. *IEEE Transactions on Software Engineering*, IEEE, v. 45, n. 3, p. 237–260, 2017.
- [14] LI, Z.; HONGJUAN, Z. Research of crowdsourcing model based on case study. In: IEEE. *ICSSSM11*. [S.l.], 2011. p. 1–5.

- [15] COCACOLA. *Coca-Cola Launches Crowdsourcing Challenges to Find Next-Generation Sweetener Innovations*. Disponível em: <<https://www.coca-colacompany.com/stories/coca-cola-launches-crowdsourcing-challenges-to-find-next-generat>>. Acesso em: 08 fev. 2020.
- [16] B9. *A internet transformou todo mundo em detetive?* Disponível em: <<https://www.b9.com.br/36830/internet-transformou-todo-mundo-em-detetive/>>. Acesso em: 08 fev. 2020.
- [17] HINCHCLIFFE, D. Crowdsourcing: 5 reasons it's not just for startups any more. *Ebizq*, URL: http://www.ebizq.net/blogs/enterprise/2009/09/crowdsourcing_5_reasons_its_no.php, 2009.
- [18] ÅGERFALK, P. J.; FITZGERALD, B.; STOL, K.-J. *Software Sourcing in the Age of Open: Leveraging the Unknown Workforce*. [S.l.]: Springer, 2015.
- [19] HOSSEINI, M. et al. The four pillars of crowdsourcing: A reference model. In: IEEE. *2014 IEEE Eighth International Conference on Research Challenges in Information Science (RCIS)*. [S.l.], 2014. p. 1–12.
- [20] SOMMERVILLE, I. *Engenharia de Software. ed.* [S.l.]: São Paulo, SP: Pearson Prentice Hall, 2011.
- [21] MALDONADO, J. C. et al. Introdução ao teste de software. *São Carlos*, p. 23, 2004.
- [22] HOSSFELD, T. et al. Best practices for qoe crowdtesting: Qoe assessment with crowdsourcing. *IEEE Transactions on Multimedia*, IEEE, v. 16, n. 2, p. 541–558, 2013.
- [23] QAINFOTECH. *Crowd Sourced Testing – is it really for you?* Disponível em: <<https://qainfotech.com/crowd-sourced-testing-is-it-really-for-you/>>. Acesso em: 08 fev. 2020.
- [24] NOVECK, B. S. *Wiki government: How technology can make government better, democracy stronger, and citizens more powerful*. [S.l.]: Brookings Institution Press, 2009.
- [25] ALTMANN, L. D. Uma proposta de ferramenta para recomendações em software crowdsourcing. 2018.
- [26] ROHRMEIER, P. *Social Networks and Online Communities*. Tese (Doutorado) — Technische Universität München, 2012.
- [27] PADOVANI, S.; SMYTHE, K. C. A. S. Investigando a compreensão de representações diagramáticas utilizadas em análise da tarefa: um estudo comparativo entre modelagem hierárquica e seqüencial. *InfoDesign-Revista Brasileira de Design da Informação*, v. 8, n. 2, p. 25–38, 2011.
- [28] MILITELLO, L. G.; HUTTON, R. J. Applied cognitive task analysis (acta): a practitioner's toolkit for understanding cognitive task demands. *Ergonomics*, Taylor & Francis, v. 41, n. 11, p. 1618–1641, 1998.
- [29] MENEZES R. C., P. A. S. D.; FRANZ, L. A. d. S. Aplicação da análise hierárquica da tarefa e estudo de demandas ergonômicas em uma empresa de doces tradicionais de pelotas. *XXXV ENCONTRO NACIONAL DE ENGENHARIA DE PRODUCAO*, 2010.

- [30] SALMON, P. et al. Hierarchical task analysis vs. cognitive work analysis: comparison of theory, methodology and contribution to system design. *Theoretical Issues in Ergonomics Science*, Taylor & Francis, v. 11, n. 6, p. 504–531, 2010.
- [31] ELLWANGER, C.; SANTOS, C. P. Avaliação de usabilidade de sistemas sob a ótica da estruturação conceitual hierárquica. In: *XVIII Conferência Internacional sobre Informática na Educação (TISE 2013)*. [S.l.: s.n.], 2013.
- [32] SOFTEX. *Melhoria do Processo de Teste Brasileiro*. Disponível em: <<http://mpt.org.br/>>. Acesso em: 09 fev. 2020.
- [33] KITTUR, A. et al. Crowdforge: Crowdsourcing complex work. In: *Proceedings of the 24th annual ACM symposium on User interface software and technology*. [S.l.: s.n.], 2011. p. 43–52.
- [34] SHU, Y. et al. The verification approach to complex tasks' functional specification in software crowdsourcing. In: IEEE. *2016 5th International Conference on Computer Science and Network Technology (ICCSNT)*. [S.l.], 2016. p. 171–176.
- [35] JIANG, H.; MATSUBARA, S. Efficient task decomposition in crowdsourcing. In: SPRINGER. *International Conference on Principles and Practice of Multi-Agent Systems*. [S.l.], 2014. p. 65–73.
- [36] DUBEY, A. et al. Dynamics of software development crowdsourcing. In: IEEE. *2016 IEEE 11th International Conference on Global Software Engineering (ICGSE)*. [S.l.], 2016. p. 49–58.
- [37] BARDIN, L. *Análise de Conteúdo*. [S.l.]: Editora Almedina, 2011.
- [38] ZANATTA, A. L. et al. Software crowdsourcing platforms. *IEEE Software*, IEEE, v. 33, n. 6, p. 112–116, 2016.
- [39] LI, K. et al. Analysis of the key factors for software quality in crowdsourcing development: An empirical study on topcoder. com. In: IEEE. *2013 IEEE 37th Annual Computer Software and Applications Conference*. [S.l.], 2013. p. 812–817.
- [40] WOHLIN, C. et al. *Experimentation in software engineering*. [S.l.]: Springer Science & Business Media, 2012.
- [41] MAO, K. et al. Developer recommendation for crowdsourced software development tasks. In: IEEE. *2015 IEEE Symposium on Service-Oriented System Engineering*. [S.l.], 2015. p. 347–356.
- [42] GERHARDT, T. E.; SILVEIRA, D. T. *Métodos de pesquisa*. [S.l.]: Plageder, 2009.
- [43] LOPES, M. de M.; BRANCO, V. T. F. C.; SOARES, J. B. Utilização dos testes estatísticos de kolmogorov-smirnov e shapiro-wilk para verificação da normalidade para materiais de pavimentação. *Transportes*, v. 21, n. 1, p. 59–66, 2013.
- [44] BARBETTA, P. A. *Estatística aplicada às ciências sociais*. [S.l.]: Ed. UFSC, 2008.
- [45] GODOY, A. S. Pesquisa qualitativa: tipos fundamentais. *Revista de Administração de empresas*, SciELO Brasil, v. 35, n. 3, p. 20–29, 1995.

- [46] CRYSTAL, A.; ELLINGTON, B. Task analysis and human-computer interaction: approaches, techniques, and levels of analysis. *AMCIS 2004 Proceedings*, p. 391, 2004.
- [47] SOFTEX, R. Melhoria do processo de teste brasileiro–mpt. *BR. Guia de Referência do Modelo*. Disponível em: <http://mpt.org.br/mpt/mpt/modelo-de-referencia-e-guia-deavaliacao>.
- [48] UFF. *Tabela T: Distribuição de t-Student segundo os graus de liberdade e uma dada probabilidade num teste bicaudal*. Disponível em: <<http://www.epi.uff.br/wp-content/uploads/2015/05/Tabela-T.pdf>>. Acesso em: 08 fev. 2020.

APÊNDICE A – QUESTIONÁRIO - RODADA 1

Questionário - Rodada 1

***Obrigatório**

Informações pessoais

1. Informe seu nome completo *

2. Informe seu endereço de e-mail *

3. Qual é sua idade? *

Marcar apenas uma oval.

- Menos de 20 anos.
- Entre 20 e 25 anos.
- Entre 26 e 30 anos.
- Mais de 30 anos.

Informações profissionais

4. Além de estudante, você desenvolve alguma atividade profissional? *

Marcar apenas uma oval.

- Não
- Sim

5. Quanto tempo de experiência em atividades relacionadas a teste de software você possui? *

Marcar apenas uma oval.

- Não tenho experiência
- Menos de 1 ano
- Entre 1 e 5 anos
- Mais de 5 anos

Informações gerais sobre Crowdsourcing

6. Você conhecia o desenvolvimento de software Crowdsourcing? *

Marcar apenas uma oval.

- Sim
- Não

7. Já participou em projetos de desenvolvimento de software via Crowdsourcing? *

Marcar apenas uma oval.

- Sim
- Não

Referente a realização das tarefas

8. Sobre a realização das tarefas, possui dificuldade na execução. *

Marcar apenas uma oval.

1 2 3 4 5

Discordo totalmente Concordo totalmente

9. Sobre as tarefas, considero a tarefa realizada complexa.

Marcar apenas uma oval.

1 2 3 4 5

Discordo totalmente Concordo totalmente

10. A forma de apresentação/descrição da tarefa foi suficiente para sua realização.

*

Marcar apenas uma oval.

1 2 3 4 5

Discordo totalmente Concordo totalmente

11. Em geral, houve documentação de apoio a realização das tarefas. *

Marcar apenas uma oval.

1 2 3 4 5

Discordo totalmente Concordo totalmente

12. A documentação foi suficiente para o entendimento e realização da tarefa. *

Marcar apenas uma oval.

1 2 3 4 5

Discordo totalmente Concordo totalmente

13. Você desistiu de finalizar alguma tarefa? *

Marcar apenas uma oval.

Sim

Não

14. Você buscou algum tipo de ajuda/apoio no fórum para realizar a tarefa? *

Marcar apenas uma oval.

Sim

Não

15. Você encontrou algum problema para realizar a tarefa? (considere questões técnicas e não-técnicas) *

Marcar apenas uma oval.

Sim

Não

16. Qual(is) foi(foram) o(s) problema(s) que você encontrou para realizar a tarefa? (considere questões técnicas e não-técnicas)

Itens em geral

17. Você percebeu a existência da figura (ou papel) de um intermediador durante a execução da tarefa? *

Marcar apenas uma oval.

- Não tenho condições de responder
- Sim
- Não

18. Foi suficiente/adequado a coordenação dos artefatos, plano e prazos para a execução da tarefa. *

Marcar apenas uma oval.

	1	2	3	4	5	
Discordo totalmente	<input type="radio"/>	Concordo totalmente				

19. É possível desenvolver software utilizando o modelo crowdsourcing? *

Marcar apenas uma oval.

- Sim
- Não

Este conteúdo não foi criado nem aprovado pelo Google.

Google Formulários

APÊNDICE B – QUESTIONÁRIO - RODADA 2

Questionário - Rodada 2

*Obrigatório

Informações pessoais

1. Informe seu nome completo *

2. Informe seu endereço de e-mail *

Questionamentos sobre o Guia de decomposição

3. Você realizou a decomposição de tarefa utilizando o guia? *

Marcar apenas uma oval.

Sim

Não

4. Caso não tenha utilizado o guia, informe o motivo da não utilização do mesmo.

5. O guia auxiliou na realização da decomposição da tarefa. *

Marcar apenas uma oval.

	1	2	3	4	5	
Discordo totalmente	<input type="radio"/>	Concordo totalmente				

6. O guia auxiliou para a compreensão das tarefas. *

Marcar apenas uma oval.

	1	2	3	4	5	
Discordo totalmente	<input type="radio"/>	Concordo totalmente				

7. O guia e a decomposição criada auxiliaram na resolução das tarefas. *

Marcar apenas uma oval.

	1	2	3	4	5	
Discordo totalmente	<input type="radio"/>	Concordo totalmente				

8. A documentação foi suficiente para o seu entendimento e posterior realização da tarefa. *

Marcar apenas uma oval.

	1	2	3	4	5	
Discordo totalmente	<input type="radio"/>	Concordo totalmente				

9. Você considera que em uma plataforma comercial o guia auxiliaria na resolução da tarefa? *

Marcar apenas uma oval.

Sim

Não

10. Você utilizaria o guia para resolver tarefas em uma plataforma comercial? *

Marcar apenas uma oval.

Sim

Não

Itens em geral

11. Você desistiu de finalizar alguma tarefa? *

Marcar apenas uma oval.

Sim

Não

12. Você realizou a decomposição da tarefa e após desistiu de executá-la? *

Marcar apenas uma oval.

Sim

Não

13. Das tarefas oferecidas, você não executou alguma? *

Marcar apenas uma oval.

Sim

Não

14. Você encontrou algum problema para realizar a tarefa? (considere questões técnicas e não-técnicas) *

Marcar apenas uma oval.

Sim

Não

15. Por que desistiu da tarefa após realizar a decomposição? *

16. Qual(is) foi(foram) o(s) problema(s) que você encontrou para realizar a tarefa? (considere questões técnicas e não-técnicas)

Este conteúdo não foi criado nem aprovado pelo Google.

Google Formulários

APÊNDICE C – DOCUMENTO DE LIVRE CONSENTIMENTO

TERMO DE CONSENTIMENTO LIVRE E ESCLARECIDO

Declaro, por meio deste termo, que concordei em ser entrevistado(a) e/ou participar na pesquisa de campo referente ao projeto/pesquisa intitulado(a) *Experimento na Realização de Tarefas Crowdsourcing* desenvolvida(o) por *Fernando Luiz Costella*. Fui informado(a), ainda, de que a pesquisa é coordenada pelo professor *Alexandre Lazaretti Zanatta*, a quem poderei contatar / consultar a qualquer momento que julgar necessário.

Afirmo que aceitei participar por minha própria vontade, sem receber qualquer incentivo financeiro ou ter qualquer ônus e com a finalidade exclusiva de colaborar para o sucesso da pesquisa. Fui informado(a) dos objetivos estritamente acadêmicos do estudo.

Fui também esclarecido(a) de que os usos das informações por mim oferecidas estão submetidos às normas éticas destinadas à pesquisa.

Minha colaboração será realizada por meio da realização de tarefas pré-determinadas em uma plataforma desenvolvida, a partir da assinatura desta autorização. O acesso e a análise dos dados coletados se farão apenas pelo(a) pesquisador(a) e/ou seu(s) orientador(es) / coordenador(es).

Fui ainda informado(a) de que posso me retirar desse(a) estudo / pesquisa / programa a qualquer momento, sem prejuízo para meu acompanhamento ou sofrer quaisquer sanções ou constrangimentos.

Passo Fundo, _____ de _____ de _____.

Assinatura do(a) participante: _____

Assinatura do(a) pesquisador: _____

APÊNDICE D – PROTOCOLO DO EXPERIMENTO

Protocolo de estudo pareado

1. Definição do estudo

O objetivo desse trabalho é auxiliar o usuário de plataformas de crowdsourcing a realizar a decomposição de uma determinada tarefa de Testes Funcionais, através do “Guia para de composição de macrotasks”. A hipótese levantada é se os alunos que utilizarão o guia estarão mais propensos a submeter uma tarefa na plataforma e se os mesmos realizarão a decomposição das tarefas utilizando o guia. Para isso, será realizado um estudo pareado, que consiste em uma pesquisa em que pessoas escolhidas por conveniência utilizarão o guia para realizar a decomposição das tarefas. Na primeira parte do experimento, realizarão as tarefas na plataforma sem utilizar o guia. Em um segundo momento o mesmo grupo de alunos receberá o guia.

Foi desenvolvida a plataforma, em sua versão beta, que será utilizada no experimento. Na plataforma estão contidas as tarefas que farão parte dos testes.

Para realização das atividades, na segunda parte do experimento, o grupo receberá recomendações e questionamentos que os auxiliem na submissão das tarefas.

2. Seleção do contexto

Será realizado um projeto offline com uma equipe de estudantes, utilizando tarefas desenvolvidas do site de Eventos da UPF.

3. Formulação das hipóteses

As hipóteses formuladas são:

H0: O guia proposto não auxilia na realização da decomposição da tarefa pelo usuário.

H1: O guia proposto auxilia na realização da decomposição da tarefa pelo usuário.

H0: A decomposição das tarefas não aumenta o número de submissões das tarefas.

H1: A decomposição das tarefas aumenta o número de submissões das tarefas.

4. Seleção das variáveis

Serão avaliadas: a variável independente complexidade das tarefas que afeta na desistência dos usuários, vista como variável dependente. A complexidade nas tarefas causa a desistência dos trabalhadores nas ferramentas de crowdsourcing.

5. Seleção dos sujeitos

A população do estudo será composta por 17 alunos selecionados por conveniência, de ambos os sexos, da disciplina de Teste de Software do professor Alexandre Lazaretti Zanatta. Por ser uma amostragem por conveniência, é considerada uma amostragem não probabilística.

6. Design do experimento

O design do experimento que será utilizado será o de um fator com dois tratamentos, onde o fator é o guia para decomposição de Macrotasks e os tratamentos são: a utilização do guia e a não utilização do guia.

Caso o número de sujeitos que inicia o primeiro tratamento é o mesmo que no segundo tratamento, há um balanceamento no número de sujeitos.

Exemplo de tratamento pareado, considerando a resolução da tarefa.

Sujeitos	Tratamento 1 (Sem utilizar o guia)	Tratamento 2 (Utilizando o guia)
1		
2		
3		
4		

6. Instrumentação

Informados abaixo a lista de documentos necessários para realização do experimento.

- Mecanismos de coleta de dados
 - Documento de consentimento
 - Formulários de caracterização de participantes
 - Material de treinamento
 - Formulário de defeito
 - Questionário

4. Local e data de realização dos testes

A realização do experimento será realizada com data previamente agendada. A primeira parte do experimento será realizada durante a aula da disciplina de Teste de Software. Cada tarefa terá um tempo determinado na ferramenta para realização. Os alunos deverão respeitar o tempo de duração para publicar a tarefa na ferramenta. A duração total do experimento será de duas semanas.

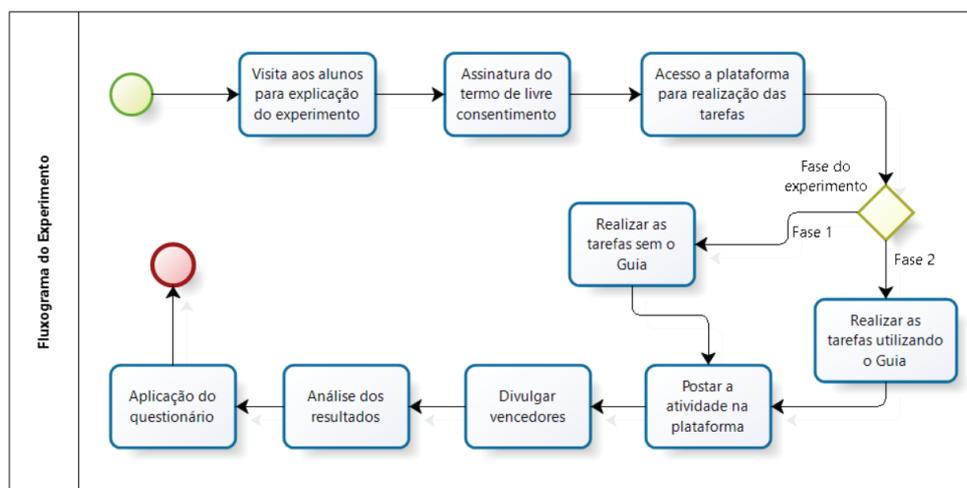
5. Metodologia

O experimento será realizado para verificar se os alunos que utilizarão o guia para decomposição de Macrotasks estarão mais propensos a submeter às tarefas. A realização do experimento seguirá os itens relatados abaixo.

- Os alunos serão visitados em sala de aula para uma explicação de como funcionará o experimento.
- Todos os participantes precisarão assinar o tempo de livre consentimento.
- Participantes irão acessar a plataforma para visualização e submissão das tarefas.
- O software que será utilizado para os testes será o do setor de Eventos da Universidade de Passo Fundo.
- Na primeira parte do experimento, os participantes realizarão as tarefas sem a utilização do “Guia para decomposição de Macrotasks”.

- Na segunda parte do experimento os participantes receberão o Guia para que os auxilie na realização das tarefas propostas. Os participantes da segunda parte do experimento são os mesmos da primeira parte da atividade.
- Os envolvidos ficarão disponíveis no fórum do Moodle durante a realização do experimento para eventuais dúvidas que os grupos venham a ter.
- O controle da realização das tarefas será realizado através da plataforma, com a avaliação manual por um dos responsáveis pelo experimento.
- Com a finalização das tarefas serão avaliadas as tarefas postadas na plataforma.
- Serão analisados os dados para validar hipótese.
- Aplicação de questionário final questionando se o guia auxiliou na decomposição das tarefas da plataforma e se auxiliou na finalização da tarefa.

Fluxograma de realização da execução do experimento:



7. Análise dos dados

A análise dos dados será baseada nas atividades realizadas pelos grupos sem a utilização do guia e utilizando o guia. A análise da realização das atividades dos grupos será fundamental para a comprovação ou não da hipótese. Será analisado se o grupo com o guia submeteu um maior número de atividades na plataforma.

APÊNDICE E – TAREFAS DO EXPERIMENTO - RODADA 1

[Tarefa 1 – Rodada 1] Realização de teste na funcionalidade de “Todos Eventos”

Situação: Ativa

Deadline: 06/11/2019 18:00

Descrição:

Esta tarefa é para realização dos testes da área do site onde estão localizados os eventos disponíveis para inscrição. A página contém as informações básicas do evento como a duração e o período de inscrições. Também é apresentada opção para visualizar mais detalhes do evento na própria página, assim como opção para ir a página do evento. A página do evento, apresenta mais informações do evento e também opção para realização da inscrição.

Essa atividade consiste em encontrar bugs na área de “Todos Eventos”, incluindo a página de detalhes, sendo localizadas na área logada da página de Eventos da UPF.

Instruções:

- Link dos testes:

Para realização dos testes, deve ser acessado o link abaixo com a URL do ambiente de homologação da página dos Eventos UPF.

Link: <http://eventos-test.upf.br/eventos-test>

- Abertura dos bugs:

Para abertura dos bugs, deve ser seguido o documento de instruções disponível no link abaixo. O documento deve conter as evidências dos testes realizados e a descrição de como deve ser reproduzido um bug.

Acesso: disponível na página do Moodle.

- Prints da tela:

Utilizar o “Ferramentas de captura” no Windows e o “Screenshot” no Linux.

- Gerador de CPF ou CNPJ

Para gerar um CPF ou CNPJ utilizar o link: <https://www.4devs.com.br>

Navegadores:

- Chrome
- Firefox
- Internet Explorer
- Safari

Dispositivos:

- Android
- IOS

Bugs aceitos:

- **Impeditivos:** impedem o uso de alguma funcionalidade do software. Não existem saídas ou alternativas para contorná-las.
- **Funcionais:** produzem um comportamento ilógico ou inesperado da aplicação onde o resultado obtido é diferente do esperado. A diferença entre uma falha funcional e uma falha impeditiva é que as falhas funcionais podem ser contornadas.
- **Texto:** qualquer ocorrência relacionada ao uso incorreto de algum idioma.
- **Interface:** são aquelas relacionadas à interface gráfica, à apresentação do software, exceto as falhas de texto. Exemplos: componentes desalinhados, renderização incorreta de interface, cores inconsistentes com guia de estilos, etc.
- **Melhorias:** não são propriamente falhas, mas sim sugestões para o aprimoramento do software. As sugestões são dadas com base na experiência do USUÁRIO e nos padrões de mercado.
- **Segurança:** são falhas que possibilitam acessar o sistema sem fazer uso de autenticação devida, sendo possível acessar funcionalidades, coletar dados, modificar dados, modificar o comportamento da aplicação ou interromper o funcionamento da aplicação.

Entregáveis:

- Lista de bugs abertos.

[Tarefa 2 – Rodada 1] Realização de teste na funcionalidade de “Meus Dados”

Situação: Ativa

Deadline: 06/11/2019 18:00

Descrição:

Esta tarefa é para realização dos testes da área onde estão os dados do usuário cadastrado na página de Eventos UPF. Nessa página estão contidas as áreas para alteração de dados pessoais, alteração de e-mail, alteração de senha e alteração dos dados bancários.

A área de “Alterar dados pessoais” possui os campos com as informações do usuário, no qual é possível realizar a alteração dos dados. Os campos apresentam informações pessoais juntamente com informações a respeito do endereço, possuindo informações obrigatórias demarcadas com asterisco.

A área de “Alterar E-mail” apresenta o campo de e-mail para informar o novo endereço, tendo que realizar a confirmação da senha para efetivar a alteração.

A área de “Alterar Senha” apresenta os campos para realizar a alteração da senha, tendo que realizar a confirmação da senha atual para efetivar a alteração.

A área de “Alterar Dados Bancários” permite realizar alteração nos dados bancários do usuário.

Essa atividade consiste em encontrar bugs na área de “Meus Dados” localizada na área logada da página de Eventos da UPF. Serão considerados bugs encontrados nas áreas de “Alterar Dados Pessoais”, “Alterar E-mail”, “Alterar Senha” e “Alterar Dados Bancários”.

Instruções:

- Link dos testes:

Para realização dos testes, deve ser acessado o link abaixo com a URL do ambiente de homologação da página dos Eventos UPF.

Link: <http://eventos-test.upf.br/eventos-test>

- Abertura dos bugs:

Para abertura dos bugs, deve ser seguido o documento de instruções disponível no link abaixo. O documento deve conter as evidências dos testes realizados e a descrição de como deve ser reproduzido um bug.

Acesso: disponível na página do Moodle.

- Prints da tela:

Utilizar o “Ferramentas de captura” no Windows e o “Screenshot” no Linux.

- Gerador de CPF ou CNPJ

Para gerar um CPF ou CNPJ utilizar o link: <https://www.4devs.com.br>

Navegadores:

- Chrome
- Firefox
- Internet Explorer
- Safari

Dispositivos:

- Android
- IOS

Bugs aceitos:

- **Impeditivos:** impedem o uso de alguma funcionalidade do software. Não existem saídas ou alternativas para contorná-las.
- **Funcionais:** produzem um comportamento ilógico ou inesperado da aplicação onde o resultado obtido é diferente do esperado. A diferença entre uma falha funcional e uma falha impeditiva é que as falhas funcionais podem ser contornadas.
- **Texto:** qualquer ocorrência relacionada ao uso incorreto de algum idioma.
- **Interface:** são aquelas relacionadas à interface gráfica, à apresentação do software, exceto as falhas de texto. Exemplos: componentes desalinhados, renderização incorreta de interface, cores inconsistentes com guia de estilos, etc.
- **Melhorias:** não são propriamente falhas, mas sim sugestões para o aprimoramento do software. As sugestões são dadas com base na experiência do USUÁRIO e nos padrões de mercado.
- **Segurança:** são falhas que possibilitam acessar o sistema sem fazer uso de autenticação devida, sendo possível acessar funcionalidades, coletar dados, modificar dados, modificar o comportamento da aplicação ou interromper o funcionamento da aplicação.

Entregáveis:

- Lista de bugs abertos.

[Tarefa 3 – Rodada 1] Realização de teste na funcionalidade de “Realizar Inscrição”

Situação: Ativa

Deadline: 06/11/2019 18:00

Descrição:

Esta tarefa é para realização dos testes na área de realização da inscrição para os eventos. Essa página contém a lista de todos os cursos disponíveis, sendo possível validar o número de vagas de cada evento, bem como o valor e tipo dos ingressos.

Essa atividade consiste em encontrar bugs na área de “Realizar Inscrição” localizada na área logada da página de Eventos da UPF.

Instruções:

- Link dos testes:

Para realização dos testes, deve ser acessado o link abaixo com a URL do ambiente de homologação da página dos Eventos UPF.

Link: <http://eventos-test.upf.br/eventos-test>

- Abertura dos bugs:

Para abertura dos bugs, deve ser seguido o documento de instruções disponível no link abaixo. O documento deve conter as evidências dos testes realizados e a descrição de como deve ser reproduzido um bug.

Acesso: disponível na página do Moodle.

- Prints da tela:

Utilizar o “Ferramentas de captura” no Windows e o “Screenshot” no Linux.

- Gerador de CPF ou CNPJ

Para gerar um CPF ou CNPJ utilizar o link: <https://www.4devs.com.br>.

Navegadores:

- Chrome
- Firefox
- Internet Explorer
- Safari

Dispositivos:

- Android
- IOS

Bugs aceitos:

- **Impeditivos:** impedem o uso de alguma funcionalidade do software. Não existem saídas ou alternativas para contorná-las.
- **Funcionais:** produzem um comportamento ilógico ou inesperado da aplicação onde o resultado obtido é diferente do esperado. A diferença entre uma falha funcional e uma falha impeditiva é que as falhas funcionais podem ser contornadas.
- **Texto:** qualquer ocorrência relacionada ao uso incorreto de algum idioma.
- **Interface:** são aquelas relacionadas à interface gráfica, à apresentação do software, exceto as falhas de texto. Exemplos: componentes desalinhados, renderização incorreta de interface, cores inconsistentes com guia de estilos, etc.
- **Melhorias:** não são propriamente falhas, mas sim sugestões para o aprimoramento do software. As sugestões são dadas com base na experiência do USUÁRIO e nos padrões de mercado.
- **Segurança:** são falhas que possibilitam acessar o sistema sem fazer uso de autenticação devida, sendo possível acessar funcionalidades, coletar dados, modificar dados, modificar o comportamento da aplicação ou interromper o funcionamento da aplicação.

Entregáveis:

- Lista de bugs abertos.

[Tarefa 4 – Rodada 1] Realização de teste na funcionalidade de “Meus eventos”

Situação: Ativa

Deadline: 06/11/2019 18:00

Descrição:

Esta tarefa é para realização dos testes na área onde estão contidos os eventos que o usuário se inscreveu. Essa área possibilita a visualização dos eventos com inscrições em andamento e os eventos anteriores que o usuário já participou. Nessa área também é possível visualizar detalhes dos eventos.

Essa atividade consiste em encontrar bugs na área de “Meus Eventos” localizada na área logada da página de Eventos da UPF.

Instruções:

- Link dos testes:

Para realização dos testes, deve ser acessado o link abaixo com a URL do ambiente de homologação da página dos Eventos UPF.

Link: <http://eventos-test.upf.br/eventos-test>

- Abertura dos bugs:

Para abertura dos bugs, deve ser seguido o documento de instruções disponível no link abaixo. O documento deve conter as evidências dos testes realizados e a descrição de como deve ser reproduzido um bug.

Acesso: disponível na página do Moodle.

- Prints da tela:

Utilizar o “Ferramentas de captura” no Windows e o “Screenshot” no Linux.

- Gerador de CPF ou CNPJ

Para gerar um CPF ou CNPJ utilizar o link: <https://www.4devs.com.br>

Navegadores:

- Chrome
- Firefox
- Internet Explorer
- Safari

Dispositivos:

- Android
- IOS

Bugs aceitos:

- **Impeditivos:** impedem o uso de alguma funcionalidade do software. Não existem saídas ou alternativas para contorná-las.
- **Funcionais:** produzem um comportamento ilógico ou inesperado da aplicação onde o resultado obtido é diferente do esperado. A diferença entre uma falha funcional e uma falha impeditiva é que as falhas funcionais podem ser contornadas.
- **Texto:** qualquer ocorrência relacionada ao uso incorreto de algum idioma.
- **Interface:** são aquelas relacionadas à interface gráfica, à apresentação do software, exceto as falhas de texto. Exemplos: componentes desalinhados, renderização incorreta de interface, cores inconsistentes com guia de estilos, etc.
- **Melhorias:** não são propriamente falhas, mas sim sugestões para o aprimoramento do software. As sugestões são dadas com base na experiência do USUÁRIO e nos padrões de mercado.
- **Segurança:** são falhas que possibilitam acessar o sistema sem fazer uso de autenticação devida, sendo possível acessar funcionalidades, coletar dados, modificar dados, modificar o comportamento da aplicação ou interromper o funcionamento da aplicação.

Entregáveis:

- Lista de bugs abertos.

[Tarefa 5 – Rodada 1] Realização de teste na funcionalidade de “Meus Pagamentos”

Situação: Ativa

Deadline: 06/11/2019 18:00

Descrição:

Esta tarefa é para realização dos testes na área onde estão contidos os eventos com as situações de pagamento. Essa área apresenta as informações de pagamento dos eventos que foram efetivadas as inscrições, podendo visualizar o valor, data do pagamento, status, entre outras informações.

Essa atividade consiste em encontrar bugs na área de “Meus Pagamentos” localizada na área logada da página de Eventos da UPF.

Instruções:

- Link dos testes:

Para realização dos testes, deve ser acessado o link abaixo com a URL do ambiente de homologação da página dos Eventos UPF.

Link: <http://eventos-test.upf.br/eventos-test>

- Abertura dos bugs:

Para abertura dos bugs, deve ser seguido o documento de instruções disponível no link abaixo. O documento deve conter as evidências dos testes realizados e a descrição de como deve ser reproduzido um bug.

Acesso: disponível na página do Moodle.

- Prints da tela:

Utilizar o “Ferramentas de captura” no Windows e o “Screenshot” no Linux.

- Gerador de CPF ou CNPJ

Para gerar um CPF ou CNPJ utilizar o link: <https://www.4devs.com.br>

Navegadores:

- Chrome
- Firefox
- Internet Explorer
- Safari

Dispositivos:

- Android
- IOS

Bugs aceitos:

- **Impeditivos:** impedem o uso de alguma funcionalidade do software. Não existem saídas ou alternativas para contorná-las.
- **Funcionais:** produzem um comportamento ilógico ou inesperado da aplicação onde o resultado obtido é diferente do esperado. A diferença entre uma falha funcional e uma falha impeditiva é que as falhas funcionais podem ser contornadas.
- **Texto:** qualquer ocorrência relacionada ao uso incorreto de algum idioma.
- **Interface:** são aquelas relacionadas à interface gráfica, à apresentação do software, exceto as falhas de texto. Exemplos: componentes desalinhados, renderização incorreta de interface, cores inconsistentes com guia de estilos, etc.
- **Melhorias:** não são propriamente falhas, mas sim sugestões para o aprimoramento do software. As sugestões são dadas com base na experiência do USUÁRIO e nos padrões de mercado.
- **Segurança:** são falhas que possibilitam acessar o sistema sem fazer uso de autenticação devida, sendo possível acessar funcionalidades, coletar dados, modificar dados, modificar o comportamento da aplicação ou interromper o funcionamento da aplicação.

Entregáveis:

- Lista de bugs abertos.

[Tarefa 6 – Rodada 1] Realização de teste na funcionalidade de “Login”

Situação: Ativa

Deadline: 06/11/2019 18:00

Descrição:

Esta tarefa é para realização dos testes na área de login da aplicação. A página apresenta inicialmente a área para informar o usuário, sendo realizada a primeira validação se o mesmo está cadastrado. Caso o usuário não possua cadastro, é apresentada mensagem informativa com opção de ir a página de realização de cadastro. Caso o usuário possua cadastro, é apresentado campo para informar a senha. É possível realizar o login informando o CPF, CNPJ ou ID estrangeiro.

Essa atividade consiste em encontrar bugs na área de “Login” da página de Eventos da UPF.

Instruções:

- Link dos testes:

Para realização dos testes, deve ser acessado o link abaixo com a URL do ambiente de homologação da página dos Eventos UPF.

Link: <http://eventos-test.upf.br/eventos-test>

- Abertura dos bugs:

Para abertura dos bugs, deve ser seguido o documento de instruções disponível no link abaixo. O documento deve conter as evidências dos testes realizados e a descrição de como deve ser reproduzido um bug.

Acesso: disponível na página do Moodle.

- Prints da tela:

Utilizar o “Ferramentas de captura” no Windows e o “Screenshot” no Linux.

- Gerador de CPF ou CNPJ

Para gerar um CPF ou CNPJ utilizar o link: <https://www.4devs.com.br>

Navegadores:

- Chrome
- Firefox
- Internet Explorer
- Safari

Dispositivos:

- Android
- IOS

Bugs aceitos:

- **Impeditivos:** impedem o uso de alguma funcionalidade do software. Não existem saídas ou alternativas para contorná-las.
- **Funcionais:** produzem um comportamento ilógico ou inesperado da aplicação onde o resultado obtido é diferente do esperado. A diferença entre uma falha funcional e uma falha impeditiva é que as falhas funcionais podem ser contornadas.
- **Texto:** qualquer ocorrência relacionada ao uso incorreto de algum idioma.
- **Interface:** são aquelas relacionadas à interface gráfica, à apresentação do software, exceto as falhas de texto. Exemplos: componentes desalinhados, renderização incorreta de interface, cores inconsistentes com guia de estilos, etc.
- **Melhorias:** não são propriamente falhas, mas sim sugestões para o aprimoramento do software. As sugestões são dadas com base na experiência do USUÁRIO e nos padrões de mercado.
- **Segurança:** são falhas que possibilitam acessar o sistema sem fazer uso de autenticação devida, sendo possível acessar funcionalidades, coletar dados, modificar dados, modificar o comportamento da aplicação ou interromper o funcionamento da aplicação.

Entregáveis:

- Lista de bugs abertos.

[Tarefa 7 – Rodada 1] Realização de teste na funcionalidade de “Criar conta”

Situação: Ativa

Deadline: 06/11/2019 18:00

Descrição:

Esta tarefa é para realização dos testes na área de login da aplicação. A página apresenta informações de nome, tipo de pessoa e-mail, senha e confirmação de senha. Ao selecionar o tipo de pessoa, aplicação apresenta o CPF, CNPJ ou ID estrangeiro, conforme a seleção. Todas as informações são obrigatórias para realização do cadastro.

Essa atividade consiste em encontrar bugs na área de “Criar Conta” da página de Eventos da UPF.

Essa funcionalidade é acessada ao clicar em “Não possuo conta” na página de login.

Instruções:

- Link dos testes:

Para realização dos testes, deve ser acessado o link abaixo com a URL do ambiente de homologação da página dos Eventos UPF.

Link: <http://eventos-test.upf.br/eventos-test>

- Abertura dos bugs:

Para abertura dos bugs, deve ser seguido o documento de instruções disponível no link abaixo. O documento deve conter as evidências dos testes realizados e a descrição de como deve ser reproduzido um bug.

Acesso: disponível na página do Moodle.

- Prints da tela:

Utilizar o “Ferramentas de captura” no Windows e o “Screenshot” no Linux.

- Gerador de CPF ou CNPJ

Para gerar um CPF ou CNPJ utilizar o link: <https://www.4devs.com.br>

Navegadores:

- Chrome
- Firefox
- Internet Explorer
- Safari

Dispositivos:

- Android
- IOS

Bugs aceitos:

- **Impeditivos:** impedem o uso de alguma funcionalidade do software. Não existem saídas ou alternativas para contorná-las.
- **Funcionais:** produzem um comportamento ilógico ou inesperado da aplicação onde o resultado obtido é diferente do esperado. A diferença entre uma falha funcional e uma falha impeditiva é que as falhas funcionais podem ser contornadas.
- **Texto:** qualquer ocorrência relacionada ao uso incorreto de algum idioma.
- **Interface:** são aquelas relacionadas à interface gráfica, à apresentação do software, exceto as falhas de texto. Exemplos: componentes desalinhados, renderização incorreta de interface, cores inconsistentes com guia de estilos, etc.
- **Melhorias:** não são propriamente falhas, mas sim sugestões para o aprimoramento do software. As sugestões são dadas com base na experiência do USUÁRIO e nos padrões de mercado.
- **Segurança:** são falhas que possibilitam acessar o sistema sem fazer uso de autenticação devida, sendo possível acessar funcionalidades, coletar dados, modificar dados, modificar o comportamento da aplicação ou interromper o funcionamento da aplicação.

Entregáveis:

- Lista de bugs abertos.

APÊNDICE F – TAREFAS DO EXPERIMENTO - RODADA 2

[Tarefa 1 – Rodada 2] Realização de teste na funcionalidade de “Tipos de ambiente”

Situação: Ativa

Deadline: 13/11/2019 18:00

Descrição:

Esta tarefa é para realização dos testes da área do site onde é possível realizar o cadastro de Tipos de Ambientes disponíveis na Universidade. Essa área apresenta na parte superior da página itens para filtro dos ambientes já cadastrados. Na área abaixo são apresentados os ambientes já cadastrados, sendo apresenta a opção para editar e excluir o registro cadastrado. Logo abaixo está disponível a opção para realizar o cadastro de um novo ambiente. Ao ser criado um novo registro, aplicação apresenta o item cadastrado juntamente com os demais.

Essa atividade consiste em encontrar bugs na página de “Tipos de Ambientes” localizada na área administrativa da página de Eventos da UPF.

Instruções:

- Link dos testes:

Para realização dos testes, deve ser acessado o link abaixo com a URL do ambiente de homologação da área administrativa da página dos Eventos UPF.

Link: <http://eventosadm-test.upf.br/eventosadm-test/tipos-ambiente>

Login: testes@testes.com

Senha: 123456f

- Abertura dos bugs:

Para abertura dos bugs, deve ser seguido o documento de instruções disponível no Moodle. O documento deve conter as evidências dos testes realizados e a descrição de como deve ser reproduzido um bug.

- Prints da tela:

Utilizar o “Ferramentas de captura” no Windows e o “Screenshot” no Linux.

- Gerador de CPF ou CNPJ

Para gerar um CPF ou CNPJ utilizar o link: <https://www.4devs.com.br>

Navegadores:

- Chrome
- Firefox
- Internet Explorer
- Safari

Dispositivos:

- Android
- IOS

Bugs aceitos:

- **Impeditivos:** impedem o uso de alguma funcionalidade do software. Não existem saídas ou alternativas para contorná-las.
- **Funcionais:** produzem um comportamento ilógico ou inesperado da aplicação onde o resultado obtido é diferente do esperado. A diferença entre uma falha funcional e uma falha impeditiva é que as falhas funcionais podem ser contornadas.
- **Texto:** qualquer ocorrência relacionada ao uso incorreto de algum idioma.
- **Interface:** são aquelas relacionadas à interface gráfica, à apresentação do software, exceto as falhas de texto. Exemplos: componentes desalinhados, renderização incorreta de interface, cores inconsistentes com guia de estilos, etc.
- **Melhorias:** não são propriamente falhas, mas sim sugestões para o aprimoramento do software. As sugestões são dadas com base na experiência do USUÁRIO e nos padrões de mercado.
- **Segurança:** são falhas que possibilitam acessar o sistema sem fazer uso de autenticação devida, sendo possível acessar funcionalidades, coletar dados, modificar dados, modificar o comportamento da aplicação ou interromper o funcionamento da aplicação.

Entregáveis:

- Lista de bugs abertos.
- Decomposição da tarefa.

[Tarefa 2 – Rodada 2] Realização de teste na funcionalidade de “Validar Resumos”

Situação: Ativa

Deadline: 13/11/2019 18:00

Descrição:

Esta tarefa é para realização dos testes da área do site onde são realizadas as validações dos resumos de artigos enviados aos eventos. A página apresenta área para realização de filtros, além de opções para download das informações.

Essa atividade consiste em encontrar bugs na página de “Validar resumos” localizada na área administrativa da página de Eventos da UPF.

Instruções:

- Link dos testes:

Para realização dos testes, deve ser acessado o link abaixo com a URL do ambiente de homologação da área administrativa da página dos Eventos UPF.

Link: <http://eventosadm-test.upf.br/eventosadm-test/atividades/valida-resumo>

Login: testes@testes.com

Senha: 123456f

- Abertura dos bugs:

Para abertura dos bugs, deve ser seguido o documento de instruções disponível no Moodle. O documento deve conter as evidências dos testes realizados e a descrição de como deve ser reproduzido um bug.

- Prints da tela:

Utilizar o “Ferramentas de captura” no Windows e o “Screenshot” no Linux.

- Gerador de CPF ou CNPJ

Para gerar um CPF ou CNPJ utilizar o link: <https://www.4devs.com.br>

Navegadores:

- Chrome
- Firefox
- Internet Explorer
- Safari

Dispositivos:

- Android
- IOS

Bugs aceitos:

- **Impeditivos:** impedem o uso de alguma funcionalidade do software. Não existem saídas ou alternativas para contorná-las.
- **Funcionais:** produzem um comportamento ilógico ou inesperado da aplicação onde o resultado obtido é diferente do esperado. A diferença entre uma falha funcional e uma falha impeditiva é que as falhas funcionais podem ser contornadas.
- **Texto:** qualquer ocorrência relacionada ao uso incorreto de algum idioma.
- **Interface:** são aquelas relacionadas à interface gráfica, à apresentação do software, exceto as falhas de texto. Exemplos: componentes desalinhados, renderização incorreta de interface, cores inconsistentes com guia de estilos, etc.
- **Melhorias:** não são propriamente falhas, mas sim sugestões para o aprimoramento do software. As sugestões são dadas com base na experiência do USUÁRIO e nos padrões de mercado.
- **Segurança:** são falhas que possibilitam acessar o sistema sem fazer uso de autenticação devida, sendo possível acessar funcionalidades, coletar dados, modificar dados, modificar o comportamento da aplicação ou interromper o funcionamento da aplicação.

Entregáveis:

- Lista de bugs abertos.
- Decomposição da tarefa.

[Tarefa 3 – Rodada 2] Realização de teste na funcionalidade de “Validar Questionários”

Situação: Ativa

Deadline: 13/11/2019 18:00

Descrição:

Esta tarefa é para realização dos testes da área do site onde são realizadas as validações dos resumos de artigos enviados aos eventos. A página apresenta área para realização de filtros, além de opções para download das informações.

Essa atividade consiste em encontrar bugs na página de “Validar questionários” localizada na área administrativa da página de Eventos da UPF.

Instruções:

- Link dos testes:

Para realização dos testes, deve ser acessado o link abaixo com a URL do ambiente de homologação da área administrativa da página dos Eventos UPF.

Link: <http://eventosadm-test.upf.br/eventosadm-test/atividades/pos/valida-questionario>

Login: testes@testes.com

Senha: 123456f

- Abertura dos bugs:

Para abertura dos bugs, deve ser seguido o documento de instruções disponível no Moodle. O documento deve conter as evidências dos testes realizados e a descrição de como deve ser reproduzido um bug.

- Prints da tela:

Utilizar o “Ferramentas de captura” no Windows e o “Screenshot” no Linux.

- Gerador de CPF ou CNPJ

Para gerar um CPF ou CNPJ utilizar o link: <https://www.4devs.com.br>

Navegadores:

- Chrome
- Firefox
- Internet Explorer
- Safari

Dispositivos:

- Android
- IOS

Bugs aceitos:

- **Impeditivos:** impedem o uso de alguma funcionalidade do software. Não existem saídas ou alternativas para contorná-las.
- **Funcionais:** produzem um comportamento ilógico ou inesperado da aplicação onde o resultado obtido é diferente do esperado. A diferença entre uma falha funcional e uma falha impeditiva é que as falhas funcionais podem ser contornadas.
- **Texto:** qualquer ocorrência relacionada ao uso incorreto de algum idioma.
- **Interface:** são aquelas relacionadas à interface gráfica, à apresentação do software, exceto as falhas de texto. Exemplos: componentes desalinhados, renderização incorreta de interface, cores inconsistentes com guia de estilos, etc.
- **Melhorias:** não são propriamente falhas, mas sim sugestões para o aprimoramento do software. As sugestões são dadas com base na experiência do USUÁRIO e nos padrões de mercado.
- **Segurança:** são falhas que possibilitam acessar o sistema sem fazer uso de autenticação devida, sendo possível acessar funcionalidades, coletar dados, modificar dados, modificar o comportamento da aplicação ou interromper o funcionamento da aplicação.

Entregáveis:

- Lista de bugs abertos.
- Decomposição da tarefa.

[Tarefa 4 – Rodada 2] Realização de teste na funcionalidade de “Validar Arquivos Enviados”

Situação: Ativa

Deadline: 13/11/2019 18:00

Descrição:

Esta tarefa é para realização dos testes da área do site onde validados os arquivos enviados para um determinado evento. A página apresenta área para realização de filtros para consulta dos documentos que passam pelos avaliadores.

Essa atividade consiste em encontrar bugs na página de “Validar arquivos enviados” localizada na área administrativa da página de Eventos da UPF.

Instruções:

- Link dos testes:

Para realização dos testes, deve ser acessado o link abaixo com a URL do ambiente de homologação da área administrativa da página dos Eventos UPF.

Link: <http://eventosadm-test.upf.br/eventosadm-test/atividades/pos/validar-arquivos>

Login: testes@testes.com

Senha: 123456f

- Abertura dos bugs:

Para abertura dos bugs, deve ser seguido o documento de instruções disponível no Moodle. O documento deve conter as evidências dos testes realizados e a descrição de como deve ser reproduzido um bug.

- Prints da tela:

Utilizar o “Ferramentas de captura” no Windows e o “Screenshot” no Linux.

- Gerador de CPF ou CNPJ

Para gerar um CPF ou CNPJ utilizar o link: <https://www.4devs.com.br>

Navegadores:

- Chrome
- Firefox
- Internet Explorer
- Safari

Dispositivos:

- Android
- IOS

Bugs aceitos:

- **Impeditivos:** impedem o uso de alguma funcionalidade do software. Não existem saídas ou alternativas para contorná-las.
- **Funcionais:** produzem um comportamento ilógico ou inesperado da aplicação onde o resultado obtido é diferente do esperado. A diferença entre uma falha funcional e uma falha impeditiva é que as falhas funcionais podem ser contornadas.
- **Texto:** qualquer ocorrência relacionada ao uso incorreto de algum idioma.
- **Interface:** são aquelas relacionadas à interface gráfica, à apresentação do software, exceto as falhas de texto. Exemplos: componentes desalinhados, renderização incorreta de interface, cores inconsistentes com guia de estilos, etc.
- **Melhorias:** não são propriamente falhas, mas sim sugestões para o aprimoramento do software. As sugestões são dadas com base na experiência do USUÁRIO e nos padrões de mercado.
- **Segurança:** são falhas que possibilitam acessar o sistema sem fazer uso de autenticação devida, sendo possível acessar funcionalidades, coletar dados, modificar dados, modificar o comportamento da aplicação ou interromper o funcionamento da aplicação.

Entregáveis:

- Lista de bugs abertos.
- Decomposição da tarefa.

[Tarefa 5 – Rodada 2] Realização de teste na funcionalidade de “Participantes Convidados”

Situação: Ativa

Deadline: 13/11/2019 18:00

Descrição:

Esta tarefa é para realização dos testes da área do site onde são cadastrados os participantes para participação nos eventos. A página apresenta área para realização de filtros para consulta dos participantes e opções para download. Ainda na página é possível realizar o cadastro de um novo participante.

Essa atividade consiste em encontrar bugs na página de “Participantes Convidados” localizada na área administrativa da página de Eventos da UPF.

Instruções:

- Link dos testes:

Para realização dos testes, deve ser acessado o link abaixo com a URL do ambiente de homologação da área administrativa da página dos Eventos UPF.

Link: <http://eventosadm-test.upf.br/eventosadm-test/participanteconvidado>

Login: testes@testes.com

Senha: 123456f

- Abertura dos bugs:

Para abertura dos bugs, deve ser seguido o documento de instruções disponível no Moodle. O documento deve conter as evidências dos testes realizados e a descrição de como deve ser reproduzido um bug.

- Prints da tela:

Utilizar o “Ferramentas de captura” no Windows e o “Screenshot” no Linux.

- Gerador de CPF ou CNPJ

Para gerar um CPF ou CNPJ utilizar o link: <https://www.4devs.com.br>

Navegadores:

- Chrome
- Firefox
- Internet Explorer
- Safari

Dispositivos:

- Android
- IOS

Bugs aceitos:

- **Impeditivos:** impedem o uso de alguma funcionalidade do software. Não existem saídas ou alternativas para contorná-las.
- **Funcionais:** produzem um comportamento ilógico ou inesperado da aplicação onde o resultado obtido é diferente do esperado. A diferença entre uma falha funcional e uma falha impeditiva é que as falhas funcionais podem ser contornadas.
- **Texto:** qualquer ocorrência relacionada ao uso incorreto de algum idioma.
- **Interface:** são aquelas relacionadas à interface gráfica, à apresentação do software, exceto as falhas de texto. Exemplos: componentes desalinhados, renderização incorreta de interface, cores inconsistentes com guia de estilos, etc.
- **Melhorias:** não são propriamente falhas, mas sim sugestões para o aprimoramento do software. As sugestões são dadas com base na experiência do USUÁRIO e nos padrões de mercado.
- **Segurança:** são falhas que possibilitam acessar o sistema sem fazer uso de autenticação devida, sendo possível acessar funcionalidades, coletar dados, modificar dados, modificar o comportamento da aplicação ou interromper o funcionamento da aplicação.

Entregáveis:

- Lista de bugs abertos.
- Decomposição da tarefa.

APÊNDICE G – GUIA PARA DECOMPOSIÇÃO DE MACROTASKS

UNIVERSIDADE DE PASSO FUNDO

Programa de Pós-Graduação em
Computação Aplicada

GUIA PARA DECOMPOSIÇÃO DE MACROTASKS



SUMÁRIO

1.	PASSO 1 – ANÁLISE DA ESTRUTURA DA TAREFA	2
2.	PASSO 2 – VALIDAÇÃO E IDENTIFICAÇÃO INDIVIDUAL DOS ITENS DA ESTRUTURA	3
3.	PASSO 3 – PASSOS PARA REALIZAÇÃO DA DECOMPOSIÇÃO	5
3.1.	DECIDIR O MOTIVO DA ANÁLISE.....	5
3.2.	COLETAR OS DADOS	5
3.3.	DEFINIR O OBJETIVO GERAL DA TAREFA	5
3.4.	DIVIDIR OS OBJETIVOS GERAIS EM SUB OBJETIVOS E OPERAÇÕES.....	6
3.5.	IDENTIFICAÇÃO DAS OPERAÇÕES PARA CUMPRIMENTO DOS OBJETIVOS	6
3.6.	EXEMPLO DE ESTRUTURA APÓS DECOMPOSIÇÃO	6
4.	EXECUÇÃO E ACOMPANHAMENTO	8
4.1.	EXECUÇÃO DOS CENÁRIOS MAPEADOS.....	8
4.2.	REPORTAR INCIDENTES	8
4.3.	ACOMPANHAMENTO DE BUGS.....	9

1. PASSO 1 – ANÁLISE DA ESTRUTURA DA TAREFA

No primeiro passo serão analisados os itens que compõem a estrutura de uma tarefa a ser testada. A estrutura básica estudada é composta por: situação, deadline, instruções, requisitos, objetivos, ajuda e pagamento. Na Figura 1 são exemplificados e indicados os itens da estrutura da tarefa.

The screenshot shows the 'Projeto Fogás' interface. At the top, the project name 'Projeto Fogás' is followed by a yellow 'PAUSADA' (PAUSED) status and a red box labeled 'Situação' with an arrow pointing to it. Below the header, there are tabs for 'Detalhes', 'Ranking', and 'Arquivos'. The main content area features the 'FOGÁS' logo with a 'MOBILE' tag. To the right, there is a section for 'Inscrições encerradas' (Registrations closed) with a 'Data de encerramento' (Closing date) of '15/02/2019 00:00 encerrada'. Below this, there are 'Outras informações' (Other information) and a 'Release criada em 25/01/2019 10:24'. The main text area contains instructions about the app, its availability in certain regions, and rules for testers. At the bottom, there is a section for 'Instruções' (Instructions) with a red box labeled 'Instruções' and an arrow pointing to it. The instructions text includes: 'Para INSTALAR O APLICATIVO, no seu aparelho ANDROID, vá em **Configurações >> Segurança >> Fontes desconhecidas** e permita a instalação de aplicativos de fontes desconhecidas. (Isso se faz necessário, porque a versão de testes está em homologação e não disponível na Play Store).'

Figura 1: Exemplo de identificação da estrutura da tarefa
Fonte: Elaborada pelo autor

2. PASSO 2 – VALIDAÇÃO E IDENTIFICAÇÃO INDIVIDUAL DOS ITENS DA ESTRUTURA

Nesse item, a estrutura básica da tarefa deve ser identificada, com base na análise realizada anteriormente. Após a identificação, validar isoladamente os itens identificados conforme Figura 2, sendo que os quatro primeiros itens são impeditivos para a sequência na tarefa escolhida na plataforma.

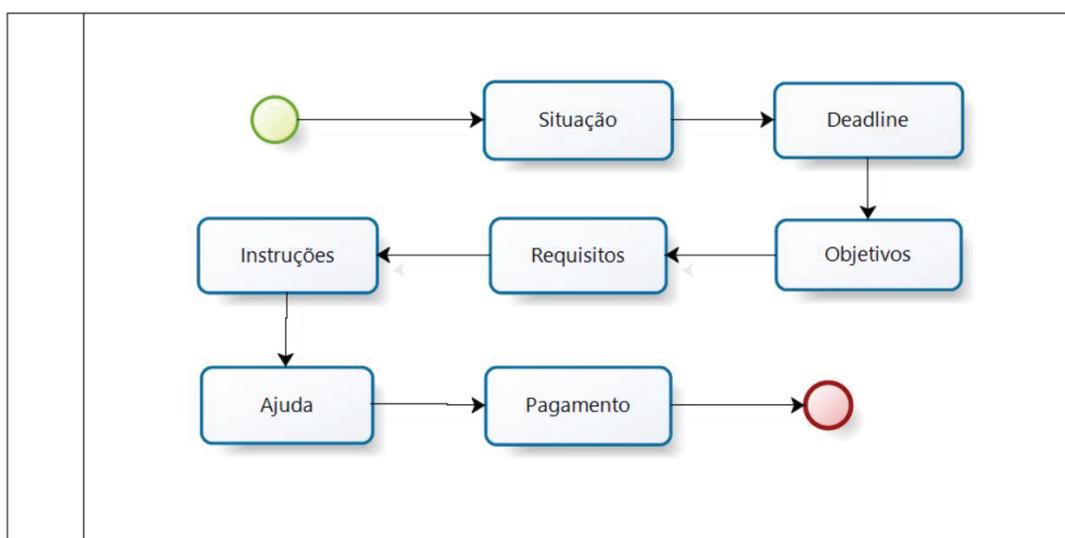


Figura 2: fluxo da estrutura das tarefas
Fonte: elaborada pelo autor

A *situação* corresponde ao estado atual da tarefa, podendo estar ativa ou inativa. Caso a tarefa esteja ativa significa que está disponível para realização dos testes, porém caso a mesma esteja inativa, suas atividades foram interrompidas e impossibilitam a realização dos testes.

A *deadline* identifica o período para realização dos testes. Esse item pode ser determinante pois o testador poderá não ter tempo suficiente para realização da tarefa. Outro ponto importante é que quanto mais próxima a data final, mais bugs estarão abertos, diminuindo a possibilidade de relatar um problema.

As *instruções* correspondem as orientações a serem seguidas na realização da tarefa. Podem ser apresentadas como pré-requisito para realização dos testes ou como orientação durante o mesmo. Uma instrução que não é seguida pode acarretar na não realização do teste ou na reprovação do bug relatado.

Os **requisitos** são um conjunto de itens a serem completados para que a tarefa seja iniciada e possa ser concluída. Caso algum dos itens não for atendido, é impeditivo para iniciar ou finalizar corretamente uma tarefa.

Os **objetivos** são as finalidades as quais a tarefa é proposta, devendo ser observado na resolução da tarefa pois indica itens importantes para sua resolução.

Para auxiliar o testador, tópicos de **ajuda** são apresentados nas tarefas. Deve ser observado pois contém guias e determinadas dúvidas frequentes entre outros testadores da plataforma.

O **pagamento** é apresentado indicando a remuneração que o testador receberá caso obtenha êxito na tarefa. Pode ser observado para orientar o testador aos tipos de bugs melhor remunerados na resolução da tarefa.

3. PASSO 3 – PASSOS PARA REALIZAÇÃO DA DECOMPOSIÇÃO

Abaixo são apresentados os passos para realização da decomposição de tarefas em crowdsourcing.

3.1. DECIDIR O MOTIVO DA ANÁLISE

Nesse item deve ser delimitado qual o objetivo da análise que será realizada, definindo a tarefa sob investigação e identificando o propósito dessa tarefa.

3.2. COLETAR OS DADOS

Para aplicação da técnica, é necessário a obtenção dos dados de como a tarefa é executada. Para isso deve-se identificar onde o teste deverá ser realizado e quais os tipos de bugs aceitos.

Por exemplo: para determinada tarefa está descrito que os testes deverão ser aplicados no mobile e não serão aceitos bugs com erros de ortografia apresentados na página. Com isso o testador deve ter atenção para não relatar esse tipo de bug durante os testes.

3.3. DEFINIR O OBJETIVO GERAL DA TAREFA

A definição do objetivo deve ser realizada, sendo apresentado como o nível superior da hierarquia. Nesse passo também deverão ser identificadas as funcionalidades que deverão ser testadas, definindo o escopo dos testes e os apresentando no próximo nível da hierarquia. Além do escopo, deverão ser definidos os itens que não fazem parte do escopo dos testes.

Por exemplo: o objetivo da tarefa é a realização dos testes no menu de simulação de renegociação da fatura do cartão de crédito, tendo como escopo a utilização desse menu. Os itens definidos como fora de escopo são os que não são acessados através deste menu. Com isso o testador deverá ignorar qualquer bug encontrado nas funcionalidades não definidas no escopo do teste.

3.4. DIVIDIR OS OBJETIVOS GERAIS EM SUB OBJETIVOS E OPERAÇÕES

Nesse passo, devem ser mapeados os sub objetivos e operações para cada objetivo e dessa forma realizar o mapeamento dos cenários de teste. O objetivo é realizar a identificação, elaboração e documentação dos cenários em alto nível.

A estrutura da árvore seguirá sendo montada tendo os sub objetivos um nível abaixo dos objetivos e as operações um nível abaixo dos sub objetivos.

Como exemplo pode ser apresentado para o objetivo “Realização de testes na tela de login”, um dos sub objetivos mapeados para testes poderia ser “Validar a impossibilidade de realização do login com dados inválidos” ou “Validar a impossibilidade de login com dados não cadastrados”. Quanto as operações para o sub objetivo “Validar a impossibilidade de realização do login com dados inválidos” poderia ser indicada como “informar os dados inválidos nos campos” e após “clicar no botão para realização do login”.

3.5. IDENTIFICAÇÃO DAS OPERAÇÕES PARA CUMPRIMENTO DOS OBJETIVOS

Nesse item deverão ser identificados os sistemas operacionais, versões e dispositivos necessários para realização dos testes. Esse item deve ser observado pois a realização dos testes em um dispositivo ou sistema incorreto acarreta em rejeição do bug aberto. Por exemplo: a determinação da plataforma é a realização do teste em computadores com sistema operacional Windows, no navegador Chrome e com sua versão superior a 35.

3.6. EXEMPLO DE ESTRUTURA APÓS DECOMPOSIÇÃO

Ao final da aplicação da técnica, os itens da tarefa ficarão estruturados em árvore, conforme apresentado na Figura 3.

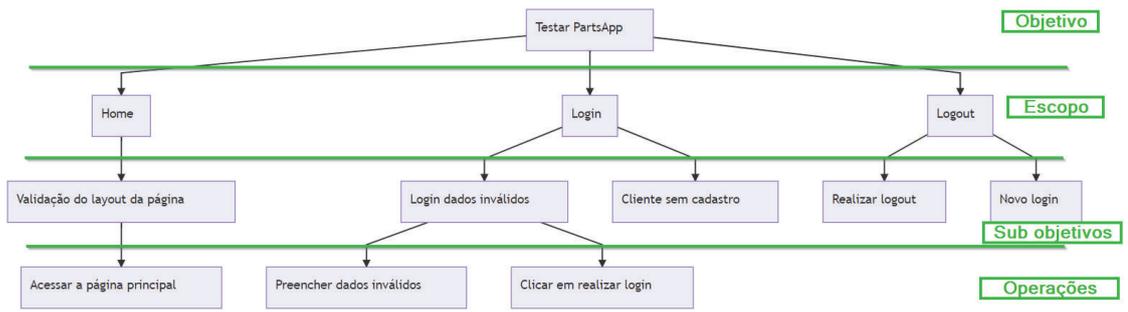


Figura 3: Estrutura após decomposição
Fonte: Elaborada pelo autor

4. EXECUÇÃO E ACOMPANHAMENTO

Nesse item serão descritos os elementos para realização da execução e acompanhamento dos cenários mapeados.

4.1. EXECUÇÃO DOS CENÁRIOS MAPEADOS

Nesse item devem ser executados os casos de teste identificados. Devem ser identificadas as inconsistências, baseando-se nos cenários mapeados antecipadamente, devendo ser observadas as saídas apresentadas pelo sistema.

4.2. REPORTAR INCIDENTES

Os cenários identificados devem ser reportados, sendo que para cada inconsistência, deve ser criado um registro de um bug. Devem ser evidenciados os bugs encontrados conforme instruções da plataforma. Caso não contenham instruções, levar em consideração os itens básicos para reportar um incidente. Segue abaixo estrutura básica com as informações para reportar um bug:

- Identificador do incidente
- Sumário
- Descrição do incidente
 - Entradas
 - Resultados esperados
 - Resultados atuais
 - Anomalias
 - Data e hora
 - Passo do procedimento
 - Ambiente
 - Tentativas de repetição
 - Testador
 - Observadores
- Impacto

Um ponto importante a ser observado antes do report do bug, o testador deve analisar os bugs já abertos e não reportar um problema já existente.

4.3. ACOMPANHAMENTO DE BUGS

Nesse item deve ser realizado o acompanhamento dos bugs enviados para verificar se os mesmos serão aceitos ou rejeitados.



UPF

UNIVERSIDADE
DE PASSO FUNDO

UPF Campus I - BR 285, São José
Passo Fundo - RS - CEP: 99052-900
(54) 3316 7000 - www.upf.br