

UNIVERSIDADE FEDERAL DO PAMPA

MICHEL SAUCEDA MARQUES

**UMA ABORDAGEM SOCIALMENTE RESPONSÁVEL AO DESENVOLVIMENTO
DE SOFTWARE**

**Alegrete
2022**

MICHEL SAUCEDA MARQUES

**UMA ABORDAGEM SOCIALMENTE RESPONSÁVEL AO DESENVOLVIMENTO
DE SOFTWARE**

Trabalho de Conclusão de Curso apresentado ao Curso de Engenharia de Software da Universidade Federal do Pampa, como requisito parcial para obtenção do Título de Bacharel em Engenharia de Software.

Orientadora: Amanda Meincke Melo

**Alegrete
2022**

Ficha catalográfica elaborada automaticamente com os dados fornecidos
pelo(a) autor(a) através do Módulo de Biblioteca do
Sistema GURI (Gestão Unificada de Recursos Institucionais) .

M357a Marques, Michel Saucedo
UMA ABORDAGEM SOCIALMENTE RESPONSÁVEL AO DESENVOLVIMENTO DE
SOFTWARE / Michel Saucedo Marques.
116 p.

Trabalho de Conclusão de Curso(Graduação)-- Universidade
Federal do Pampa, ENGENHARIA DE SOFTWARE, 2022.
"Orientação: Amanda Meincke Melo".

1. Engenharia de Software. 2. Ética. 3. Desenvolvimento
Socialmente Responsável de Software. 4. Interação Humano-
Computador. I. Título.

MICHEL SAUCEDA MARQUES

Uma Abordagem Socialmente Responsável ao Desenvolvimento de Software

Trabalho de Conclusão apresentado ao curso de Engenharia de Software da Universidade Federal do Pampa, como requisito parcial para obtenção do Título de Bacharel em Engenharia de Software.

TCC defendido e aprovado em: 3/8/22.

Banca examinadora:

Prof.^a Dra.^a Amanda Meincke Melo

Orientadora

UNIPAMPA

Prof.^a Dr.^a Aline Vieira de Mello

UNIPAMPA

Prof. Dr. João Pablo Silva da Silva

UNIPAMPA



Assinado eletronicamente por **AMANDA MEINCKE MELO, PROFESSOR DO MAGISTERIO SUPERIOR**, em 03/08/2022, às 21:44, conforme horário oficial de Brasília, de acordo com as normativas legais aplicáveis.



Assinado eletronicamente por **JOAO PABLO SILVA DA SILVA, PROFESSOR DO MAGISTERIO SUPERIOR**, em 03/08/2022, às 21:44, conforme horário oficial de Brasília, de acordo com as normativas legais aplicáveis.



Assinado eletronicamente por **ALINE VIEIRA DE MELLO, PROFESSOR DO MAGISTERIO SUPERIOR**, em 03/08/2022, às 21:46, conforme horário oficial de Brasília, de acordo com as normativas legais aplicáveis.



A autenticidade deste documento pode ser conferida no site https://sei.unipampa.edu.br/sei/controlador_externo.php?acao=documento_conferir&id_orgao_acesso_externo=0, informando o código verificador **0876723** e o código CRC **9C500494**.

RESUMO

Temos uma sociedade na qual a tecnologia digital está cada vez mais imersa na vida das pessoas, trazendo impactos ao seu cotidiano. Com isso, é indispensável que os softwares desenvolvidos tenham uma preocupação com os aspectos éticos, tanto por causa das leis de proteção de dados dos cidadãos e regulamentações de softwares autônomos, quanto pela importância de produzir software de qualidade em que as necessidades reais dos usuários sejam atendidas. Nos trabalhos relacionados, identificou-se a preocupação com esses aspectos, mas em apenas dois trabalhos, processos, métodos e ferramentas são explicitamente abordados. Ademais, nota-se também a importância das pessoas nas tomadas de decisões éticas e o seu impacto para que os processos e métodos sejam socialmente responsáveis. Este trabalho portanto, tem como objetivo propor uma abordagem socialmente responsável ao desenvolvimento de *software*, no qual o software produzido tenha impacto positivo na sociedade e que o seu desenvolvimento considere aspectos éticos como um fator de qualidade, contemplados nos processos, nos métodos e nas ferramentas adotados. Além disso, que as pessoas integrem o processo de tomada de decisão ética durante todo o desenvolvimento. Para buscar esse objetivo, tem-se a condução de uma abordagem qualitativa de pesquisa, integrando aspectos éticos com a Engenharia de Software em três etapas: Organização de Pesquisa, Engenharia de Requisitos e Desenvolvimento Incremental. Os resultados obtidos são uma abordagem de desenvolvimento de software que integra aspectos éticos à Engenharia de Software e aplicativo móvel ComuniApp, que possibilita a publicação de itens para doação e de avisos para a comunidade, a consulta em tempo real da posição de ônibus no mapa da cidade e informações de instituições públicas locais.

Palavras-Chave: Engenharia de Software, Ética, Desenvolvimento Socialmente Responsável de Software, Interação Humano-Computador

ABSTRACT

We have a society in which digital technology is increasingly immersed in people's lives, impacting their daily lives. With this, it is essential that the software developed has a concern with ethical aspects, both because of the laws for the protection of citizens' data and autonomous software regulations, as well as the importance of producing quality software in which the real needs of users are met. In related works, the concern with these aspects was identified, but in only two quality processes, methods and tools are explicitly addressed. Furthermore, it is also noted the importance of people in ethical decision-making and their impact so that processes and methods are socially responsible. This work, therefore, aims to propose a socially responsible approach to software development, in which the software produced has a positive impact on society and that its development considers ethical aspects as a quality factor, contemplated in the processes, methods and tools. adopted. Also, that people integrate the ethical decision-making process throughout development. To achieve this objective, a qualitative research approach is carried out, integrating ethical aspects with Software Engineering in three stages: Research Organization, Requirements Engineering and Incremental Development. The results obtained are a software development approach that integrates ethical aspects to Software Engineering and the ComuniApp mobile application, which allows the publication of items for donation and notices for the community, the real-time consultation of the bus position on the map of the city and information from local public institutions.

Keywords: Software Engineering, Ethics, Socially Responsible Software Development, Human-Computer Interaction

LISTA DE FIGURAS

Figura 1 – Camadas da Engenharia de Software.....	28
Figura 2 – Tipos de requisitos não funcionais.....	29
Figura 3 – Macroprocesso metodologia.....	32
Figura 4 – Organização da pesquisa.....	32
Figura 5 – Processo de Engenharia de Requisitos.....	34
Figura 6 – Aplicações desenvolvidas.....	37
Figura 7 – Processo de Desenvolvimento Incremental.....	38
Figura 8 – Ciclo de Desenvolvimento.....	39
Figura 9 – Diagrama de partes interessadas.....	45
Figura 10 – Protótipo em papel.....	47
Figura 11 – Protótipo em papel 2.....	48
Figura 12 – Mock-up 1 tela início.....	49
Figura 13 – Mock-up 2.....	50
Figura 14 – <i>Splash Screen</i>	52
Figura 15 – Tela de autenticação.....	53
Figura 16 – Tela Início.....	54
Figura 17 – Tela Ônibus.....	55
Figura 18 – Tela Doações.....	56
Figura 19 – Tela Adicionar Item.....	57
Figura 20 – Mensagem de confirmação.....	58
Figura 21 – Tela Adicionar Aviso.....	59
Figura 22 – Tela Instituições.....	60
Figura 23 – Tela de informações das instituições.....	61
Figura 24 – Tela Perfil	62
Figura 25 – Termo de Uso.....	63
Figura 26 – Políticas de Privacidade.....	64

LISTA DE QUADROS

Quadro 1 – Strings Aplicadas às bases de pesquisa	14
Quadro 2 – Relação de trabalhos identificados com as camadas de Engenharia de Software	17
Quadro 3 – Participação e papel das pessoas.....	22
Quadro 4 – Camadas de Engenharia de Software neste trabalho.....	28
Quadro 5 – Tecnologias e ferramentas utilizadas.....	38
Quadro 6 – Cronograma de participação da turma.....	42
Quadro 7 – Funcionalidades.....	43
Quadro 8 – Requisitos éticos.....	44
Quadro 9 – Quadro de Avaliação.....	45
Quadro 10 – Matriz de Urgência/Importância.....	51

LISTA DE SIGLAS

ACM – *Association for Computing Machinery*

DP – *Design Participativo*

EEEM – *Escola Estadual de Ensino Médio*

GGDB – *GGNetworks Database*

HCAI – *Human-Centered Artificial Intelligence*

IDE – *Inovação em Domínio Educacional*

IEC – *International Electrotechnical Commission*

IEEE – *Institute of Electrical and Electronic Engineers*

ISO – *International Organization for Standardization*

LGPD – *Lei Geral de Proteção de Dados*

OSp – *Oficina Semio-participativa*

RE – *Requisito Ético*

RF – *Requisito Funcionalidade*

RNF – *Requisito Não-Funcional*

SWEBOK – *Software Engineering Body of Knowledge*

TCC – *Trabalho de Conclusão de Curso*

TCLE – *Termo de Consentimento Livre e Esclarecido*

UML – *Unified Modeling Language*

SUMÁRIO

1 INTRODUÇÃO.....	12
2 ESTADO DA ARTE.....	14
2.1 Questão de Pesquisa.....	14
2.2 Strings e Bases de Busca.....	14
2.3 Critérios de Inclusão e Exclusão.....	15
2.4 Resultados e Análises dos Trabalhos Relacionados.....	16
2.5 Considerações Finais do Capítulo.....	25
3 METODOLOGIA.....	27
3.1 Camadas de Engenharia de Software neste Trabalho.....	28
3.2 Etapas para o desenvolvimento do Trabalho.....	31
3.3 Considerações finais do capítulo.....	40
4 RESULTADOS E DISCUSSÕES.....	42
4.1 Organização da Pesquisa.....	42
4.2 Engenharia de Requisitos.....	43
4.3 Desenvolvimento Incremental.....	51
4.4 Considerações finais do capítulo.....	65
5 CONSIDERAÇÕES FINAIS	66
REFERÊNCIAS.....	69
APÊNDICES.....	73

1 INTRODUÇÃO

O uso de software é parte da rotina diária de inúmeras pessoas no desempenho de uma série de tarefas, desde as mais simples – ex.: despertar, anotar, conversar – até as mais complexas – ex.: comprar, realizar procedimentos médicos. Software afetam, assim, muitas vidas (DEKHTYAR *et al.*, 2020).

De acordo com o Código de Ética e Prática Profissional de Engenharia de Software ACM/IEEE (ACM/IEEE, 2015), os profissionais da Engenharia de Software (ES) têm um compromisso com a sociedade ao desenvolver aplicações. Estas têm impacto direto sobre os usuários e esse impacto muitas vezes é difícil de mensurar, de forma que os engenheiros de *software* têm muitas oportunidades de fazer o bem ou causar danos. Portanto, o profissional de Engenharia de Software deve agir de forma ética, ou seja, colocando o bem público em primeiro lugar e considerando sua relação com pessoas e organizações. Desse modo, é de suma importância o desenvolvimento de práticas, métodos e ferramentas que enfatizem princípios éticos e de responsabilidade social na produção de aplicações.

Responsabilidade social, segundo Ribeiro (2002), pode ser definida como o respeito ao direito dos outros, também entendido como a busca por agir corretamente ou de forma a contribuir para o desenvolvimento social. Para Benedicto *et al.* (2008) o conceito de responsabilidade social pode variar de pessoa para pessoa, onde para muitos pode representar a ideia de responsabilidade ou obrigação legal, enquanto que para outros indica um comportamento responsável no sentido ético.

Ao realizar uma revisão de literatura, contudo, constatou-se que ainda são poucos os trabalhos que explicitam o desenvolvimento socialmente responsável de *software* em todas as camadas de Engenharia de Software mencionadas por Pressman e Maxim (2016): Foco na Qualidade, Processos, Métodos e Ferramentas. Pode-se destacar os trabalhos de Dekhtyar *et al.* (2020) e Schneiderman (2020), que trazem contribuições para todas as camadas. O trabalho de Dekhtyar *et al.* (2020) visa o desenvolvimento de software socialmente responsável e o define como um software que tem como objetivo ajudar a comunidade local. Enquanto que Schneiderman (2020), propõe uma forma de gestão de sistemas de inteligência artificial visando alinhar os princípios éticos com as práticas de desenvolvimento.

Este trabalho de conclusão tem como objetivo, portanto, propor uma abordagem socialmente responsável ao desenvolvimento de *software*. Como objetivos específicos, tem-se:

- Caracterizar o desenvolvimento socialmente responsável de *software*;
- Adotar processos, métodos e ferramentas de Engenharia de Software para o desenvolvimento socialmente responsável de *software*;
- Desenvolver *software* de modo socialmente responsável;
- Compartilhar lições aprendidas durante o desenvolvimento deste Trabalho de Conclusão de Curso (TCC).

Para atingir esses objetivos, adota-se uma abordagem qualitativa de pesquisa (TERENCE; FILHO, 2006), utilizando de observação, da interpretação e da descrição de forma a abordar os aspectos éticos em uma experiência de desenvolvimento socialmente responsável de *software*. Envolve uma abordagem indissociável entre ensino, pesquisa e extensão.

O texto deste projeto de TCC está organizado em 4 capítulos, da seguinte forma: no Capítulo 1, tem-se a introdução com o contexto, justificativa, objetivos geral e específicos; no Capítulo 2, apresenta-se a revisão de literatura realizada; no Capítulo 3, a metodologia proposta para este trabalho é detalhada; no Capítulo 4, estão presentes os resultados e discussões; no Capítulo 5, as considerações finais e trabalhos futuros são apresentados.

2 ESTADO DA ARTE

A revisão de literatura realizada neste TCC adaptou as diretrizes para revisões sistemáticas propostas por Kitchenham (2004), visando analisar trabalhos relacionados a esta pesquisa. As seguintes atividades constituem a revisão de literatura:

- Definição da pergunta de pesquisa;
- Pesquisa em bases de busca;
- Definição de critérios de inclusão e exclusão;
- Leitura completa dos artigos e extração de dados.

2.1 Questão de Pesquisa

A primeira etapa realizada durante a revisão foi a formulação da questão de pesquisa: Como a Engenharia de Software tem abordado o desenvolvimento de software de modo socialmente responsável?

2.2 Strings e Bases de Busca

Levando em consideração a questão de pesquisa, foram definidas as palavras-chave e seus respectivos sinônimos: *Software Engineering* e *Software Development* representam Engenharia de Software; *Socially Responsible* e *Socially Conscious* representam responsabilidade social ou socialmente responsável, *Ethical* e *Ethic* significam Ética. O operador OR foi utilizado para relacionar sinônimos, enquanto que o operador AND foi utilizado para delimitar o escopo da busca.

Para realizar a busca, selecionaram-se bases que indexam trabalhos na área de Computação: ACM Digital Library, IEEE Xplore e Scopus. Conforme o padrão de cada base de busca, foram montadas *strings*, conforme apresentado no Quadro 1, a seguir.

Quadro 1 – Strings Aplicadas às bases de pesquisa

Base	String	Data da pesquisa
ACM Digital Library	Title:(("Software Engineering" OR "Software Development") AND ("Socially Responsible" OR "Socially Conscious" OR "Ethical" OR "Ethic")) OR Abstract:(("Software Engineering" OR "Software Development") AND ("Socially Responsible" OR "Socially Conscious" OR "Ethical" OR "Ethic")) OR Keyword:(("Software Engineering" OR "Software Development") AND ("Socially Responsible" OR "Socially Conscious" OR "Ethical" OR "Ethic"))	10/12/2021
IEEE Xplore	("Software Engineering" OR "Software Development") AND ("Socially Responsible" OR "Socially Conscious" OR "Ethical" OR "Ethic")	10/12/2021
Scopus	("Software Engineering" OR "Software Development") AND ("Socially Responsible" OR "Socially Conscious" OR "Ethical" OR "Ethic")	10/12/2021

Fonte: Autoria Própria

Na data de 10/12/2021, quando foi realizada a pesquisa, aplicando-se as respectivas *strings* nas bases de busca, foram encontrados: 25 trabalhos na ACM Digital Library, 100 na IEEE Xplore e 7 na Scopus. Importante ressaltar que na base ACM, foi realizada uma busca restrita ao título, à palavra-chave e ao resumo.

2.3 Critérios de Exclusão e Inclusão

Os critérios de exclusão e inclusão foram definidos de forma a auxiliar na seleção dos trabalhos encontrados após a aplicação das *strings* nas bases de busca, com o objetivo de identificar trabalhos que poderiam contribuir na abordagem da questão de pesquisa.

Esses critérios foram aplicados a partir na leitura do título, do resumo e das palavras-chave visando remover trabalhos que não possuíssem relação com a questão de pesquisa deste trabalho. Os critérios de exclusão aplicados foram:

- Estudos em forma de editoriais, anais de eventos, capítulos de livros, artigos resumidos, resumos, pôsteres e apresentações;

- Estudos duplicados;
- Estudos de acesso privado;
- Estudos com mais de 5 anos de publicação.

Na sequência da aplicação dos critérios de exclusão, foram aplicados os critérios de inclusão, seguindo o mesmo processo de leitura do título, resumo e palavras-chave, com o objetivo de selecionar os artigos com potencial relação com o proposto neste projeto. Os critérios de inclusão aplicados foram:

- Estudos publicados em inglês ou português;
- Artigos completos de conferências ou periódicos;
- Estudos que abordem qualidade de software, processo, métodos ou ferramentas de Engenharia de Software.

Após a aplicação dos critérios de exclusão e inclusão, chegou-se a 36 trabalhos, sendo 7 advindos da ACM Digital Library e 29 da IEEE Xplore. Estes foram, então, lidos em sua totalidade.

2.4 Resultados e Análises dos Trabalhos Relacionados

Nesta seção são analisados e discutidos os trabalhos que colaboram à abordagem da questão de pesquisa, iniciando com a identificação de métodos, processos, ferramentas e atributos de qualidade propostos nos trabalhos. Além disso, considerando o papel das pessoas na tomada de decisão ética e sua participação também é considerada.

2.4.1 Camadas da Engenharia de Software

Com a revisão dos trabalhos, identificaram-se 21 estudos que colaboram à abordagem da questão de pesquisa deste TCC, ou seja, os estudos apresentam de alguma forma soluções ou propostas para o desenvolvimento de software de modo socialmente responsável. O Quadro 2, a seguir, apresenta esses trabalhos e suas contribuições, organizadas e sumarizadas conforme as camadas de Engenharia de Software propostas por Pressman e Maxim (2016).

Quadro 2 – Relação de trabalhos identificados com as camadas de Engenharia de Software

Trabalho	Foco na Qualidade	Processos	Métodos	Ferramentas
Alidoosti (2021)			– Análise de Impacto – Modelo focado em Valores	– Mapa de <i>stakeholders</i>
Borg <i>et al.</i> (2021)				– Lista de Avaliação
Evangelou <i>et al.</i> (2021)	– Usabilidade		– Gamificação	
Serban <i>et al.</i> (2021)	– Confiabilidade – Segurança		– Teste de Viés Sociais – Auditorias	
Strandberg (2021)	– Confiabilidade			– Checklist
Dekhtyar <i>et al.</i> (2020)	– Acessibilidade	– Elicitação de Requisitos – Projeto – Implementação	– Reunião com <i>stakeholders</i> – Prototipação de <i>wireframes</i> – Histórias de usuário – Modelagem UML – Modelagem de Banco de Dados	– Post-it – UML – GGDB
Chakraborty <i>et al.</i> (2020)	– Confiabilidade		– Aprendizado de máquina	– FairWay (algoritmo proposto pelo autor)
Shneiderman (2020)	– Confiabilidade – Segurança (<i>safety</i>) – Usabilidade	– Verificação – Validação	– Teste de viés – Auditoria – Teste de usabilidade – Interface de usuário	– Ferramentas de análise
Cemiloglu <i>et al.</i> (2020)	– Requisitos éticos	– Engenharia de Requisitos	– Análise de documentos	

Continua

Continuação

Trabalho	Foco na Qualidade	Processos	Métodos	Ferramentas
Mustaffa <i>et al.</i> (2020)	<ul style="list-style-type: none"> – Requisitos éticos/religiosos – Autenticidade – Confiabilidade – Credibilidade 		<ul style="list-style-type: none"> – Escala de Classificação de Aplicativos Móveis 	
De (2020)	<ul style="list-style-type: none"> – Segurança 	<ul style="list-style-type: none"> – Abordagem ágil 	<ul style="list-style-type: none"> – Scrum – <i>Design Thinking</i> – Modelagem de Ameaças – Prototipação 	
Vakkuri <i>et al.</i> (2020)	<ul style="list-style-type: none"> – Transparência 		<ul style="list-style-type: none"> – <i>Design</i> Eticamente Alinhado 	<ul style="list-style-type: none"> – Anotações em cartões
Alami <i>et al.</i> (2019)	<ul style="list-style-type: none"> – Qualidade do código 		<ul style="list-style-type: none"> – Revisão de Código – Revisão por pares 	
Grant (2019)	<ul style="list-style-type: none"> – Segurança (<i>safety</i>) – Confiabilidade 			
Milosevic (2019)	<ul style="list-style-type: none"> – Privacidade 		<ul style="list-style-type: none"> – <i>Design</i> por Ética 	
Nagaria <i>et al.</i> (2019)		<ul style="list-style-type: none"> – Abordagem ágil 	<ul style="list-style-type: none"> – Scrum – Programação em pares 	<ul style="list-style-type: none"> – Google Forms – Microsoft Excel
Strandberg (2019)	<ul style="list-style-type: none"> – Requisitos éticos 		<ul style="list-style-type: none"> – Entrevistas éticas 	<ul style="list-style-type: none"> – <i>Checklist</i>
Jiandong e Jiaqi (2018)	<ul style="list-style-type: none"> – Qualidade do Processo 	<ul style="list-style-type: none"> – Gerenciamento de Engenharia de Software 		
Ghasempouri e Ladani (2017)	<ul style="list-style-type: none"> – Segurança (<i>security</i>) 			
Kammüller <i>et al.</i> (2017)	<ul style="list-style-type: none"> – Qualidade de Requisitos – Confiabilidade – Segurança – Privacidade 	<ul style="list-style-type: none"> – Elicitação de Requisitos 		<ul style="list-style-type: none"> – eFRIEND (Elicitação de Requisitos éticos) – Isabelle (Análise de Segurança)

Continua

Conclusão

Trabalho	Foco na Qualidade	Processos	Métodos	Ferramentas
Yilmaz <i>et al.</i> (2016)	– Confiabilidade		– Método Equilíbrio reflexivo	

Fonte: Autoria Própria

Práticas socialmente responsáveis no desenvolvimento de software e/ou aspectos éticos estão relacionados ao impacto social do produto final, como no caso do trabalho de Dekhtyar *et al.* (2020), onde foi produzida uma aplicação *mobile* com o objetivo de ajudar a comunidade local de realização do evento, caracterizando-se como um *software* socialmente responsável; Chakraborty *et al.* (2020), em sua proposta de modelo, visam a remoção de aspectos antiéticos e preconceituosos dos modelos existentes, apresentando ênfase no aprendizado de máquina; enquanto que os trabalhos de Jiandong e Jiaqi (2018) e Schneiderman (2020) enfatizam mais a qualidade do processo: Jiandong e Jiaqi (2018) têm como proposta a integração das diretrizes éticas com o gerenciamento do *software*, enquanto que Schneiderman (2020) tem como foco a governança de sistemas de Inteligência Artificial Centrados no Humano, preenchendo a lacuna entre a teoria e a prática com os princípios éticos.

Ao analisar esses trabalhos, pode-se notar que os atributos de qualidade mais recorrentes são confiabilidade, segurança e privacidade, aparecendo em mais da metade dos trabalhos que têm alguma contribuição na camada de Qualidade de Software. É possível notar com isso uma relação entre os aspectos éticos com esses atributos de qualidade.

Analisando a coluna relacionada à camada Processos, nota-se que em apenas sete estudos o processo está presente de modo explícito, com Abordagem Ágil e Engenharia de Requisitos sendo mencionados duas vezes cada. Enquanto os trabalhos de Cemiloglu *et al.* (2020) e Kammüller *et al.* (2017) enfatizam processos de Engenharia de Requisitos, os trabalhos de Nagaria *et al.* (2019) e De (2020) objetivam a unificação dos princípios dos códigos de ética com as práticas da abordagem ágil.

Ao verificar a coluna correspondente aos métodos, percebe-se maior diversidade, sendo possível identificar métodos utilizados em diferentes atividades

do ciclo de desenvolvimento de um *software*. Podem-se notar métodos que visam resolver problemas específicos, como por exemplo: teste de viés social no trabalho de Serban *et al.* (2021), que resolve o problema específico de aprendizado de máquina; a análise de impacto no trabalho de Alidoosti (2021), que visa clarificar sobre os possíveis impactos do software na sociedade e que a cada decisão o desenvolvedor sempre deve tê-los em mente; finalmente, Cemiloglu *et al.* (2020), que faz uso da análise de documentos como forma de definir os requisitos éticos.

Com relação às ferramentas, tem-se: o trabalho de Strandberg (2019), que propõe entrevistas éticas e aplicações de avaliações com o uso de *checklist* para validar se as diretrizes éticas foram respeitadas; Alidoosti (2021), que propõe a utilização de mapa de *stakeholders* como forma de apoiar a análise de impacto e conscientizar os desenvolvedores do impacto que o *software* pode gerar na sociedade; Vakkuri *et al.* (2020), que tem como ferramenta, anotações em cartões das decisões éticas realizadas pelos desenvolvedores como forma de sempre manter em mente a importância da ética durante o desenvolvimento.

Dentre os trabalhos analisados, destacam-se Dekhtyar *et al.* (2020) e Schneiderman (2020). Ambos abordam de modo explícito todas as camadas da Engenharia de Software mencionadas por Pressman e Maxim (2016).

O trabalho de Dekhtyar *et al.* (2020) propõem que eventos da comunidade de Engenharia de Software produzam software socialmente responsáveis nos locais de sua realização, ou seja, software que têm como objetivo ajudar a comunidade local. No projeto descrito pelos autores, foi produzido um aplicativo móvel de transporte público focado nas pessoas com mobilidade reduzida. O desenvolvimento foi dividido em 3 dias, com etapas bem definidas: elicitación e captura de requisitos, projeto e desenvolvimento. Acessibilidade era o atributo de qualidade em perspectiva, enquanto que os métodos utilizados foram reunião com partes interessadas, prototipação de *wireframes*, elaboração de histórias de usuário, modelagem UML (do inglês, *Unified Modeling Language*) e de banco de dados. Entre as ferramentas utilizadas estão *post-its*, ferramenta de modelagem UML para a produção do diagrama de classes e de casos de uso e a ferramenta GGDB (do inglês, *GGNetworks Database*) para a produção do modelo entidade relacionamento do Banco de Dados.

Já Schneiderman (2020) propõe em seu trabalho uma forma de gestão de sistemas de Inteligência Artificial Centrado no Humano (HCAI, *Human-Centered*

Artificial Intelligence), que visa preencher a lacuna entre os princípios éticos e a prática. Confiabilidade e segurança (*safety*) são os atributos de qualidade abordados no trabalho somados a privacidade, segurança (*security*), proteção ambiental, justiça social e direitos humanos. O autor propõe cinco práticas de Engenharia de Software que habilitam sistemas de HCAI, que são: auditoria/ferramenta de análise, fluxo de trabalho, verificação e validação, teste de viés e interfaces de usuário explicáveis. O autor menciona em sua conclusão que sistemas HCAI serão mais bem-aceitos caso as declarações sobre justiça, transparência, responsabilidade, segurança e privacidade saiam do papel e entrem de fato nas práticas conforme é indicado no trabalho.

2.4.2 Papel das Pessoas

O Manifesto Ágil (KENT *et al.*, 2001) reitera a importância do papel das pessoas no desenvolvimento de software. Conforme esse manifesto, “Indivíduos e interações, software em funcionamento, colaboração com o cliente e resposta a mudanças” devem ser mais valorizados que “Processos e ferramentas, documentação abrangente, negociação de contratos e seguir planos”, de modo que mesmo havendo valor nos últimos itens, valorizam-se mais os primeiros.

Além disso, é preciso ter em mente que software são utilizados por pessoas em suas rotinas diárias ou organizações. Seu desenvolvimento, com apoio de processos, métodos e ferramentas, é realizado por pessoas, que desempenham diferentes papéis em seu ciclo de vida – ex.: gestor, analista, projetista, *designer* de interação, programador, testador etc. Pessoas, portanto, sofrem impacto do software em suas vidas e têm o poder de provocar o impacto na vida de outras pessoas

A partir da análise dos 21 estudos identificados, que colaboram à abordagem da questão de pesquisa deste TCC, ou seja, estudos que apresentam de alguma forma soluções ou propostas para o desenvolvimento de software de modo socialmente responsável, evidencia-se, nesta seção, o papel das pessoas nesses trabalhos. O Quadro 3, a seguir, apresenta os estudos e suas contribuições, organizadas e sumarizadas conforme a participação e papel das pessoas no processo de desenvolvimento no que tange as questões éticas e de responsabilidade social.

Quadro 3 – Participação e papel das pessoas

Trabalho	Papel	Participação
Alidoosti (2021)	– Arquiteto	– Incorporação de aspectos éticos na tomada de decisões de arquitetura de software
Borg <i>et al.</i> (2021)	– Engenheiro de Software	– Uso de lista de avaliação para inteligência artificial confiável, no desenvolvimento de sistemas avançados de apoio ao motorista
Evangelou <i>et al.</i> (2021)	– Docente	– Uso de gamificação no ensino de dilemas tecnológicos e éticos na engenharia de software
Serban <i>et al.</i> (2021)	– Engenheiro de Software	– Uso de testes de viés social e auditorias para a produção de aplicativos de <i>Machine Learning</i> éticos.
Strandberg (2021)	– Testador	– Integração de Inteligência Artificial para a seleção de teste de regressão de modo que englobem os desafios éticos
Dekhtyar <i>et al.</i> (2020)	– <i>Designer</i> (Comunidade) – Engenheiro de software	– Levantamento de necessidades – Garantir que as necessidades da comunidade sejam atendidas com o uso de reuniões, validações e levantamento de <i>feedback</i>
Cemiloglu <i>et al.</i> (2020)	– Analista – <i>Designer</i>	– Uso das categorias de requisitos éticos propostas para uso de analistas e <i>designer</i> no contexto de tecnologias viciantes
Chakraborty <i>et al.</i> (2020)	– Engenheiro de Software	– Aplicação do método proposto (<i>Fairway</i>) durante o desenvolvimento, de modo a mitigar o viés ético nos modelos de <i>Machine Learning</i>
De (2020)	– Engenheiro de Software	– Uso de <i>Design Thinking</i> para modelar o sistema e de ameaças de modo a protegê-lo de falhas de segurança

Continua

Continuação

Trabalho	Papel	Participação
Mustaffa <i>et al.</i> (2020)	– Desenvolvedor de software islâmico	– Integração do código de ética no trabalho da religião islâmica ao desenvolvimento de aplicativos móveis
Shneiderman (2020)	– Auditor – Testador – <i>Designer</i>	– Realização de auditorias e testes de forma a garantir que as diretrizes éticas estão sendo aplicadas – Desenvolvimento de interface de usuário autoexplicável
Vakkuri <i>et al.</i> (2020)	– Engenheiro de Software	– Aplicação do método proposto (ECCOLA), que visa tornar os princípios éticos de Inteligência Artificial mais facilmente aplicáveis
Alami <i>et al.</i> (2019)	– Revisor de código	– Revisão de código e aplicação da ética do cuidado em códigos <i>open source</i>
Grant (2019)	– Engenheiro de Software	– Uso de considerações éticas no desenvolvimento de software para a indústria aviônica, de modo a garantir o alto nível de confiança necessário
Milosevic (2019)	– Engenheiro de Software	– Uso de lógica deontológica de modo a explicitar as preocupações éticas no desenvolvimento de software
Nagaria <i>et al.</i> (2019)	– Engenheiro de Software	– Uso de formulários éticos como forma de analisar a aplicação e vantagens da programação em pares
Strandberg (2019)	– Entrevistador	– Aplicação dos <i>checklists</i> propostos para a execução de entrevistas éticas
Jiandong e Jiaqi (2018)	– Docentes	– Inclusão dos princípios do código de ética no currículo dos cursos
Ghasempouri e Ladani (2017)	– Analista	– Uso do modelo proposto para análise de segurança de sistemas de confiança e reputação

Continua

Conclusão

Trabalho	Papel	Participação
Kammüller <i>et al.</i> (2017)	– Analista	– Uso das ferramentas (eFRIEND e Isabelle) para a produção de requisitos éticos em sistemas de internet das coisas nos cenários de saúde
Yilmaz <i>et al.</i> (2016)	– Engenheiro de Software	– Aplicação do método proposto de modo a desenvolver sistemas autônomos regidos por princípios éticos

Fonte: Autoria Própria

Dentre esses trabalhos, dois estão relacionados a docentes e/ou discentes: Jiandong e Jiaqi (2018) indicam a inclusão do ensino dos princípios éticos na disciplina de Gerenciamento de Engenharia de Software, enquanto que Evangelou *et al.* (2021) indicam que a gamificação pode ser uma boa alternativa para a abordagem pedagógica dos dilemas tecnológicos e éticos da Engenharia de Software.

Há também o papel do arquiteto de software, que aparece no trabalho de Alidoosti (2021). Este indica o uso de mapa de *stakeholders* e do modelo focado em valor como forma de incorporar aspectos éticos na tomada de decisão quanto à arquitetura de *software*.

Alami *et al.* (2019) trazem uma discussão interessante sobre o porquê que em comunidades *Open source* as revisões de código têm os melhores resultados. Chegam à conclusão que a ética do cuidado tem um papel fundamental quanto ao funcionamento das revisões. Mostram também o quanto é importante o papel da pessoa para atingir esses resultados.

Com relação ao papel de *designer*, este está presente em três trabalhos. Em Dekhtyar *et al.* (2020), tem-se a participação da comunidade de modo a explorar necessidades e indicar possíveis soluções de *design* durante o processo de desenvolvimento de um software socialmente responsável. Em Shneiderman (2020), tem-se a indicação de que a produção de software autoexplicativos é uma forma de atender a requisitos éticos e aumentar a confiabilidade de um sistema. Em Cemiloglu *et al.* (2020), tem-se uma indicação do uso de categorias éticas durante a produção de tecnologias viciantes.

Já o papel do Engenheiro de Software aparece em dez trabalhos, onde três – Vakkuri *et al.* (2020), Chakraborty *et al.* (2020) e Yilmaz *et al.* (2016) – discorrem sobre a aplicação dos métodos propostos em seus respectivos trabalhos, enquanto que os demais propõem o uso de ferramentas ou métodos diversos para a integração de princípios éticos no desenvolvimento de software.

2.5 Considerações Finais do Capítulo

Ao analisar os trabalhos relacionados, pode-se caracterizar o desenvolvimento socialmente responsável de software como o desenvolvimento de software onde os aspectos éticos devem ser considerados por suas partes interessadas, na definição de seus atributos de qualidade, assim como em seus processos, métodos e ferramentas, de forma que o software produzido atenda as necessidades implícitas e explícitas dos usuários, de forma ética e responsável.

Com relação à qualidade, nota-se uma diversidade quanto aos aspectos de qualidade que os trabalhos abordam. Há um alto índice de atributos de qualidade de produto, onde confiabilidade tem amplo destaque, aparecendo em mais da metade dos trabalhos. Já a privacidade se configura nos trabalhos que abordam a utilização de dados do usuário e têm forte preocupação com as novas leis de gestão de dados. Segurança, no sentido de *safety*, está presente nos casos de desenvolvimento de sistemas autônomos, onde a preocupação com a vida deve ser priorizada. Tem-se presente também requisitos éticos e legais no trabalho de Mustaffa *et al.* (2020), no qual tem-se a relação entre os requisitos éticos com a questão religiosa.

É possível notar, na camada relacionada a processos, que a Engenharia de Requisitos e a Abordagem Ágil estão relacionadas ao desenvolvimento socialmente responsável de *software*, visto que nos trabalhos identificados são os mais mencionados, enquanto que, no que tange aos métodos, tem-se uma diversidade de opções que se devem ao fato de que estão presentes em todas as fases do ciclo de desenvolvimento. Com relação às ferramentas, o *checklist* se destaca como forma de avaliação com base em listas e requisitos legais. Nessa perspectiva, tem-se o trabalho de Borg *et al.* (2021), que faz uma análise crítica da lista de avaliação de inteligência artificial confiável, proposta pela União Europeia, e de Vakkuri *et al.* (2020), que propõem a utilização de anotações em cartões para cada decisão ética

tomada durante o processo de desenvolvimento relacionando essa decisão com um requisito.

Apenas os trabalhos de Dekhtyar *et al.* (2020) e Schneiderman (2020) abordam de forma explícita todas as camadas da Engenharia de Software. Aspectos éticos, contudo, devem ser considerados em todas as camadas, sendo relevante abordá-los de modo intencional, no desenvolvimento de software.

A responsabilidade ética no desenvolvimento de software é uma atribuição das pessoas. Essa responsabilidade precisa ser evidenciada. Processos, métodos e ferramentas devem fomentar e instigar as pessoas participantes do processo de desenvolvimento a pensar de maneira ética a cada tomada de decisão, promovendo a qualidade do software que atenda às suas necessidades e tragam impacto positivo em suas vidas.

3 METODOLOGIA

Neste Capítulo, é apresentada a metodologia utilizada neste trabalho. Adota-se uma abordagem qualitativa de pesquisa (TERENCE; FILHO, 2006), utilizando de observação, da interpretação e da descrição de forma a abordar os aspectos éticos em uma experiência de desenvolvimento socialmente responsável de software.

Envolve uma abordagem indissociável entre ensino, pesquisa e extensão: tem-se a articulação deste TCC com o programa de extensão TRAMAS (código SAP 522), acrônimo para Tecnologia, Responsabilidade, Autoria, Movimento, Amorosidade e Sociedade, que tem como objetivo a promoção do respeito à multiplicidade das diferenças; com o projeto de pesquisa Inovação em Domínio Educacional (IDE) (registro SIPPEE 20210319171721), que objetiva a inovação dos processos de ensino, aprendizagem e gestão no domínio educacional; e com o projeto de ensino GEIHC – Grupo de Estudos em Interação Humano-Computador (registro SIPPEE 01.003.21) que propõe acompanhar estudos dirigidos na área de Interação Humano-Computador.

Este trabalho adota as camadas de Engenharia de Software descritas por Pressman e Maxim (2016), de modo a organizar e explicitar que a abordagem proposta integra os aspectos éticos com as camadas de Engenharia de Software, caracterizando assim o desenvolvimento socialmente responsável de *software*. Além disso, valoriza o papel de suas partes interessadas nas tomadas de decisão ética.

Na Figura 1, tem-se as camadas de Engenharia de Software propostas por Pressman e Maxim (2016), onde a Engenharia de Software é fundamentada no comprometimento com a qualidade, de forma que o foco na qualidade é sua “pedra fundamental”. Para alcançar a qualidade, há os processos, os métodos e as ferramentas, onde os processos são constituídos pela base do gerenciamento dos projetos e pela definição do contexto de emprego dos métodos, de quando artefatos devem ser produzidos e marcos estabelecidos. Acima da camada de processos, têm-se os métodos que são atividades que fornecem informações técnicas durante o desenvolvimento. Estes envolvem, por exemplo: comunicação, análise de requisitos, modelagem de projeto, construção de programa, testes etc. Por fim, estão as ferramentas que oferecem apoio automatizado ou semiautomatizado para os processos ou métodos.

Figura 1 – Camadas da Engenharia de Software



Fonte: Pressman e Maxim (2016, p.16)

3.1 Camadas de Engenharia de Software neste Trabalho

O Quadro 4, a seguir, explicita como as camadas de Engenharia de Software, apresentadas em Pressman e Maxim (2016), foram abordadas neste trabalho.

Quadro 4 – Camadas de Engenharia de Software neste trabalho

Foco na Qualidade	Processos	Métodos	Ferramentas
<ul style="list-style-type: none"> – ISO 25010 (2011) – Requisitos Éticos/Legais 	<ul style="list-style-type: none"> – Engenharia de Requisitos – Desenvolvimento incremental 	<ul style="list-style-type: none"> – <i>Brainstorming</i> – Oficina Semio-participativa – <i>Mockups</i> – Validação de Software – Modelagem UML – Avaliação Heurística de Usabilidade – Avaliação SUS 	<ul style="list-style-type: none"> – Termos de Consentimento Livre e Esclarecido (TCLE) – Mapa de <i>Stakeholders</i> – Quadro de Avaliação – Escada Semiótica – <i>Checklist</i> – Figma – Google (Drive, Docs e Forms) – Astah UML – Trello – Lápis, papel e caneta

Fonte: Autoria Própria

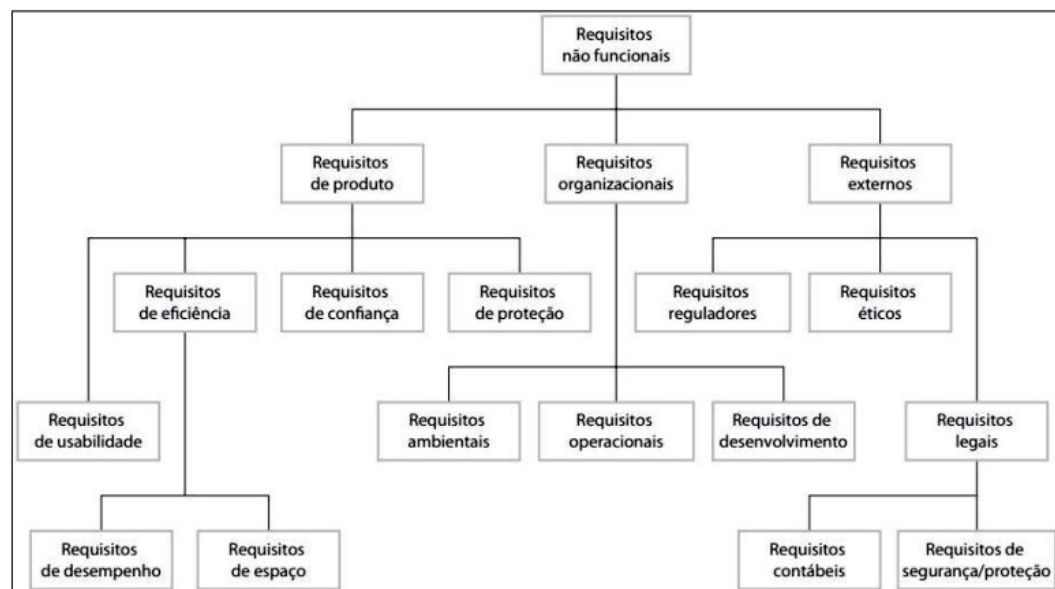
3.1.1 Foco na Qualidade

De acordo com a IEEE (1990), qualidade é definida como o grau de conformidade entre um sistema, componente ou processo com os seus respectivos requisitos. De modo a fornecer um modelo de referência com os atributos de qualidade que um produto deve ter para satisfazer as necessidades implícitas e

explícitas, foram desenvolvidas diversas normas pela *International Organization Standardization* (ISO) em conjunto com a *International Electrotechnical Commission* (IEC). A ISO/IEC 25010:2011, por exemplo, define algumas características de qualidade, como: Funcionalidade, Confiabilidade, Usabilidade, Eficiência, Manutenibilidade, Portabilidade, Segurança e Compatibilidade (ISO, 2011). Trata-se, assim, de um modelo de referência básico para a avaliação da qualidade de um software. Neste trabalho, Funcionalidade e Usabilidade foram definidos como atributos de qualidade a serem priorizados, de modo que, o software produzido tenha completude, correção e adequação funcional e seja acessível, fácil de aprender e operar.

Na Figura 2, a seguir, apresenta-se a categorização dos requisitos não funcionais proposta por Sommerville (2011), onde os Requisitos Éticos são parte dos requisitos que derivam de fatores externos ao desenvolvimento, de modo que representam aqueles que asseguram que o sistema seja aceitável pelos seus usuários e a sociedade em geral. Como, por exemplo: o sistema não fará uso dos dados privados dos usuários sem o seu consentimento.

Figura 2 – Tipos de requisitos não funcionais



Fonte: Sommerville (2011, p. 61)

Neste trabalho, foram utilizadas os atributos de qualidade propostos pela ISO e os requisitos não funcionais externos, procurando alinhá-los ao código de ética da

ACM/IEEE para produzir uma lista de requisitos éticos para guiar o desenvolvimento de forma socialmente responsável.

3.1.2 Processos e Métodos

Este trabalho, após a realização da etapa de organização de pesquisa, foi dividido em dois subprocessos principais, que são: Engenharia de Requisitos e Desenvolvimento Incremental.

A Engenharia de Requisitos é definida como um processo para a descoberta, análise, documentação e verificação das restrições e necessidades do cliente (SOMMERVILLE, 2011). Os métodos que apoiaram a execução do processo de Engenharia de Requisitos são: Tempestade de ideias (do inglês, *Brainstorming*), Oficina Semio-participativa (OSp), *Mockups* e Validação de Requisitos. Ao final do processo, foi produzido um Documento de Requisitos que integra as necessidades dos usuários com as diretrizes éticas.

Para Sommerville (2011), o Desenvolvimento Incremental é uma abordagem baseada na ideia de desenvolver uma série de incrementos, de modo que a cada versão uma nova funcionalidade seja adicionada. Especificação de requisitos, desenvolvimento (implementação e testes) e validação são realizados de forma intercalada. Neste trabalho, implementação, testes e validação são realizados de forma intercalada.

Os métodos mencionados são detalhados e explorados na seção 3.2, que aborda sua aplicação no processo de desenvolvimento de software organizado para este TCC.

3.1.3 Ferramentas

Nesta seção, abordam-se as ferramentas utilizadas neste projeto, explicitando qual método ou processo elas apoiaram.

Este trabalho fez uma adaptação do modelo de Termo de Consentimento Livre e Esclarecido (TCLE) apresentado em Freitas (2021), tendo como objetivo esclarecer a todos os participantes do projeto as atividades, vantagens e desvantagens quanto à colaboração com este trabalho.

O Mapa de *Stakeholders* (BARANAUSKAS, 2013), também chamado de Diagrama de Partes Interessadas, foi utilizado de forma a clarificar o impacto do *software* proposto na sociedade e desta no próprio *software*. Foi elaborado durante a OSp, de forma que o grupo de participantes classificou as partes interessadas nas categorias propostas por Baranauskas *et al.* (2013), que são: Atores e Responsáveis, aqueles que contribuem diretamente ao problema ou solução; Clientes e Fornecedores, fornecem dados ou são fonte de informações; Parceiros e Concorrentes, aspectos do mercado; Espectador e Legislador, representantes da comunidade que influenciam ou são influenciados pelo problema.

Também como subproduto da OSp, o Quadro de Avaliação (BARANAUSKAS, 2013) foi utilizado para auxiliar na articulação do problema e levantamento de ideias ou soluções vislumbradas, que terão potencial impacto no desenho da solução para o problema. Já a Escada Semiótica (BARANAUSKAS, 2013; LIU, 2000) foi utilizada para auxiliar na organização e levantamento de questões em todas as suas seis camadas: Social, Pragmática, Semântica, Sintática, Empírica e no Mundo Físico.

Foi utilizado também, a Figma¹ para o desenvolvimento dos *mockups* e lápis, papel e caneta durante a prototipação em papel. A ferramenta Trello² foi adotada como no gerenciamento e no controle das atividades do ciclo de desenvolvimento. As ferramentas da Google (*Docs*³, *Form*⁴, *Drive*⁵) foram utilizadas para a produção e armazenamento de documentos e formulários que apoiam este trabalho.

3.2 Etapas para o desenvolvimento do Trabalho

A Figura 3, a seguir, apresenta como foi organizada a metodologia proposta para o desenvolvimento deste TCC. Conforme já mencionado, a metodologia está organizada em três etapas: Organização da Pesquisa, Engenharia de Requisitos e Desenvolvimento Incremental.

1 <https://figma.com>

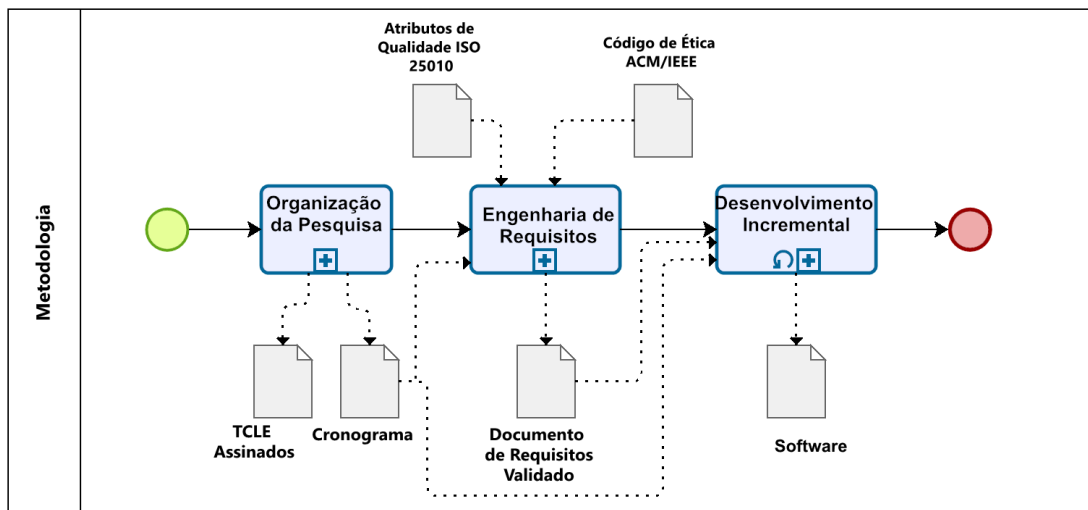
2 <https://trello.com>

3 <https://docs.google.com>

4 <https://docs.google.com/forms>

5 <https://drive.google.com>

Figura 3 – Macroprocesso metodologia

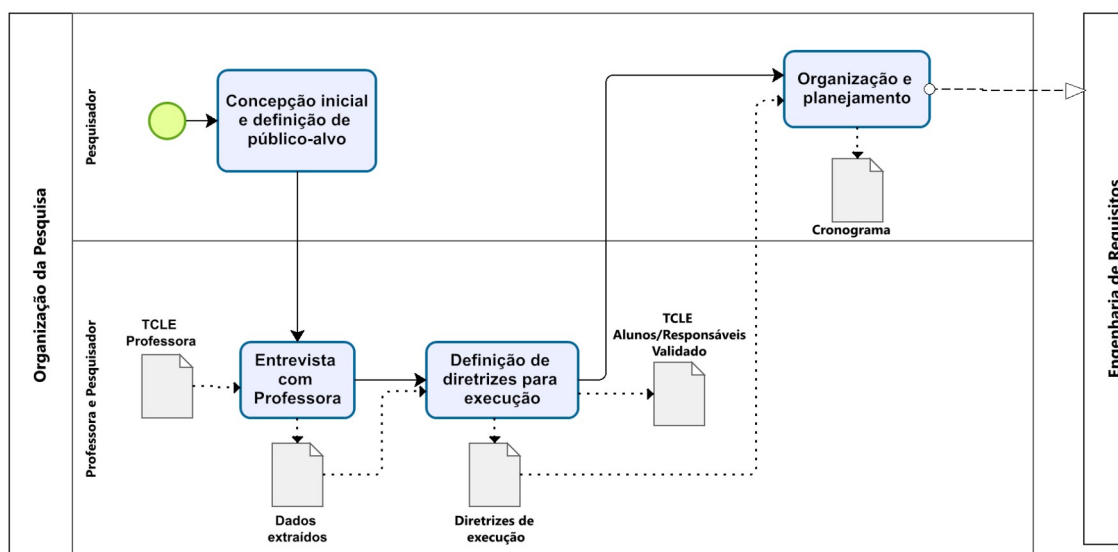


Fonte: Autoria Própria

3.2.1 Organização da Pesquisa

Esta etapa teve como objetivo a verificação da viabilidade, organização e planejamento deste TCC. Na Figura 4, a seguir, destacam-se os papéis do pesquisador, que representa o autor deste trabalho, e da professora de uma escola pública, que colaborou com sua execução.

Figura 4 – Organização da pesquisa



Fonte: Autoria Própria

O processo teve início na concepção e na definição do público-alvo. Visto que o objetivo deste projeto visa propor uma abordagem socialmente responsável ao desenvolvimento de *software*, buscou-se um grupo de pessoas para a realização de Engenharia de Requisitos e desenvolvimento participativo. Com isso, inicialmente teve-se a ideia de realizar este projeto com estudantes do ensino médio da EEEM Waldemar Borges de Alegrete/RS.

Como primeiro passo, envolvendo a professora e o pesquisador, ocorreu a leitura do TCLE (APÊNDICE B). Nesse momento, a professora foi informada dos objetivos do projeto, do caráter voluntário de sua colaboração, das vantagens e desvantagens ao participar, sendo convidada a assiná-lo.

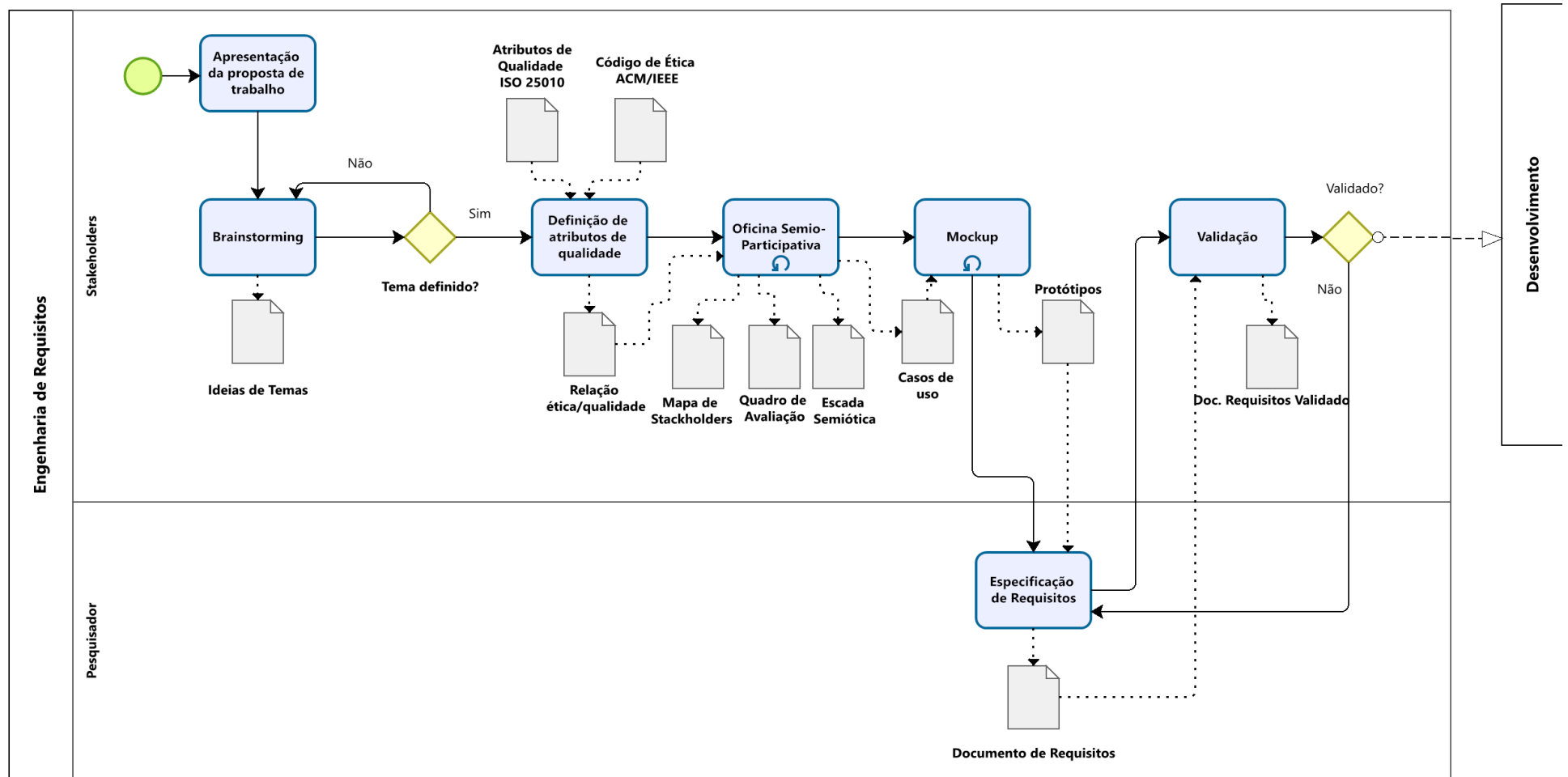
Na sequência, foi realizada uma entrevista com a professora, seguindo o roteiro (APÊNDICE C), tendo como objetivo a busca por mais informações com relação à infraestrutura e disponibilidade de tempo das turmas – visto que este projeto necessitaria de alguns encontros com os estudantes, era necessário verificar a viabilidade em questões de tempo e em questões de interesse por parte dos alunos e da professora. Como saída dessa atividade, tem-se um documento com os dados extraídos dessa entrevista.

A atividade de definição de diretrizes teve como objetivo a discussão entre pesquisador e professora de aspectos importantes no contato com os alunos e definição de que forma e quantos encontros seriam viáveis de serem realizados. Nessa etapa foi realizada a validação do Termo de Consentimento Livre e Esclarecido para os Alunos e Responsáveis (APÊNDICE D). Após isso, na atividade Organização e Planejamento, foi possível produzir um cronograma de atividades com a turma.

3.2.2 Engenharia de Requisitos

A Figura 5, a seguir, representa a etapa de Engenharia de Requisitos que teve como objetivo definir o software desenvolvido de forma colaborativa, levando sempre em consideração as diretrizes éticas, visto que tanto o software desenvolvido quanto o processo de desenvolvimento devem estar eticamente alinhados.

Figura 5 – Processo de Engenharia de Requisitos



Fonte: Autoria Própria

Esta etapa começou com a apresentação da proposta de trabalho e sobre responsabilidade social e ética para a turma. Então, foi verificado se a turma tinha interesse na participação deste projeto, sendo realizada a leitura e a distribuição dos TCLE aos alunos (APÊNDICE D).

Como segunda atividade, foi realizado um *Brainstorming* – técnica de dinâmica em grupo, que tem como objetivo explorar a criatividade da equipe, na forma de uma ou várias reuniões onde são levantadas e exploradas ideias sem críticas ou julgamentos (CARVALHO; CHIOSSI, 2001). Tinha-se como objetivo levantar possíveis temas para o software que seria desenvolvido, levando em consideração que esse software deve apresentar uma contribuição social.

Na definição de atributos de qualidade, foi apresentada aos *stakeholders* as diretrizes de qualidade da ISO 25012 e as diretrizes éticas da ACM/IEEE, para que assim fossem selecionadas e relacionadas na forma de um documento para servir como base para validações e tomada de decisões éticas durante todo o desenvolvimento. Foi produzido como resultado dessa atividade uma lista de requisitos éticos a partir das diretrizes éticas.

Então, foram conduzidas duas Oficinas Semio-participativas, conforme Baranauskas *et al.* (2013), que são práticas participativas e inclusivas alinhadas à visão da Semiótica Organizacional (LIU, 2000). Essas práticas têm como base o Design Participativo (DP) (MULLER *et al.*, 1997) e visam abordar em um mesmo ambiente diferentes visões de mundo e pluralidade de experiências em um processo de *co-design*. A parte inicial das OSp tem como objetivo a clarificação do problema, buscando as percepções de todos os envolvidos no processo de *design* e vai se aprimorando ao longo do tempo. Na primeira oficina, realizada foram produzidos o Mapa de *Stakeholders* e o Quadro de Avaliação, enquanto que na segunda oficina foi validada a Escada Semiótica produzida.

Este trabalho fez uso da técnica *Mockups* que utiliza o *design* iterativo para criar ciclos de prototipação e avaliação da interface de usuário (MULLER *et al.*, 1997), de modo a clarificar ideias e validar os requisitos com a turma. E também como forma de avaliação foi aplicada a Avaliação Heurística de Usabilidade (NIELSEN, 1994), que é uma técnica de inspeção de interface com o objetivo de avaliar a usabilidade do *design* proposto, explicitando problemas de usabilidade. Adotou-se a planilha apresentada no APÊNDICE A como apoio à aplicação dessa

avaliação. A atividade de *mockup* foi realizada em três iterações: na primeira e na segunda ocorreu a prototipação em papel pela turma, enquanto que na terceira foi produzido um protótipo por este pesquisador com o auxílio da ferramenta Figma.

A Prototipação, para Pressman e Maxim (2016), é um paradigma que pode ser utilizado tanto como um modelo de processo de desenvolvimento, quanto como técnica de Engenharia de Requisitos para auxiliar na compreensão do que deve ser desenvolvido. Para Barbosa e Silva (2010), a prototipação quando usada na Engenharia de Requisitos é uma atividade na qual versões iterativas de propostas de *design* são produzidas para serem validadas e pode ser realizada em papel (média ou baixa fidelidade) ou de modo digital (alta fidelidade). A prototipação em papel, conforme Rogers *et al.* (2013), pode ser usada como forma de comunicação entre desenvolvedor e usuários, visto que os usuários podem expressar de forma rápida e já testar um *design*. A prototipação em papel foi utilizada para a ajuda na clarificação dos requisitos e levantamento de ideias de *design*. A prototipação foi utilizada pois é uma forma mais objetiva de elicitação e de validação de requisitos (BARBOSA; SILVA, 2010).

Após a aplicação da Avaliação Heurística de Usabilidade e correção dos problemas de usabilidade encontrados, iniciou-se a atividade de Especificação de Requisitos por parte do pesquisador, onde foi produzido um documento de especificação que expressa formalmente os requisitos e os relaciona aos aspectos éticos que orientam a etapa de Desenvolvimento Incremental.

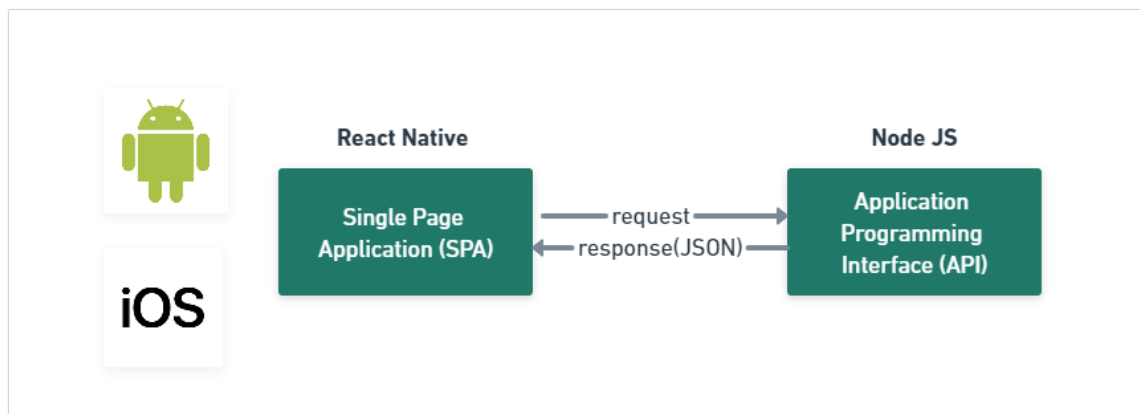
Com o Documento de Requisitos concluído, fez-se necessária a atividade de validação de requisitos que tem como objetivo verificar se o que está especificado é de fato o que os *stakeholders* solicitaram.

3.2.3 Desenvolvimento Incremental

O processo de desenvolvimento executado neste projeto de TCC foi uma adaptação do desenvolvimento incremental proposto por Sommerville (2011). No entanto, apenas as atividades de implementação, testes e validação foram incrementais, visto que a especificação foi realizada de forma completa no Processo de Engenharia de Requisitos. Como primeiro passo no processo de desenvolvimento, foram definidas as tecnologias que seriam usadas e como a

solução seria implementada. A Figura 6 representa as duas aplicações: uma desenvolvida em Node e outra aplicação *mobile* em *React Native*.

Figura 6 – Aplicações desenvolvidas



Fonte: Autoria Própria

Optou-se pelo desenvolvimento de duas aplicações independentes e desacopladas. Cada uma com uma responsabilidade bem definida, onde a aplicação desenvolvida em Node é responsável pela parte de servir dados e a aplicação em *React Native* é a responsável pelo lado do cliente. A comunicação entre elas se dá seguindo os padrões da arquitetura cliente-servidor. No entanto, na implementação da API foi aplicado o padrão *Model-View-Controller (MVC)* e, na aplicação *mobile*, foi implementada a arquitetura em camadas, devido à divisão de componentes por sua funcionalidade, possibilitando assim o reúso.

Com essas decisões tomadas, já foi possível definir banco de dados, ferramenta para controle de versão, forma de hospedagem da aplicação de *back-end* e ferramenta de testes, conforme Quadro 5.

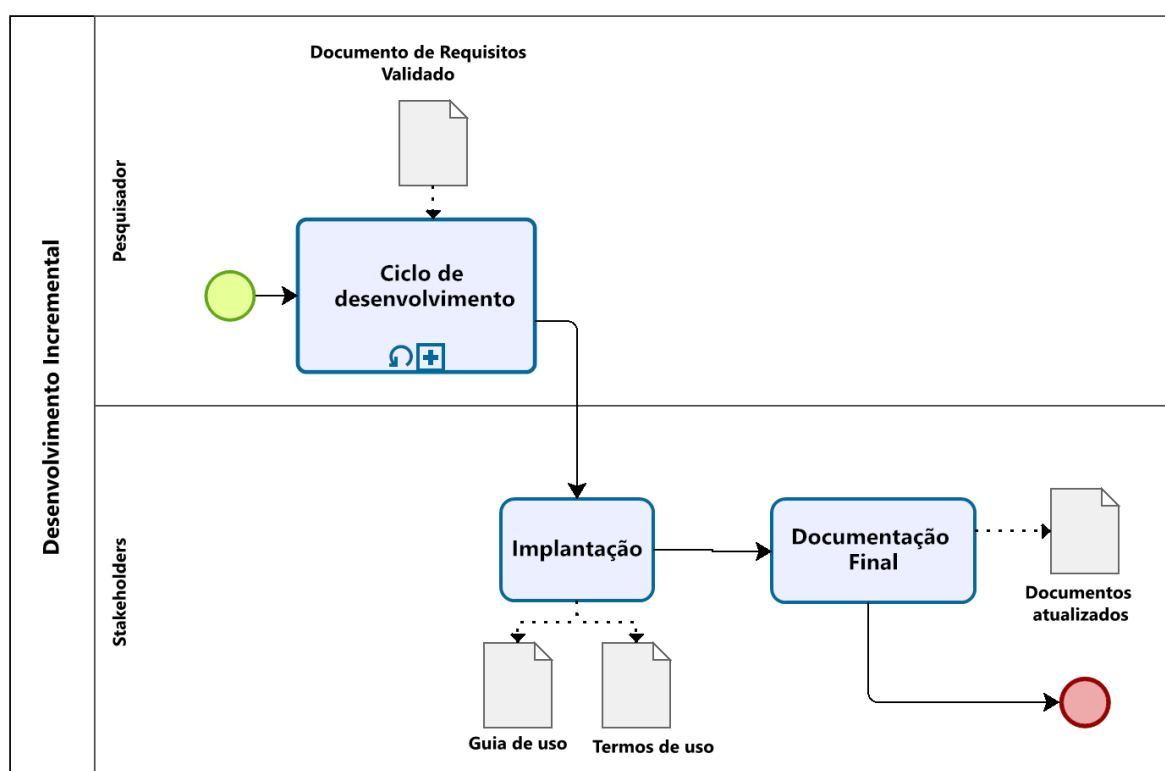
Quadro 5 – Tecnologias e ferramentas utilizadas

Atividade/Método	Tecnologia/Ferramenta
Banco de dados	SQLite
Controle de versão	Git (Github)
Servidor de hospedagem	Kinghost
Teste(Back-end e Front-end)	Jest

Fonte: Autoria Própria

Na Figura 7, a seguir, é representado o processo de desenvolvimento incremental do produto deste TCC.

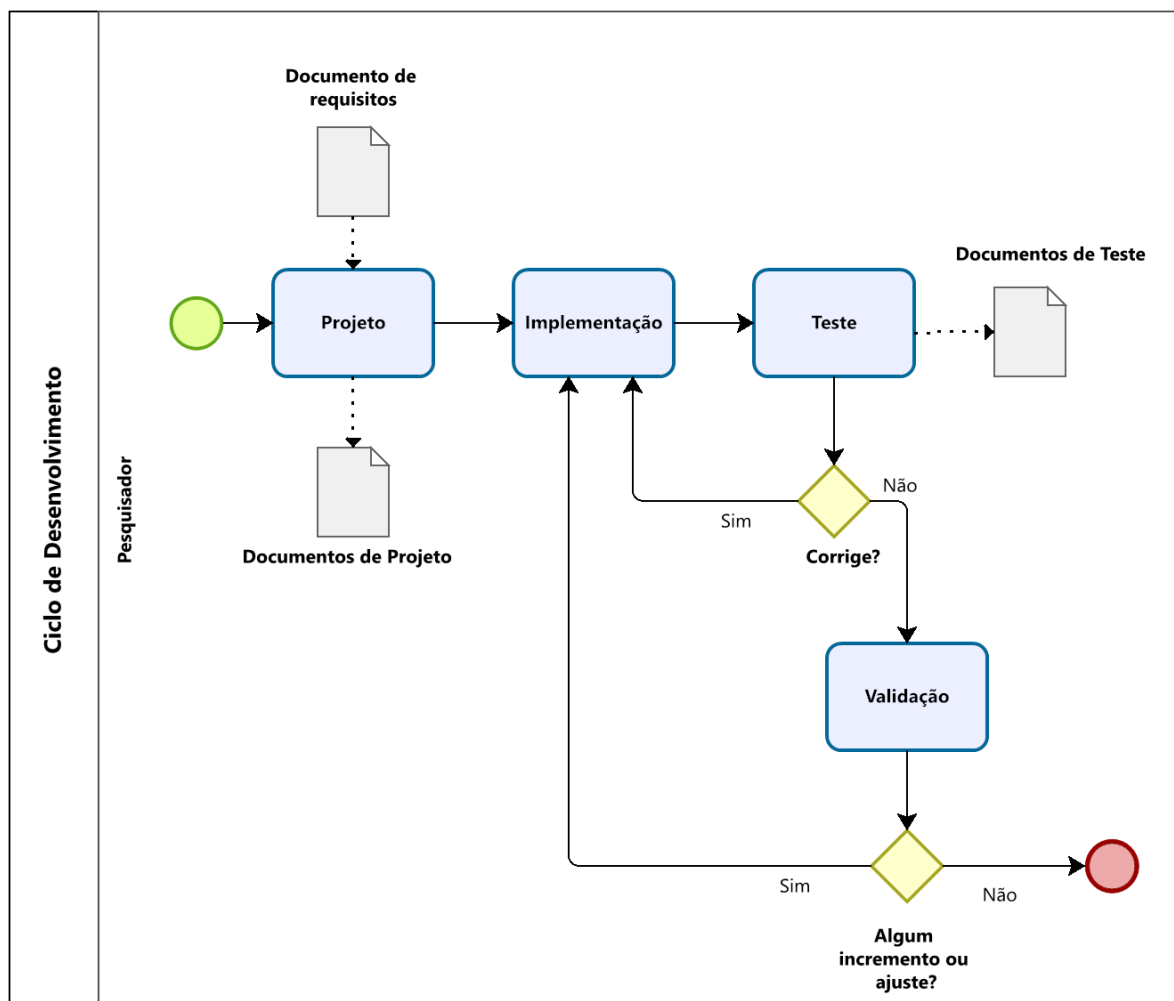
Figura 7 – Processo de Desenvolvimento Incremental



Fonte: Autoria Própria

Esse processo de desenvolvimento foi composto por 2 ciclos de desenvolvimento que priorizou os casos de uso de maior prioridade, seguido da implantação e atualização da documentação final. Na Figura 8, estão as atividades realizadas a cada ciclo de desenvolvimento.

Figura 8 – Ciclo de desenvolvimento



Fonte: Autoria Própria

Com os requisitos priorizados e as decisões de ferramentas e tecnologias que seriam utilizadas definidas, organizaram-se os Casos de Uso especificados em dois ciclos de desenvolvimento, que teve como primeira atividade o projeto, a implementação, testes e a validação.

Foram realizados dois ciclos de desenvolvimento incremental, onde em cada uma desses ciclos foram realizadas as atividades de projeto, implementação, teste e validação. No primeiro ciclo de desenvolvimento, foi definida a implementação dos seguintes Casos de Uso: autenticar-se, manter itens para doação e manter informações dos ônibus. O período desse ciclo de desenvolvimento foi entre os dias 13 de junho de 2022 e 27 de junho de 2022. Onde durante esse tempo, foi produzida uma versão conceitual do banco de dados, que apresentava apenas os dados

necessários para a execução das atividades propostas. Ocorreu a implementação e logo após aplicados testes unitários com a ferramenta *Jest*.

No dia 27 de junho de 2022, foi disponibilizada a primeira versão do aplicativo mobile “ComuniApp”, que funcionava apenas localmente visto que a aplicação de *back-end* não havia sido hospedada. Foi aplicada uma avaliação SUS (APÊNDICE N), de modo a avaliar a interface desenvolvida.

A partir do dia 27 de junho de 2022 até o dia 18 de julho de 2022, foi executado então o segundo ciclo de desenvolvimento incremental, objetivando inicialmente corrigir os problemas pontuados durante a primeira avaliação e após implementando os Casos de Uso que faltavam. Como primeiro passo ocorreu o ajuste da autenticação com a plataforma Google, fazendo com que o fluxo de autenticação ocorresse corretamente, então, na sequência foi refatorado o banco de dados para as novas funcionalidades.

Com isso foi implementada e atualizada a aplicação do *back-end* e realizados testes unitários em todas as funções disponíveis. Com a parte do servidor completa, foi iniciada a implementação dos Casos de Uso que ainda faltavam ser implementados, de modo que foi realizada a sua implementação e realizados testes unitários com a ferramenta *Jest*.

Após a realização dos dois ciclos de desenvolvimento, teve início a atividade de implantação de *software*. Esta teve como objetivo produzir artefatos que ajudem os usuários na utilização de software no seu ambiente e a produção dos termos de uso. Por fim, foi realizada a atividade de Documentação Final que teve como foco a revisão e atualização dos documentos produzidos e a publicação do código desenvolvido em um repositório.

3.3 Considerações Finais do Capítulo

Propõe-se, neste TCC, portanto, uma abordagem socialmente responsável para o desenvolvimento de software, que gere um produto de software que traga benefícios aos seus usuários.

Com relação à qualidade, este trabalho aborda a integração dos atributos de qualidade da ISO/IEC 25010 (ISO, 2011) e o Código de Ética e Prática Profissional

de Engenharia de Software ACM/IEEE (ACM/IEEE, 2015), gerando assim Requisitos de Qualidade Éticos.

Já para a camada de processos, faz-se uso de Engenharia de Requisitos, presente nos trabalhos de Dekhtyar *et al.* (2020), Cemiloglu *et al.* (2020) e Kammüller *et al.* (2017), e Desenvolvimento Incremental como forma de aplicação da Abordagem Ágil, que está presente nos trabalhos de Nagaria *et al.* (2019) e De (2020). A Engenharia de Requisitos, em particular, produz um documento de requisitos no qual estão incorporados requisitos éticos junto aos requisitos não funcionais.

Entre os métodos utilizados, tem-se: *Brainstorming* e Prototipação como forma de elicitação de requisitos, já as Oficinas Semio-participativas para apoiar a análise e a especificação de requisitos. Destaca-se o Mapa de *Stakeholders* que auxilia a clarificar o impacto do software proposto na sociedade e desta sobre o próprio software.

Todos os processos, métodos e artefatos têm como base as diretrizes éticas propostas pela ACM/IEEE e foram guiados pelos requisitos éticos produzidos. Além disso, a participação do usuário final durante o processo de desenvolvimento é de fundamental importância, visto que eles são os maiores impactados pela solução final e devem estar presentes em todas as decisões éticas tomadas com relação ao produto.

4 RESULTADOS E DISCUSSÕES

Neste Capítulo, são apresentados e discutidos os resultados obtidos durante o desenvolvimento deste TCC.

4.1 Organização da Pesquisa

A partir da realização da entrevista com a professora, foi possível verificar a disponibilidade de uma turma de primeiro ano do ensino médio, com cerca de 25 alunos, para colaborar com este trabalho. Essa colaboração poderia ocorrer na disciplina chamada “Cultura Digital e Tecnologias”, com duas aulas semanais.

Nesta etapa, o TCLE para os alunos e seus respectivos responsáveis foi também elaborado (APÊNDICE D), além de serem definidas as diretrizes de execução: encontros com a turma nas segundas-feiras, das 19:30 as 20:15; caso necessário outro dia, sextas-feiras, das 20:30 as 21:15, mas sempre com o aceite do professor responsável; em caso de necessidade de uso de projetor, o professor teria que ser informado com antecedência; a melhor forma para interação com os alunos seria na forma de levantamento de discussões e atividades em grupo.

Organizou-se, ainda, um cronograma de atividades inicial (APÊNDICE E). Este, ao longo do desenvolvimento das atividades, foi atualizado. O Quadro 6 apresenta a última atualização do cronograma.

Quadro 6 – Cronograma de participação da turma

Data	Atividade
1º/04	<ul style="list-style-type: none"> • Apresentação da proposta de trabalho • Entrega de TCLE aos alunos • Apresentação sobre ética e responsabilidade social
11/04	<ul style="list-style-type: none"> • Brainstorming • Relação ética e qualidade
18/04	<ul style="list-style-type: none"> • 1ª Oficina Semio-Participativa
02/05	<ul style="list-style-type: none"> • 2ª Oficina Semio-Participativa
09/05	<ul style="list-style-type: none"> • Prototipação em papel • Validação documento de requisitos
16/05	<ul style="list-style-type: none"> • Prototipação em papel e consolidação
10/06	<ul style="list-style-type: none"> • Validação 2º mock-up • Priorização de requisitos

Continua

Conclusão

Data	Atividade
27/06	<ul style="list-style-type: none"> Validação primeira versão aplicativo
18/07	<ul style="list-style-type: none"> Validação final

Fonte: Autoria Própria

4.2 Engenharia de Requisitos

Como resultado da atividade *Brainstorming*, foram geradas ideias para um software socialmente responsável. No Quadro 7, a seguir, essas ideias são listadas.

Quadro 7 – Funcionalidades

Funcionalidade
Manter os dados de funcionamento de postos de saúde, escolas e Centro de Referência de Assistência Social (CRAS) do bairro.
Manter dados de itens disponíveis para doação, como forma de ser uma interface entre doador e donatário.
Manter horários de ônibus que tenham itinerário no bairro.
Publicação de avisos para a comunidade.

Fonte: Autoria Própria

Os temas abordados inicialmente foram: Registro de avisos para a comunidade, Solicitações ao poder público, Guia do bairro, Registro de bens para doação, Registro de denúncias, Banco de questões para uso didático e Validação de presenças via aplicativo. A partir da priorização dos temas em grupo, foram definidas as 3 funcionalidades principais do aplicativo: fazer solicitações via aplicativo ao poder público, registrar informações das instituições públicas locais e registrar avisos.

Quanto aos atributos de qualidade, a turma definiu que Funcionalidade, Usabilidade e Acessibilidade deveriam ser priorizados. Sobre os aspectos éticos, no APÊNDICE F, é possível visualizar os dados extraídos, enquanto que no Quadro 8, a seguir, está a relação dos requisitos éticos (RE) mapeados.

Quadro 8 – Requisitos éticos

RE	Descrição
RE 1	Desenvolver software que satisfaça as especificações, seja testado e não diminua a qualidade de vida ou a privacidade, ou cause danos ao meio ambiente.
RE 2	Considerar questões de deficiência física, alocação de recursos, desvantagem econômica ou outros fatores que possam diminuir o acesso aos benefícios do software. (RNF - Acessibilidade)
RE 3	Não usar, se souber, software obtido ou mantido de forma ilegal ou aética.
RE 4	Manter privada qualquer informação confidencial obtida por intermédio do trabalho profissional, desde que essa confidencialidade seja consistente com o interesse público e com a legislação.
RE 5	Assegurar que o método adequado é usado por qualquer projeto em que trabalhe ou se proponha a trabalhar.
RE 6	Assegurar que as especificações do software no qual trabalhe foram bem documentadas, satisfaçam os requisitos dos usuários e tenham as aprovações apropriadas.
RE 7	Trabalhar para desenvolver software e documentos relacionados que respeitem a privacidade daqueles que serão afetados por eles.
RE 8	Considerar em todos os julgamentos técnicos as necessidades de apoiar e manter os valores humanos.
RE 9	Assegurar boa gestão para qualquer projeto no qual trabalhe, incluindo procedimentos efetivos para a promoção da qualidade e redução dos riscos.

Fonte: Autoria Própria

A Figura 9, a seguir, apresenta o Diagrama de Partes Interessadas, resultado da primeira Osp. Esse buscou mapear *stakeholders* e classificá-los em contribuição, fonte, mercado ou comunidade. Em contribuição, foram explicitadas as pessoas que são impactadas diretamente pelo software: cidadão, morador novo no bairro e comerciante local; fonte são os *stakeholder* que podem servir como clientes ou fornecedores de dados para a aplicação: CRAS – Centro de Referência em Assistência Social, Posto de Saúde, Escola, Assistente Social e Empresa de Ônibus; mercado são os concorrentes e parceiros: Redes Sociais; e, na parte de comunidade, estão relacionados os legisladores e espectadores: Prefeitura e Políticos.

Figura 9 – Diagrama de Partes Interessadas



Fonte: Autoria Própria

Ainda como resultado da primeira OSp, tem-se o Quadro de Avaliação produzido (Quadro 9).

Quadro 9 – Quadro de Avaliação

Partes Interessadas	Problemas e Questões	Ideias e Soluções
Contribuição <ul style="list-style-type: none"> • Cidadão • Comerciante Local • Morador novo 	1. Como será realizado o registro de solicitações para a Prefeitura? 2. De que modo ajuda um morador novo no bairro?	1. Sistema vinculado com a Prefeitura ou um cidadão protocola as todas as solicitações 2. Ajuda como forma de um guia para o cidadão que não conhece o bairro, visto que terá dados das instituições públicas e do comércio local

Continua

Conclusão

Partes Interessadas	Problemas e Questões	Ideias e Soluções
Fonte <ul style="list-style-type: none"> • CRAS • Posto de Saúde • Escola • Assistente Social • Empresa de Ônibus 	1. Quem atualiza informações sobre as instituições públicas	1. Próprio funcionário ou um cidadão se informa e atualiza os dados
Mercado <ul style="list-style-type: none"> • Redes Sociais 	1. Qual a vantagem do software com relação a publicar em redes sociais? 2. Qual a vantagem de registrar solicitação na aplicação?	1. Vantagem da aplicação ser voltada para o público da própria região 2. Redução de burocracia, podendo simplesmente registrar a solicitação na aplicação, sem precisar ir na Prefeitura presencialmente
Comunidade <ul style="list-style-type: none"> • Prefeitura • Políticos 	1. Qual a vantagem para a Prefeitura?	1. Centralização das solicitações de uma determinada região

Fonte: Autoria Própria

Com a produção desses artefatos, foi possível ter uma ideia geral do que o sistema deveria fazer, além de surgirem algumas questões que deveriam ser discutidas, como, por exemplo: Como seria a relação entre as solicitações e seu protocolo na Prefeitura? Se definir uma pessoa para fazer isso, não estaríamos imputando uma responsabilidade a ela arbitrariamente?

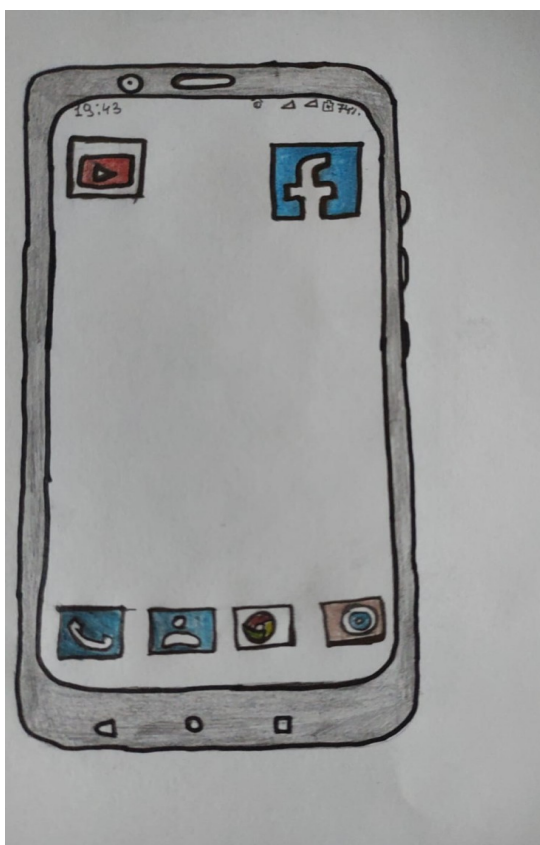
A partir desses questionamentos, decidiu-se, em conjunto com a turma, pela remoção da funcionalidade de registro de solicitações ao poder público, de modo que as funcionalidades que a aplicação deve abranger sejam: publicar, editar e buscar itens para doação; manter dados de funcionamento de instituições públicas locais (Posto de Saúde, CRAS e Escola); manter os horários, itinerários e linhas de ônibus coletivo na região e publicação de avisos em geral para a comunidade.

Paralelamente à execução das OSp, foi elaborado, pelo pesquisador, a Escada Semiótica (APÊNDICE G), que é um artefato que visa esclarecer a relação entre os aspectos técnicos e humanos da aplicação. Com o artefato produzido, este foi validado com a turma. Este artefato ajudou com o levantamento desde questões

relacionadas à infraestrutura quanto a questões relacionadas aos valores humanos e expectativas dos usuários.

A atividade *Mockup* teve duas rodadas de prototipação em papel. Em seu primeiro ciclo de execução, foram produzidos 8 protótipos em papel (APÊNDICE H). A Figura 10 apresenta um exemplo de protótipo produzido.

Figura 10 – Protótipo em Papel

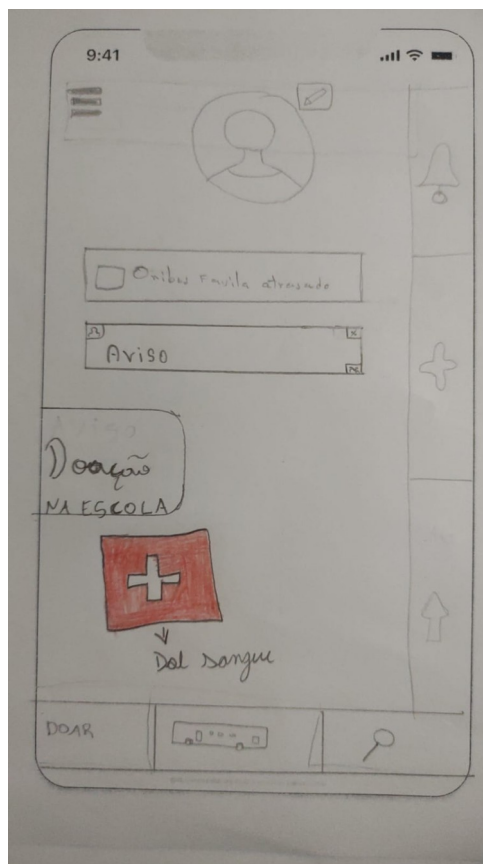


Fonte: Acervo de Pesquisa

Nesse protótipo, observam-se elementos de interface comuns em vários *smartphones*, sem trazer contribuições relacionados ao tema do aplicativo em desenvolvimento. Esses resultados inesperados podem ter sido causados pela falta de experiência deste pesquisador ou alguma falha na comunicação.

Na segunda iteração, foram gerados 9 protótipos (APÊNDICE I). Estes possuem muitos elementos de interface e ideias bem claras do que é solicitado com relação a *layout* e funcionalidades. Na Figura 11, tem-se um exemplo de protótipo produzido.

Figura 11 – Protótipo em papel 2

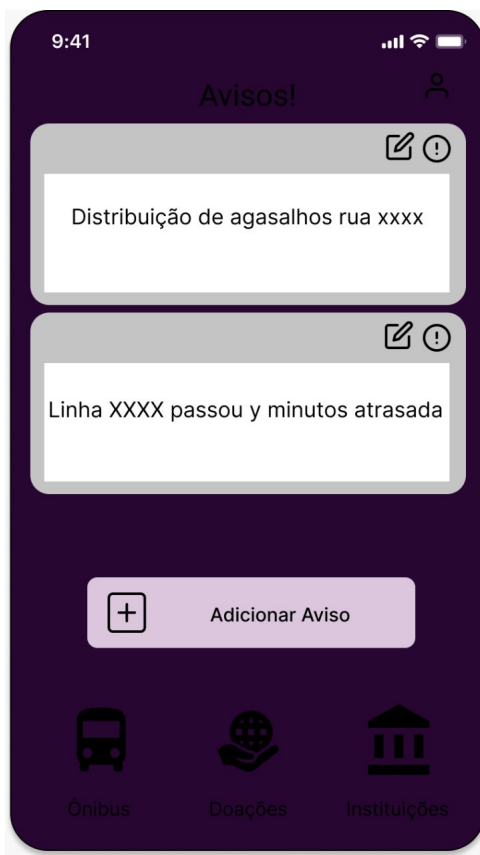


Fonte: Acervo de Pesquisa

Notam-se, nesse protótipo, algumas funcionalidades expressas e alguns padrões de interface, como o menu na parte inferior da tela.

Após as atividades de prototipação em papel, foi produzido pelo pesquisador o primeiro *mock-up* (APÊNDICE K) com o auxílio da ferramenta Figma. A Figura 12 ilustra a tela inicial do aplicativo, após a autenticação do usuário.

Figura 12 – Mock-up 1 tela início

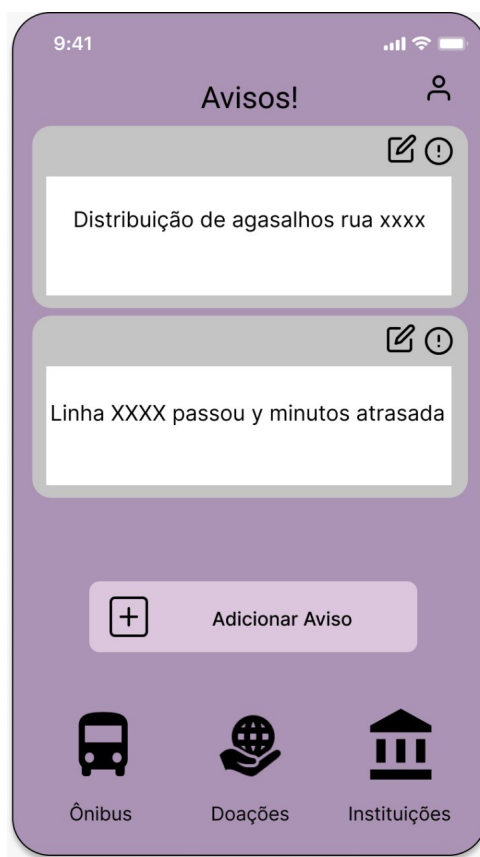


Fonte: Autoria Própria

Durante a Avaliação Heurística de Usabilidade, a qual esses protótipos foram submetidos, 8 erros foram pontuados, sendo a heurística número 1, que trata da visibilidade e do estado do sistema, a mais citada. Com relação ao grau de severidade, foi encontrado apenas 1 problema considerado uma catástrofe de usabilidade (Grau 4), pois não tinha em todas as telas a opção de *logout*, com o usuário tendo que ir até a tela inicial para poder encerrar a sua sessão no aplicativo. Destaca-se, ainda, que 4 problemas foram mapeados como grau de severidade 3, portanto problemas de usabilidade grandes, relacionados ao contraste de cores, falta de opção ajuda e mensagens de confirmação.

A partir dos resultados e *feedbacks* da avaliação, foi produzida a segunda versão dos *mockups* (APÊNDICE M). A Figura 13 ilustra a tela inicial do aplicativo.

Figura 13 – Mock-up 2



Fonte: Autoria Própria

O Mockup 2 veio a ser validado com a turma com auxílio da avaliação SUS (APÊNDICE N). Nesta validação foi definido que a cor tema do aplicativo deveria ser roxo escuro, visto que houve reclamação pela tonalidade mais rosada do segundo *mockup* (Figura 13).

Também foi aplicado um formulário de questões com a Matriz de Eisenhower (APÊNDICE O), também conhecida como matriz de urgência e importância para a priorização dos requisitos.

O Quadro 10 apresenta como foram priorizados os requisitos pela turma.

Quadro 10 – Matriz de Urgência/Importância

3 – Importante, mas não urgente RF02 – Publicar aviso para a comunidade RF03 – Consultar informações, horários de atendimento e avisos gerais RF07 – Gerenciar avisos RF08 – Gerenciar informações, dados de horários de atendimento e avisos	1 – Urgente e Importante RF06 – Autenticar usuário RF04 – Consultar informações sobre horários de ônibus RF09 – Gerenciar horários de ônibus RF01 – Publicar item para doação RF05 – Gerenciar item para doação
4 – Nem urgente, nem importante RF10 – Denunciar conteúdo impróprio	2 – Urgente, mas não importante

Fonte: Autoria Própria

Pode-se notar o maior interesse com relação às doações e com relação às informações dos horários dos ônibus, enquanto que o requisito que foi considerado menos importante e urgente foi o reporte de conteúdo indevido. Essas decisões são interessantes do ponto de vista ético, pois os reportes de conteúdo indevido é um aspecto ético importante quanto ao uso. No entanto, nota-se uma maior priorização em requisitos que impactam diretamente os usuários.

Ao final da etapa de Engenharia de Requisitos, chegou-se a um documento de requisitos (APÊNDICE J), validado com a turma, que especifica o que deveria ser feito, tendo no seu corpo a lista de requisitos funcionais, não funcionais e éticos.

4.3 Desenvolvimento Incremental

Durante o primeiro ciclo de desenvolvimento, foram implementadas as seguintes funcionalidades: autenticar-se, manter itens para doação e manter informações dos ônibus.

Na Figura 14, tem-se a tela de *splash screen* apresentada ao usuário quando o aplicativo está fazendo o carregamento de algum conteúdo.

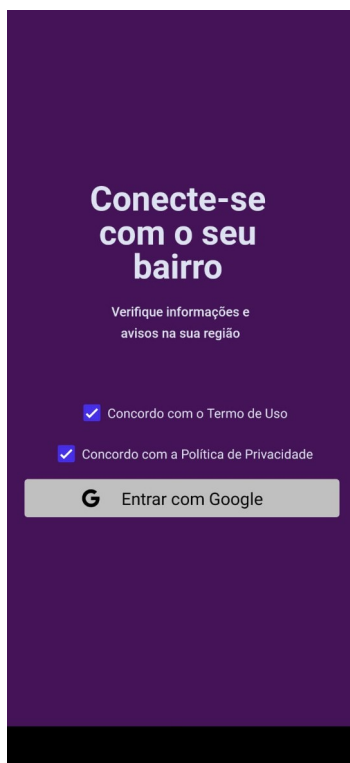
Figura 14 – *Splash Screen*



Fonte: Autoria Própria

Na Figura 15, está representada a tela de autenticação. Conforme definido durante a Engenharia de Requisitos, faz-se a autenticação utilizando a conta Google.

Figura 15 – Tela de autenticação



Fonte: Autoria Própria

Na Figura 16, tem-se a tela inicial do aplicativo que apresenta um menu na parte inferior com as opções Ônibus, Doações e Instituições.

Figura 16 – Tela Início

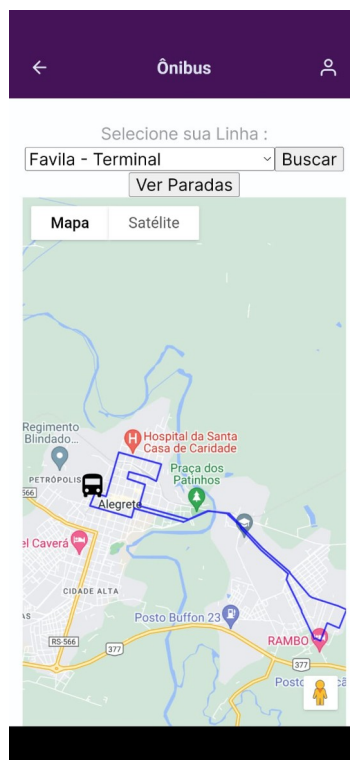


Fonte: Autoria Própria

Na parte superior central, há uma legenda, situando o usuário de sua localização no aplicativo e, na direita, o ícone de perfil com as suas opções. No corpo da tela, apresentam-se os avisos publicados no dia, seguidos de um botão com a palavra adicionar, que torna possível a publicação de novos avisos.

Na Figura 17, tem-se a tela com a funcionalidade relacionada aos ônibus. Nesta tela é possível selecionar a linha que deseja visualizar, então o ônibus aparecerá no mapa com o seu respectivo trajeto.

Figura 17 – Tela Ônibus

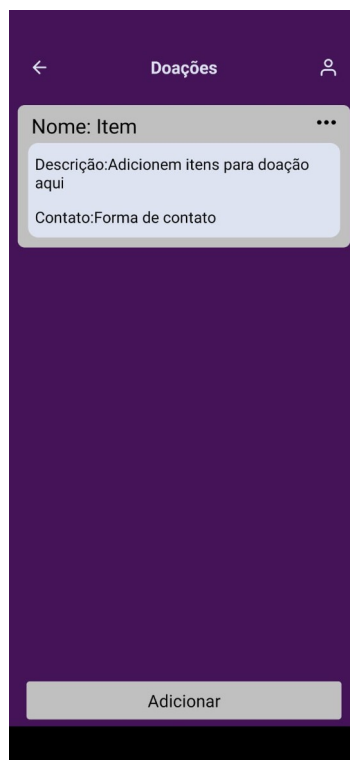


Fonte: Autoria Própria

Essa funcionalidade usa como fonte de dados a aplicação desenvolvida e disponibilizada pela empresa de transporte coletivo da cidade. O aplicativo desenvolvido neste TCC torna visual as informações disponibilizadas. Pode-se considerá-la de alto impacto na sociedade, devido ao público-alvo ser, em sua maioria, usuários de serviço de transporte coletivo. Levando em conta a pequena quantia de ônibus que atende a região, saber onde um ônibus está, se está atrasado ou adiantado, é de grande utilidade. Com relação à acessibilidade, contudo, essa tela não é compatível com a utilização de leitores de tela, o que precisa ser tratado em versão futura do software para que alcance mais usuários e atenda adequadamente ao RNF 05 e ao RE 02.

Na Figura 18, apresenta-se a tela de doações, na qual é possível ter acesso aos itens publicados para doação, com seu nome, descrição e forma de contato. Logo abaixo, está um botão de adicionar item para doação.

Figura 18 – Tela Doações

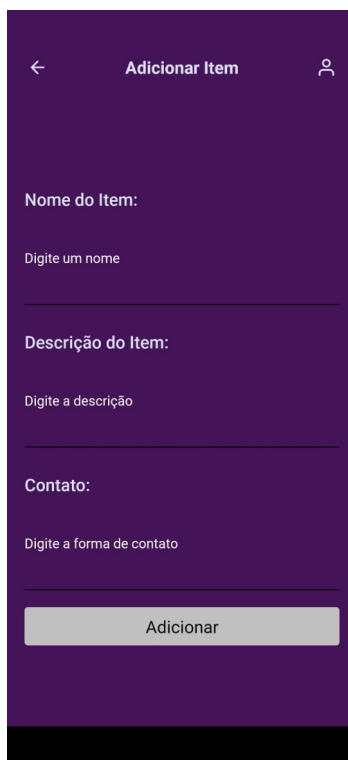


Fonte: Autoria Própria

Cada publicação de doação fica pública para os demais usuários pelo prazo de quinze dias, de modo que a informação de contato, que é um dado sensível do usuário, não fica exposto por tempo indeterminado.

Na Figura 19, tem-se a tela adicionar item para doação, onde é possível realizar a publicação de um item para doação.

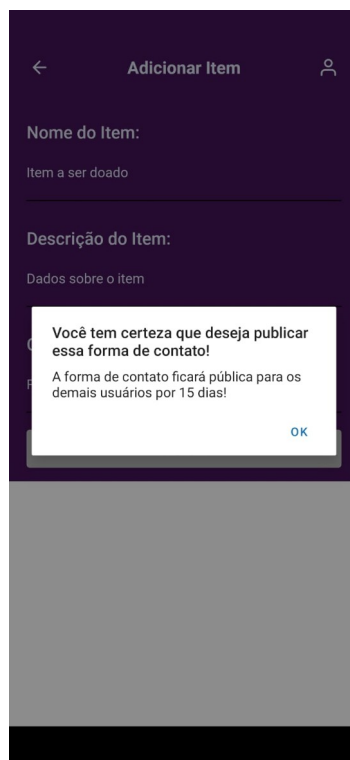
Figura 19 – Tela Adicionar Item



Fonte: Autoria Própria

Nessa tela, o usuário pode o publicar item a ser doado informando o nome do item, sua descrição e a forma de contato disponibilizada pelo doador. Ao clicar em adicionar tem-se uma mensagem de alerta apresentada na Figura 20.

Figura 20 – Mensagem de confirmação



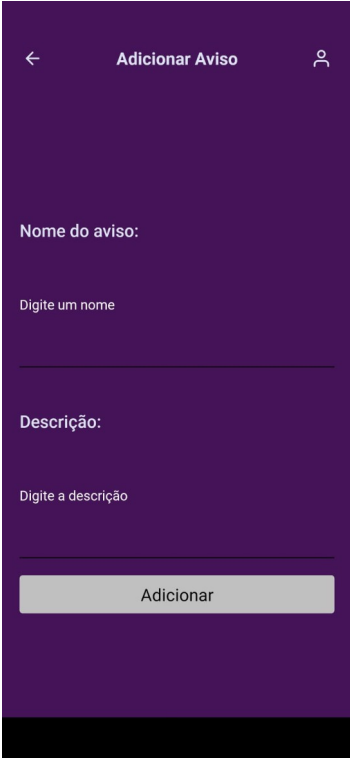
Fonte: Autoria Própria

Essa mensagem de confirmação tem como objetivo informar que a forma de contato ficará visível junto ao item de doação pelo prazo de quinze dias, solicitando a confirmação do usuário antes da publicação. Observa-se, ainda, que é facultado ao usuário remover um item publicado para doação a qualquer momento.

A seguir serão apresentadas as funcionalidades desenvolvidas durante o segundo ciclo de desenvolvimento: manter avisos para a comunidade, manter informações sobre as instituições e reporte de conteúdo inapropriado.

Na Figura 21, está representada a tela Adicionar Aviso, onde é possível realizar a publicação de um aviso para a comunidade.

Figura 21 – Tela Adicionar Aviso

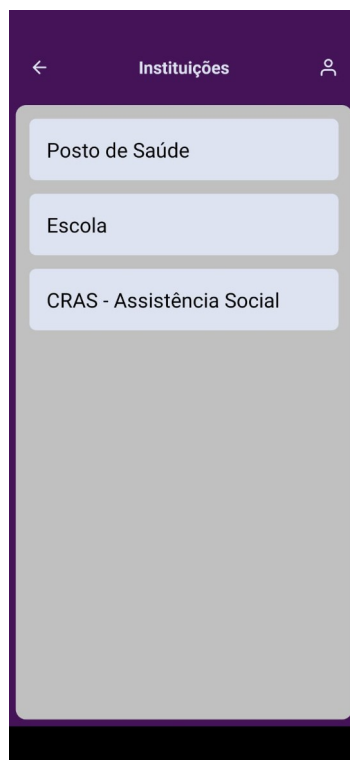


Fonte: Autoria Própria

Para a publicação de aviso é necessário preencher os campos “Nome do aviso” e “Descrição”.

Na Figura 22, está a tela relacionada a informações das instituições, onde é possível realizar a seleção do tipo de instituição que deseja visualizar, com as opções Posto de Saúde, Escola e CRAS.

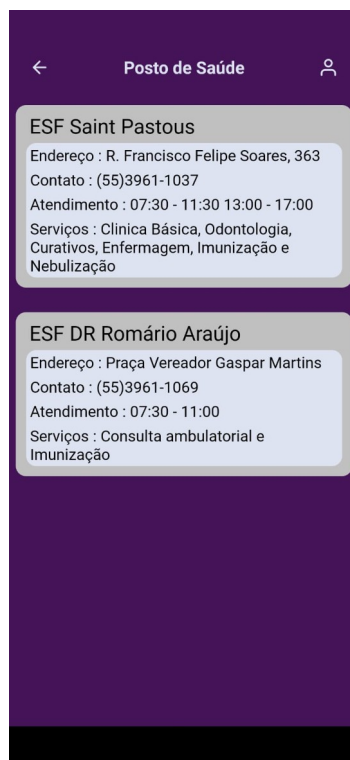
Figura 22 – Tela Instituições



Fonte: Autoria Própria

Na Figura 23, apresenta-se a tela onde é possível ver as informações detalhadas dos postos de saúde, com nome, endereço, telefone de contato, horário de atendimento e os serviços prestados.

Figura 23 – Tela de informações das instituições

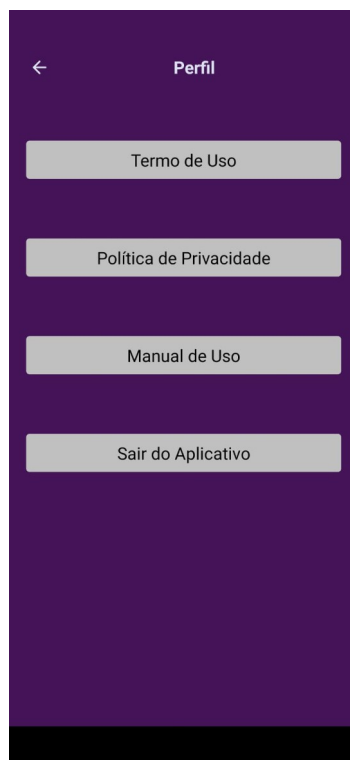


Fonte: Autoria Própria

As informações disponibilizadas foram levantadas por este pesquisador e podem ser mantidas por outro usuário que tenha perfil de moderador no aplicativo.

Na Figura 24, a seguir, ilustra a tela de perfil, onde é possível acessar as funções de usuário.

Figura 24 – Tela Perfil

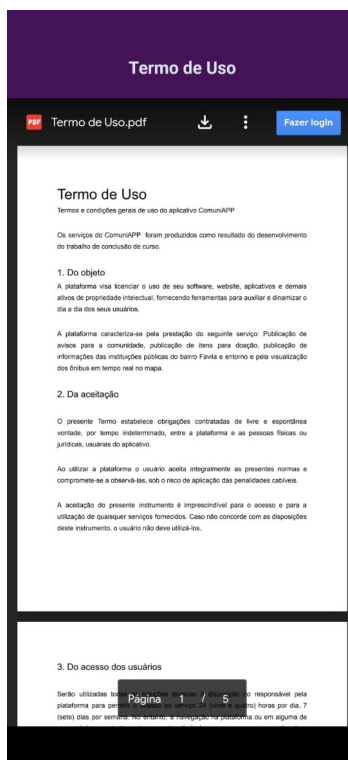


Fonte: Autoria Própria

Nesta tela, é possível ver o Termo de Uso, Política de Privacidade e Manual de Uso. Também tem um botão para sair da aplicação.

Na Figura 25, apresenta-se a tela onde é possível consultar o termo de uso (APÊNDICE R).

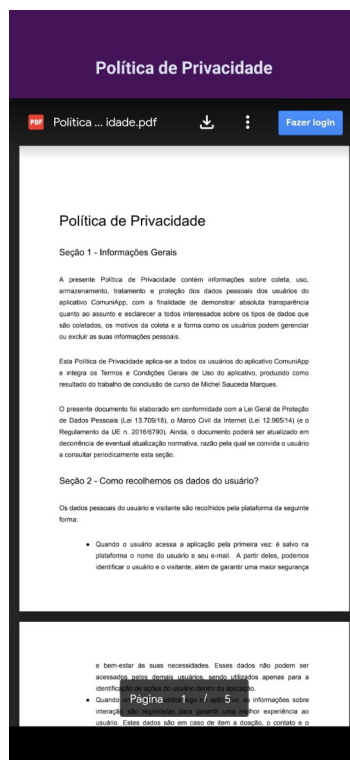
Figura 25 – Termo de Uso



Fonte: Autoria Própria

Na Figura 26, apresenta-se a tela onde é possível consultar as Políticas de Privacidade (APÊNDICE S).

Figura 26 – Políticas de Privacidade



Fonte: Autoria Própria

Sobre a Validação do software, realizada com apoio de testes funcionais e de aceitação, a turma se mostrou bastante satisfeita com os resultados. O APÊNDICE T organiza as respostas dessa atividade.

O escopo relacionado aos ônibus, instituições, avisos e requisitos éticos obteve resposta máxima quanto à satisfação dos usuários, enquanto que a autenticação e as funcionalidades relacionadas às doações tiveram algumas avaliações “concordo parcialmente”. Isso ocorre, pois é exigido ao usuário fazer login a cada vez que abrir o aplicativo. Quanto às doações, os estudantes chamaram atenção para a falta da imagem do item de doação que não foi implementado devido a questões técnicas e de prazos.

A ética se mostrou presente no produto, na forma da funcionalidade de realizar reporte de conteúdo, visto que o reporte é para ser usado quando o conteúdo em questão estiver errado, for inapropriado ou indevido. Na criptografia dos dados pessoais, no registro de data e hora do aceite do termo de uso e das políticas de privacidade e na possibilidade do usuário excluir os seus dados da aplicação. Aparece também no tempo de exposição do dado de contato quando

publicado um item para doação. Além disso, está presente no Termo de uso e nas Políticas de privacidade. Essas características emergiram a partir das provocações às partes interessadas sobre questões éticas no desenvolvimento de software, começando pela delimitação de requisitos éticos.

4.4 Considerações finais do capítulo

O software desenvolvido é socialmente responsável em seu objetivo, pois tem alto potencial de impacto no dia a dia de seus usuários, ajuda na melhoria da comunidade a partir da observação de diretrizes éticas.

Com o *app*, uma pessoa pode auxiliar outra com a doação de algum item; seus usuários, que dependem de ônibus para se deslocar, podem monitorar a localização de determinada linha; pode-se publicar avisos que ficarão visíveis para os demais usuários; além disso, o acesso à informações de instituições como escolas, postos de saúde e centros de assistência social é facilitado.

5 CONSIDERAÇÕES FINAIS

Aspectos éticos estiveram presentes durante todo o processo de desenvolvimento de software em tela neste TCC, visto que ocorreram diversos momentos onde foi necessário tomar decisões que remetem ao impacto social do software em perspectiva. Destaca-se a participação dos alunos durante todas as etapas desse processo e seu envolvimento nas tomadas de decisão, desde o início do processo até chegar no resultado final. A ideia de produzir um software socialmente responsável resultou em um software que reflete muita consciência social de suas partes interessadas, onde valores humanos estão presentes especialmente nas funcionalidades relacionadas às doações e aos avisos.

Com relação às camadas da Engenharia de Software, requisitos éticos e atributos de qualidade, sobretudo Funcionalidade e Usabilidade, serviram como base para o desenvolvimento do aplicativo ComuniApp. No processo de Engenharia de Requisitos, esses requisitos e atributos foram definidos, orientando o processo de Desenvolvimento Incremental. Fez-se uso de diversos métodos eticamente orientados com o apoio de artefatos para que a ética estivesse sempre em foco durante as atividades. Destaca-se o uso de Oficinas Semio-participativas, onde foram produzidos artefatos importantes na construção de um entendimento mútuo no grupo. Com relação às ferramentas, tanto sua escolha quanto a sua utilização envolvem decisões éticas, visto que o uso de software ilegal pode ser considerado aético ou anti-ético. Na metodologia deste TCC, de modo geral, a ética esteve presente desde o início, já na etapa de Organização da Pesquisa, com a adoção de termos de consentimento livre e esclarecido para os professores colaboradores, alunos e responsáveis.

No produto, a ética e a responsabilidade social são refletidas em seus requisitos, como, por exemplo, o reporte de conteúdo inapropriado, que surgiu durante uma das últimas tarefas da etapa de Engenharia de Requisitos – fruto de uma conversa sobre os aspectos éticos relacionados ao uso da aplicação. Já a visualização da rota do ônibus no mapa pode ser considerada de grande impacto social, visto que as pessoas que moram em bairros mais distantes e dependem dos poucos ônibus que têm na cidade podem se beneficiar de forma a evitar atrasos ao ir para a parada. Outro exemplo é a acessibilidade como requisito não funcional, que surgiu a partir da fala de um aluno, indicando uma preocupação com o alcance da

aplicação. Além de observar aspectos éticos no produto, é importante destacar que o comportamento ético do usuário é de suma importância para que o aplicativo alcance seu propósito.

Os objetivos deste trabalho foram todos atingidos: o desenvolvimento de software socialmente responsável é caracterizado; processos, métodos, ferramentas e artefatos de Engenharia de Software voltados ao desenvolvimento socialmente responsável de software foram identificados e utilizados; e um software socialmente responsável foi produzido com alto potencial de benefício para a sociedade. Além disso, destacam-se algumas lições aprendidas, como: a importância de uma comunicação clara e objetiva para o alinhamento de expectativas e para que o objetivo de determinada atividade seja alcançado; a relevância de participação do usuário final para que possam apresentar seu ponto de vista e para que suas expectativas sejam alcançadas; a necessidade de a acessibilidade ser sistematizada desde a Engenharia de Requisitos para que possa ser efetivamente contemplada no produto final; questões éticas devem ser abordadas durante todo o ciclo de vida de um software, inclusive em seu uso e evolução.

Entre as principais contribuições deste TCC estão: o produto gerado, constituído de duas aplicações com potencial significativo de impacto social - uma que mantém os dados e outra um aplicativo móvel para a publicação de itens para a doação, avisos para a comunidade, consulta da posição de ônibus no mapa da cidade em tempo real e informações das instituições públicas locais; e as reflexões sobre ética na Engenharia de Software, considerando atributos de qualidade, processo, métodos, ferramentas e pessoas. Finalmente, pode-se destacar que foi muito benéfico, tanto para o desenvolvedor quanto para a turma que colaborou com este TCC, a participação nesse processo de desenvolvimento de software socialmente responsável.

Quanto às limitações deste trabalho, é importante mencionar que a funcionalidade relacionada aos ônibus depende da aplicação da empresa de transporte coletivo, de modo que qualquer alteração realizada pela empresa impactará diretamente no produto deste TCC. Além disso, embora o produto atual possa ser utilizado, não está suficientemente maduro para ser amplamente difundido junto à população em geral, devido à necessidade de atribuição de papel de moderador para as publicações – o que atualmente é realizado pelo próprio pesquisador; apresentação de informação de contato do usuário junto a uma

doação. Ademais, a atualização de dados das instituições atualmente também deve ser realizada pelo pesquisador. Pode-se citar, ainda, que a promoção da acessibilidade não foi bem sistematizada no desenvolvimento do aplicativo, mas deve ser considerada para a publicação de uma versão que alcance mais usuários, além dos atuais.

Como trabalhos futuros, tem-se a possibilidade de escalar as aplicações para que sejam capazes de atender a cidade como um todo. Contudo, por mais que sua arquitetura seja pensada de forma escalável, os custos de manutenção devem ser considerados. Como a parte de login foi desenvolvida de forma a possibilitar a adição da autenticação com outras redes sociais, implementá-las futuramente. Ademais, na condução de um processo de evolução, aspectos éticos também devem ser observados, por exemplo, mediando as doações em mecanismo de comunicação próprio, evitando a exposição do dado de contato. Finalmente, pode ser desenvolvido um *checklist* que auxilie a contemplar aspectos éticos na definição de qualidade, processos, métodos e ferramentas no desenvolvimento de *software* com a participação das partes interessadas nas tomadas de decisão.

REFERÊNCIAS

ACM/IEEE, 2015. Software Engineering 2014, **Curriculum Guidelines for Undergraduate Degree Programs in Software Engineering, A Volume of the Computing Curricula Series**. Disponível em:

<https://www.acm.org/binaries/content/assets/education/se2014.pdf> Acesso em: 02/01/2022.

ALAMI, A.; LEAVITT COHN, M.; WASOWSKI, A. Why Does Code Review Work for Open Source Software Communities?. *In: INTERNATIONAL CONFERENCE ON SOFTWARE ENGINEERING*, 41., 2019, Montreal. **Proceedings** [...]. New Jersey: IEEE, 2019. p. 1073-1083, doi: 10.1109/ICSE.2019.00111.

ALIDOOSTI, R.. Ethics-driven Software Architecture Decision Making. *In: INTERNATIONAL CONFERENCE ON SOFTWARE ARCHITECTURA COMPANION*, 18., 2021, Stuttgart. **Proceedings** [...] New Jersey: IEEE, 2021. p. 90-91, doi: 10.1109/ICSA-C52384.2021.00022.

BADAMPUDI, Deepika. Reporting Ethics Considerations in Software Engineering Publications. *In: INTERNATIONAL SYMPOSIUM ON EMPIRICAL SOFTWARE ENGINEERING AND MEASUREMENT*, 2017, Toronto. **Proceedings** [...]. New Jersey: IEEE, 2017. doi: 10.1109/ESEM.2017.32

BARANAUSKAS, M. C. C. O Modelo Semioparticipativo de Design. *In: BARANAUSKAS, M. C. C.; MARTINS, M. C.; VALENTE, J. A. **Codesign de redes digitais: Tecnologia e educação a serviço da inclusão social***. Penso Editora, 2013.

BARBOSA, S.; SILVA, B. **Interação humano-computador**. Elsevier Brasil, 2010.

BENEDICTO, S. *et al.* Surgimento e evolução da responsabilidade social: Uma reflexão teórico-analítica. *In: ENCONTRO NACIONAL DE ENGENHARIA DE PRODUÇÃO*, 28., 2008, Rio de Janeiro. **Anais** [...]. São José dos Campos: ABEPRO, 2008.

BECK, K. *et al.* **Manifesto for Agile Software Development**. Disponível em: <http://agilemanifesto.org/> Acesso em 10/07/2022.

BORG, M. *et al.* Exploring the Assessment List for Trustworthy AI in the Context of Advanced Driver-Assistance Systems. *In: INTERNATIONAL WORKSHOP ON ETHICS IN SOFTWARE ENGINEERING RESEARCH AND PRACTICE*, 2., 2021, Madrid. **Proceedings** [...]. New Jersey: IEEE, 2021. p. 5-12. DOI: 10.1109/SEthics52569.2021.00009.

CARVALHO, Ariadne Maria Brito Rizzoni; CHIOSSI, Thelma Cecília dos Santos. **Introdução à Engenharia de Software**. 1. Ed. São Paulo: Editora da UNICAMP, 2001.

CEMILOGLU, D. *et al.* Towards Ethical Requirements for Addictive Technology: The Case of Online Gambling. *In: WORKSHOP ON ETHICS IN REQUIREMENTS*

ENGINEERING RESEARCH AND PRACTICE, 1., 2020, Zurich. **Proceedings** [...]. New Jersey: IEEE, 2020. p. 1-10. DOI: 10.1109/REthics51204.2020.00007.

CHAKRABORTY *et al.* Fairway: A Way to Build Fair ML Software. *In: ESEC/FSE*, 28., 2020, Sacramento. **Proceedings** [...]. New York: ACM Digital Library, 2020.

COHN, M. **User stories applied**: For agile software development. [S.l.]: Addison-Wesley Professional, 2004.

DE, Suman. A Novel Perspective to Threat Modelling using Design Thinking and Agile Principles. *In: INTERNATIONAL CONFERENCE ON PARALLEL, DISTRIBUTED AND GRID COMPUTING*, 16., 2020, Wakhnaghat. **Proceedings** [...]. New Jersey: IEEE, 2020. p. 31-35. DOI: 10.1109/PDGC50313.2020.9315844.

DEKHTYAR *et al.* From RE Cares to SE Cares: Software Engineering for Social Good, One Venue at a Time. *In: INTERNATIONAL CONFERENCE ON SOFTWARE ENGINEERING: SOFTWARE ENGINEERING IN SOCIETY TRACK*, 42., 2020, Seoul. **Proceedings** [...]. New Jersey: IEEE, 2020.

EVANGELOU, S. M.; STAMOULAKATOU, G.; XENOS, M.. A Serious Game for Mobile Phones used in a Software Engineering Course: Usability Evaluation and Educational Effectiveness. *In: GLOBAL ENGINEERING EDUCATION CONFERENCE*, 2021, Tunis. **Proceedings** [...]. New Jersey: IEEE, 2021. p. 219-225. DOI: 10.1109/EDUCON46332.2021.9453987.

FREITAS, D. S., **Desenvolvimento de um software mobile com crianças em domínio educacional**: Uma perspectiva de engenharia de software. Monografia (Bacharelado) – Curso de Engenharia de Software, Universidade Federal do Pampa, Alegrete, 2021.

GHASEMPOURI, S. A.; TORK LADANI, B.. A Formal Model for Security Analysis of Trust and Reputation systems. *In: INTERNATIONAL ISC (IRANIAN SOCIETY OF CRYPTOLOGY) CONFERENCE ON INFORMATION SECURITY AND CRYPTOLOGY*, 14., 2017, Isfahan. **Proceedings** [...]. New Jersey: IEEE, 2017. p. 13-18. DOI: 10.1109/ISCISC.2017.8488379.

GRANT, E. S.. Principles of Ethical Consideration in Safety Critical Software Systems Development. *In: TECHNOLOGY INNOVATION MANAGEMENT AND ENGINEERING SCIENCE INTERNATIONAL CONFERENCE*, 4., 2019, Bangkok. **Proceedings** [...]. New Jersey: IEEE, 2019. p. 1-5. DOI: 10.1109/TIMES-iCON47539.2019.9024377.

ISO – INTERNATIONAL ORGANIZATION FOR STANDARDIZATION. ISO/IEC 25010:2011 **Systems and software engineering**: Systems and software Quality Requirements and Evaluation (SQuARE) — System and software quality models. 1 ed., 2011. 34 p

JINGDONG, Jia; JIAQI Xin. Integration of Ethics Issues into Software Engineering Management Education. *In: TURING CELEBRATION CONFERENCE*, 18., 2018, Shanghai. **Proceedings** [...]. New York: ACM Digital Library, 2018.

KAMMULLER, F.; J. C., Augusto; JONES, S.. Security and privacy requirements engineering for human centric IoT systems using eFRIEND and Isabelle. INTERNATIONAL CONFERENCE ON SOFTWARE ENGINEERING RESEARCH, MANAGEMENT AND APPLICATIONS, 15., 2017, London. **Proceedings** [...]. New Jersey: IEEE, 2017. p. 401-406. DOI: 10.1109/SERA.2017.7965758.

KITCHENHAM, Bárbara. **Procedimentos para realizar revisões sistemáticas**. Keele, Reino Unido, Keele University, v. 33, n. 2004, p. 1-26, 2004.

LIU, Kecheng. 2000. **Semiotics in Information Systems Engineering**. UK: Cambridge University Press.

MILOSEVIC, Z.. Ethics in Digital Health: A Deontic Accountability Framework. *In*: INTERNATIONAL ENTERPRISE DISTRIBUTED OBJECT COMPUTING CONFERENCE, 23., 2019, Paris. **Proceedings** [...]. New Jersey: IEEE, 2019. p. 105-111. DOI: 10.1109/EDOC.2019.00022.

MULLER, M. J.; HASLWANTER, J. H.; DAYTON, T.. **Participatory Practices in the Software Lifecycle**. In Handbook of Human-Computer Interaction, 2nd Rev. ed., ed. M. G. Helander, T. K. Landauer and P. V. Prabhu, 255-297. The Netherlands: Elsevier Science. 1997.

MUSTAFA, N. A. *et al.* Modeling the Structural of Intention and Attitude of Muslim Software Developer to Use Islamic Work Ethic in Developing Islamic Content Mobile Apps. *In*: INTERNATIONAL CONFERENCE ON INTERACTIVE DIGITAL MEDIA, 6., 2020, Bandung. **Proceedings** [...]. New Jersey: IEEE, 2021. p. 1-6. DOI: 10.1109/ICIDM51048.2020.9339606.

NAGARIA, J.; SADATH, L.; AHMED, S.. Agile Implementation-A milestone for Academics using Software Engineering Industry Practices. *In*: ADVANCES IN SCIENCE AND ENGINEERING TECHNOLOGY INTERNATIONAL CONFERENCE, 2019, Dubai. **Proceedings** [...]. New Jersey: IEEE, 2019. p. 1-6. DOI: 10.1109/ICASET.2019.8714575.

PEREIRA, L. B., **Um método de desenvolvimento de software com idosos: Integrando Design Participativo à Engenharia de Software**. Monografia (Bacharelado) – Curso de Engenharia de Software, Universidade Federal do Pampa, Alegrete, 2021.

PRESSMAN, Roger S.; MAXIM, Bruce R.. **Engenharia de software: uma abordagem profissional**. 8 ed. Porto Alegre: AMGH, 2016.

RIBEIRO, M. S. A evolução dos conceitos de responsabilidade social. *In*: CONGRESSO BRASILEIRO DE CUSTOS, 9., 2002, São Paulo. **Anais** [...]. São Leopoldo: ASSOCIAÇÃO BRASILEIRA DE CUSTOS, 2002.

ROGERS, Y., SHARP, H., PREECE, J. **Design de Interação: Além da Interação Humano-Computador**. 3. ed. Porto Alegre: Editora BOOKMAN, 2013.

SERBAN, A. *et al.* Practices for Engineering Trustworthy Machine Learning Applications. *In: WORKSHOP ON AI ENGINEERING – SOFTWARE ENGINEERING FOR AI*, 1., 2021, Madrid. **Proceedings** [...]. New Jersey: IEEE, 2019. p. 97-100. DOI: 10.1109/WAIN52551.2021.00021.

SHNEIDERMAN, Ben. Bridging the Gap Between Ethics and Practice: Guidelines for Reliable, Safe, and Trustworthy Human-centered AI Systems. **ACM Transactions on Interactive Intelligent Systems**, Vol. 10, No. 4, Artigo 26. Data de Publicação: Outubro de 2020

SOMMERVILLE, I. **Engenharia de Software**. Tradução: Ivan Bosnic; Kalinka G. de O. Gonçalves. 9. ed. São Paulo: Pearson Education, 2011. 521 p. Título original: Software Engineering. ISBN: 978-85-7936-108-1.

STRANDBERG, P. E.; FRASHERI, M.; ENOIU, E. P.. Ethical AI-Powered Regression Test Selection. *In: INTERNATIONAL CONFERENCE ON ARTIFICIAL INTELLIGENCE TESTING*, 2021, Oxford. **Proceedings** [...]. New Jersey: IEEE, 2021. p. 83-84. DOI: 10.1109/AITEST52744.2021.00025.

STRANDBERG, P. E.. Ethical Interviews in Software Engineering. *In: INTERNATIONAL SYMPOSIUM ON EMPIRICAL SOFTWARE ENGINEERING AND MEASUREMENT*, 2019, Porto de Galinhas. **Proceedings** [...]. New Jersey: IEEE, 2019. p. 1-11. DOI: 10.1109/ESEM.2019.8870192.

TERENCE, A. C. F.; FILHO, E. E. Abordagem quantitativa, qualitativa e a utilização da pesquisa-ação nos estudos organizacionais. *In: Encontro Nacional de Engenharia de Produção*, 26., 2006, Fortaleza. **Anais** [...]. São José do Campos: ABEPRO. Disponível em: http://www.abepro.org.br/biblioteca/enegep2006_tr540368_8017.pdf

VAKKURI, V.; KEMELL, K.; ABRAHAMSSON, P.. ECCOLA – a Method for Implementing Ethically Aligned AI Systems. *In: EUROMICRO CONFERENCE ON SOFTWARE ENGINEERING AND ADVANCED APPLICATIONS*, 46., 2020, Portoroz. **Proceedings** [...]. New Jersey: IEEE, 2020. p. 196-204. DOI: 10.1109/SEAA51224.2020.00043.

YILMAZ, L.; FRANCO-WATKINS, A.; KROECKER, T. S.. Coherence-driven Reflective Equilibrium model of ethical decision-making. *In: INTERNATIONAL MULTI-DISCIPLINARY CONFERENCE ON COGNITIVE METHODS IN SITUATION AWARENESS AND DECISION SUPPORT*, 2016, San Diego. **Proceedings** [...]. New Jersey: IEEE, 2016. p. 42-48. DOI: 10.1109/COGSIMA.2016.7497784.

YUAN, LING *et al.* Ethical leadership, leader-member exchange and voice behavior: Test of mediation and moderation processes. *In: INTERNATIONAL CONFERENCE ON MANAGEMENT ENGINEERING, SOFTWARE ENGINEERING AND SERVICE SCIENCES*, 2., 2018, Wuhan. **Proceedings** [...]. New York: ACM Digital Library, 2018. p. 33-42. ISBN: 978-1-4503-5431-8.

APÊNDICES

APÊNDICE A – Planilha de Avaliação Heurística

Problema e sua localização	Heurísticas (ex.: H1, H2, ..., H10)	Grau de Severidade

Heurísticas

H1 Visibilidade do estado do sistema

H2 Mapeamento entre o sistema e o mundo real

H3 Controle e liberdade do usuário

H4 Consistência e padrões

H5 Prevenção de erros

H6 Reconhecimento em vez de lembrar

H7 Flexibilidade e eficiência de uso

H8 Design estético e minimalista

H9 Suporte para o usuário reconhecer, diagnosticar e recuperar erros

H10 Ajuda e documentação

Graus de Severidade

0 – Eu não concordo que isto seja um problema de usabilidade

1 – Problema apenas cosmético: não precisa ser corrigido a menos que haja tempo extra disponível para o projeto

2 – Problema de usabilidade menor: deveria ser dada baixa prioridade para sua correção

3 – Problema de usabilidade maior: importante corrigir e, assim, deveria ser dada alta prioridade

4 – Catástrofe de usabilidade: imperativo corrigi-la

APÊNDICE B – Termos de Consentimento Livre e Esclarecido (Professora)

TERMO DE CONSENTIMENTO LIVRE E ESCLARECIDO

Título do Projeto: Uma abordagem socialmente responsável ao desenvolvimento de software

Orientadora: Amanda Meincke Melo

Pesquisador: Michel Saucedo Marques

Instituição: Universidade Federal do Pampa – UNIPAMPA

Telefone do pesquisador para contato: _____

Sr./Sr^a/Você está sendo convidado(a) a participar, de forma voluntária, em uma pesquisa intitulada “Desenvolvimento de Software Eticamente Alinhado”, que tem por objetivo conceber e desenvolver um software de maneira colaborativa e socialmente responsável com adolescentes.

Através deste documento e em qualquer momento do processo, o Sr./Sr^a/Você poderá solicitar esclarecimentos adicionais sobre o estudo em qualquer aspecto que desejar. Também será possível retirar seu consentimento ou interromper sua participação a qualquer momento, sem sofrer qualquer tipo de penalidade ou prejuízo.

Após os esclarecimentos a seguir, no caso de aceitar fazer parte deste estudo, deves indicar seu consentimento. Propõe-se, em colaboração com adolescentes, o desenvolvimento de uma tecnologia socialmente responsável, ou seja, que tem como objetivo contribuir com a melhoria da sociedade. Para isso, em um primeiro momento, será realizada uma entrevista que visa coletar dados acerca da infraestrutura da escola e viabilidade da execução do projeto. Em seguida, junto a professora, o pesquisador realizará a validação dos Termos de Consentimento Livre e Esclarecido a ser enviado para os responsáveis dos adolescentes.

O Sr./Sr.^a/Você não terá nenhum custo, nem receberá vantagem financeira pela sua participação no projeto. Possíveis gastos necessários (se houver) na pesquisa serão assumidos pelo pesquisador.

Alguns riscos inerentes à participação são eventuais desconfortos quanto a sentir-se obrigado(a) a colaborar ou a responder alguma questão. Entre os

benefícios estão a aprendizagem sobre tecnologias digitais, assim como o bem-estar ocasionado pela colaboração

Seu nome e identidade serão mantidos em sigilo e os dados da pesquisa serão armazenados pelo pesquisador responsável. Os resultados podem ser divulgados em publicações científicas e apresentará apenas resultados gerais, sem que haja informações pessoais dos participantes.

Nome do Participante

Michel Saucedá Marques

Alegrete, 24 de fevereiro de 2022

APÊNDICE C – Entrevista com Professora

Roteiro: Entrevista com Professora

Objetivos: apresentar o pesquisador, apresentar a proposta de trabalho, verificar a viabilidade de execução do projeto

Procedimentos:

1. Apresentar o pesquisador;
2. Apresentar a proposta de trabalho e seus objetivos;
3. Verificar viabilidade para a execução do projeto;
 1. Dados sobre a entrevistada:
 1. Nome completo:
 2. Idade:
 3. Escolaridade:
 4. Escola(s):
 5. Função na(s) escola(s):
 2. Na EEEM Waldemar Borges, com quantas turmas trabalha? Em quais horários?
 3. Quais turmas teriam disponibilidade para participar de práticas participativas para o desenvolvimento de *software*?
 4. Qual a disponibilidade de tempo dessas turmas para a participação? Em quais disciplinas? (ex.: artes, sociologia etc.) Ou apenas no contra-turno?
 5. Acredita que haveria interesse dos alunos em participar desse tipo de atividade?
 6. Qual a infraestrutura de informática na escola? Há disponibilidade de laboratório de informática? E rede de Internet sem fio?
 7. Há algum projeto interdisciplinar em desenvolvimento na escola ou temas transversais em discussão?
 8. Os professores adotam algum recurso digital em suas aulas?
 9. De acordo com as faixas etárias dos alunos, o que você recomendaria como abordagens para levantar ideias? (ex.: entrevistas, tempestade de ideias, desenhos etc.)

APÊNDICE D – Termos de Consentimento Livre e Esclarecido para Alunos e Responsáveis

TERMO DE CONSENTIMENTO LIVRE E ESCLARECIDO

Título do Projeto: Uma abordagem socialmente responsável ao desenvolvimento de software

Orientadora: Amanda Meincke Melo

Pesquisador: Michel Saucedá Marques

Instituição: Universidade Federal do Pampa – UNIPAMPA

Telefone do pesquisador para contato: _____

O/A filho/filha do Sr./Sr^a/Você está sendo convidado(a) a participar, de forma voluntária, em uma pesquisa intitulada “Desenvolvimento de Software Eticamente Alinhado”, que tem por objetivo conceber e desenvolver um software de maneira colaborativa e socialmente responsável com adolescentes.

Através deste documento e em qualquer momento do processo, o Sr./Sr^a/Você poderá solicitar esclarecimentos adicionais sobre o estudo em qualquer aspecto que desejar. Também será possível retirar seu consentimento ou interromper a participação de seu/sua filho/filha a qualquer momento, sem sofrer qualquer tipo de penalidade ou prejuízo.

Após os esclarecimentos a seguir, no caso de aceitar que seu/sua filho/filha faça parte do estudo, assine ao final deste documento, que está em duas vias. Uma delas é sua e a outra será arquivada pelo pesquisador responsável.

Propõe-se o desenvolvimento de uma tecnologia socialmente responsável, ou seja, que tem como objetivo contribuir com a melhoria da sociedade. Para isso, o seu/sua filho/filha terá a participação no processo de desenvolvimento, ajudando na definição do tema a ser abordado, na concepção do um software e em sua validação.

Alguns riscos inerentes à participação de seu/sua filho/filha são eventuais desconfortos quanto a sentir-se obrigado(a) a colaborar. Entre os benefícios a se destacar estão a satisfação ao participar de um projeto de software que busca a melhoria da sociedade, o sentimento de coautoria e a própria experimentação dos métodos de Engenharia de Software.

Para seu/sua filho/filha participar deste estudo, o(a) Sr./Sr.^a/Você não terá nenhum custo, nem receberá vantagem financeira pela sua participação no projeto. Possíveis gastos necessários (se houver) na pesquisa serão assumidos pelo pesquisador.

Seu nome e identidade serão mantidos em sigilo, assim como de seu/sua filho/filha e os dados da pesquisa serão armazenados pelo pesquisador responsável. Os resultados podem ser divulgados em publicações científicas e apresentará apenas resultados gerais, sem que haja informações pessoais dos participantes. A imagem de seu/sua filho/filha poderá ser publicada contanto que haja sua autorização explícita para fazê-lo.

Nome do(a) Participante da Pesquisa

Nome do Responsável

Assinatura do Responsável

Nome do Pesquisador

Assinatura do Pesquisador

Alegrete, Fevereiro de 2022

Universidade Federal do Pampa – UNIPAMPA**AUTORIZAÇÃO**

Eu, _____, portador (a) da cédula de identidade nº _____, responsável legal pelo (a) menor _____, portador (a) da cédula de identidade nº _____, autorizo o uso da imagem, da voz e conteúdos produzidos pelo(a) menor acima citado(a) durante o trabalho de conclusão de curso para fins didáticos, de pesquisa e de divulgação sem quaisquer ônus e restrições. Fica ainda autorizada, de livre e espontânea vontade, para os mesmos fins, a cessão de direitos da veiculação dos itens citados acima, não recebendo para tanto qualquer tipo de remuneração.

Alegrete, _____ de _____ de 2022.

APÊNDICE E – Termos de Consentimento Livre e Esclarecido para Alunos

TERMO DE CONSENTIMENTO LIVRE E ESCLARECIDO

Título do Projeto: Uma abordagem socialmente responsável ao desenvolvimento de software

Orientadora: Amanda Meincke Melo

Pesquisador: Michel Saucedá Marques

Instituição: Universidade Federal do Pampa – UNIPAMPA

Telefone do pesquisador para contato: _____

O/A Sr./Sr^a/Você está sendo convidado(a) a participar, de forma voluntária, em uma pesquisa intitulada “Desenvolvimento de Software Eticamente Alinhado”, que tem por objetivo conceber e desenvolver um software de maneira colaborativa e socialmente responsável com adolescentes.

Através deste documento e em qualquer momento do processo, o Sr./Sr^a/Você poderá solicitar esclarecimentos adicionais sobre o estudo em qualquer aspecto que desejar. Também será possível retirar seu consentimento ou interromper sua participação a qualquer momento, sem sofrer qualquer tipo de penalidade ou prejuízo.

Após os esclarecimentos a seguir, no caso de aceitar fazer parte do estudo, assine ao final deste documento, que está em duas vias. Uma delas é sua e a outra será arquivada pelo pesquisador responsável.

Propõe-se o desenvolvimento de uma tecnologia socialmente responsável, ou seja, que tem como objetivo contribuir com a melhoria da sociedade. Para isso, você terá a participação no processo de desenvolvimento, ajudando na definição do tema a ser abordado, na concepção do um software e em sua validação.

Alguns riscos inerentes à participação são eventuais desconfortos quanto a sentir-se obrigado(a) a colaborar. Entre os benefícios a se destacar estão a satisfação ao participar de um projeto de software que busca a melhoria da sociedade, o sentimento de coautoria e a própria experimentação dos métodos de Engenharia de Software.

Para participar deste estudo, o(a) Sr./Sr^a/Você não terá nenhum custo, nem receberá vantagem financeira pela sua participação no projeto. Possíveis gastos necessários (se houver) na pesquisa serão assumidos pelo pesquisador.

Seu nome e identidade serão mantidos em sigilo e os dados da pesquisa serão armazenados pelo pesquisador responsável. Os resultados podem ser divulgados em publicações científicas e apresentará apenas resultados gerais, sem que haja informações pessoais dos participantes. A sua imagem poderá ser publicada contanto que haja sua autorização explícita para fazê-lo..

Nome do(a) Participante da Pesquisa

Assinatura do(a) Participante da Pesquisa

Nome do Pesquisador

Assinatura do Pesquisador

Alegrete, Fevereiro de 2022

Universidade Federal do Pampa – UNIPAMPA**AUTORIZAÇÃO**

Eu, _____, portador (a) da cédula de identidade nº _____, portador (a) da cédula de identidade nº _____, autorizo o uso da imagem, da voz e conteúdos produzidos durante o trabalho de conclusão de curso para fins didáticos, de pesquisa e de divulgação sem quaisquer ônus e restrições. Fica ainda autorizada, de livre e espontânea vontade, para os mesmos fins, a cessão de direitos da veiculação dos itens citados acima, não recebendo para tanto qualquer tipo de remuneração.

Alegrete, _____ de _____ de 2022.

APÊNDICE F – Funcionalidades e Diretrizes Éticas

Funcionalidades

Data: 11/04/2022

- Fazer solicitações ao poder público(Ex.: Troca de lâmpada queimada, limpeza de rua, boeiramento, etc);
- Guia do bairro
 - Dados sobre atendimento de Posto de saúde, escola, CRAS, etc;
 - Linhas de ônibus disponíveis no bairro com horários e paradas;
 - Dados sobre o comércio local;
- Registro de bens para doações;
- ~~• Registro de denúncias;~~
- ~~• Banco de questões para uso didático;~~
- ~~• Leitor de QR code para validação de presenças;~~

Atributos de qualidade

Funcionalidade, Confiabilidade, Usabilidade, Eficiência, Manutenibilidade e Portabilidade

Entre os 6 atributos de qualidade externa e interna descritos pela ISO 9126, foi considerado como prioridade em consenso os atributos Usabilidade e Funcionalidade. Vale ressaltar que foi levantado questionamento com relação a Acessibilidade, de forma que será um atributo de qualidade que o software desenvolvido deve buscar satisfazer.

Requisitos éticos

Código de ética IEEE

Princípio 1: Engenheiros de software devem agir consistentemente com o interesse público

1.3 Aprovar software se tiver uma convicção bem-fundamentada de que seja seguro, satisfaça as especificações, passe nos testes apropriados e não diminua a qualidade de vida ou a privacidade, ou cause problemas ao meio ambiente.

1.7 Considerar questões de deficiência física, alocação de recursos, desvantagem econômica ou outros fatores que possam diminuir o acesso aos benefícios do software.

Princípio 2: Engenheiros de software devem agir da melhor maneira com seu cliente e empregador e de acordo com o interesse público.

2.2 Não usar; se souber, software obtido ou mantido de forma ilegal ou aética.

2.5 Manter privada qualquer informação confidencial obtida por intermédio do trabalho profissional, desde que essa confidencialidade seja consistente com o interesse público e com a legislação.

Princípio 3: Engenheiros de Software devem assegurar que seus produtos e as modificações relacionadas satisfaçam aos mais altos padrões profissionais.

3.5 Assegurar que o método adequado é usado por qualquer projeto em que trabalhe ou se proponha a trabalhar.

3.8 Assegurar que as especificações do software no qual trabalhe foram bem documentadas, satisfaçam aos requisitos dos usuários e tenham as aprovações apropriadas.

3.12 Trabalhar para desenvolver software e documentos relacionados que respeitem a privacidade daqueles que serão afetados por eles.

Princípio 4: Engenheiros de software devem manter a integridade e independência em seu julgamento profissional.

4.1 Considerar em todos os julgamentos técnicos as necessidades de apoiar e manter os valores humanos.

Princípio 5: Engenheiro de software em posições de gerência e liderança devem adotar e promover uma abordagem ética para a gestão do desenvolvimento e manutenção de software.

5.1 Assegurar boa gestão para qualquer projeto no qual trabalhe, incluindo procedimentos efetivos para a promoção da qualidade e redução dos riscos.

Princípios 6, 7 e 8 são relacionados a gestão, colegas e consigo mesmo.

APÊNDICE G – Escada Semiótica

Escada Semiótica

Data: 09/05/2022

A Escada Semiótica é um artefato que foi adaptado para organizar requisitos do sistema interativo nas 6 camadas de informação, bem como para organizar elementos de avaliação do sistema, considerando aspectos desde sua infraestrutura tecnológica (mundo físico, empírico, camada sintática), até o sistema de informação humano (camada semântica, pragmática e mundo social).

1 Mundo Social

Valores, compromissos estabelecidos e expectativas:

- Respeitar as diferentes situações de inclusão digital dos usuários
- O software desenvolvido deve ser amplamente acessível e usável
- O software desenvolvido deve funcionar corretamente
- Preservar a integridade física dos usuários (segurança/safety)
- Respeitar a privacidade dos usuários

2 Pragmática

Métodos de trabalho e intenções dos usuários:

- O software deve facilitar o acesso a informações do serviço público local
- O software deve prover um local para o registro de bens para doações
- O software deve promover para maior interação dentro da comunidade
- O software deve ser eficaz, eficiente e satisfazer seus usuários (usabilidade)
- O software deve ser utilizado em diferentes dispositivos
- O software deve funcionar adequadamente atendendo o propósito para o qual foi desenvolvido (funcionalidade)

3 Semântica

Significado atribuído ao sistema pelos usuários:

- Os usuários devem aprender facilmente a usar o software
- Os usuários devem conseguir lembrar facilmente como utilizar as funcionalidades apresentadas pelo software

- Os usuários devem se sentir seguros e satisfeitos ao utilizarem o software
- Os usuários devem conseguir utilizar o software em diferentes dispositivos de modo eficiente

4 Sintática

Ferramentas, métodos e estrutura da informação:

- Definir estratégias de divulgação do sistema para os demais cidadãos
- Realizar prototipação participativa, de modo que os colaboradores possam expressar como as tarefas/funcionalidades podem ser representadas no software
- Adotar padrões de projeto de interface de usuário
- Avaliar a usabilidade do software com apoio de métodos como Avaliação Heurística e Avaliação Cooperativa
- Definir planos de manutenção do software
- Definir ferramenta de gerenciamento de configuração

5 Empírica

Quantificação dos recursos necessários:

- Definir largura de banda necessária para funcionamento do software

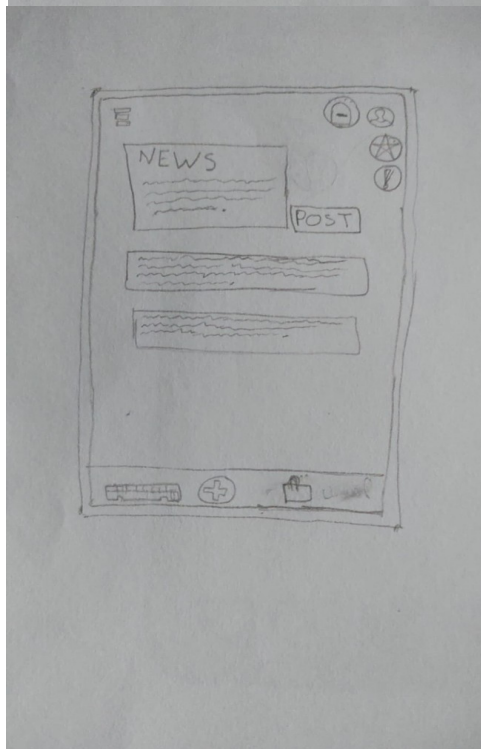
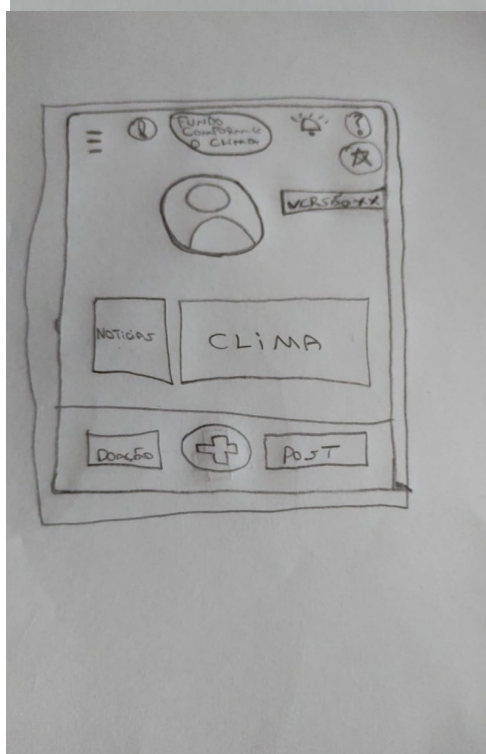
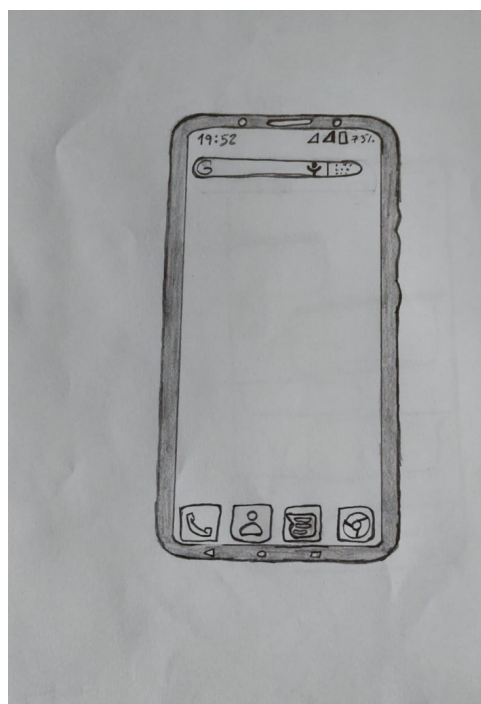
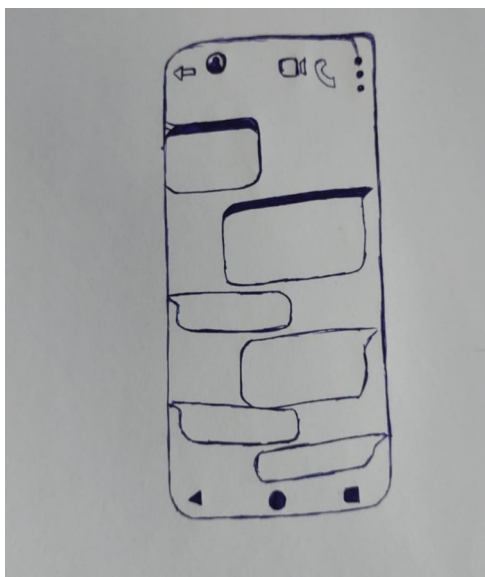
6 Mundo Físico

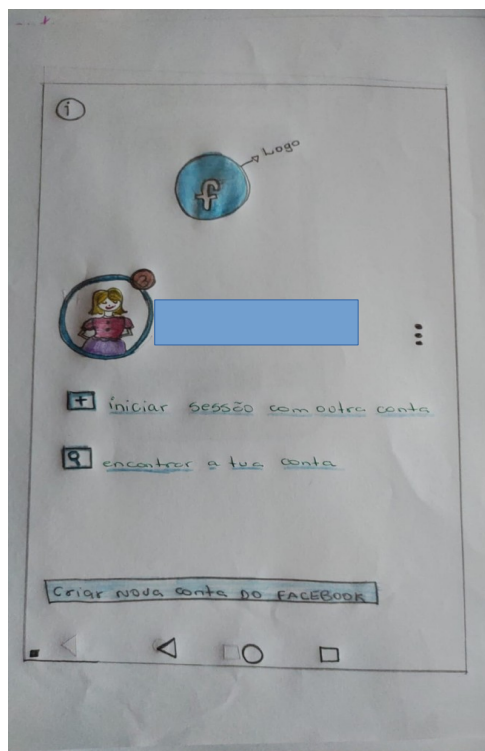
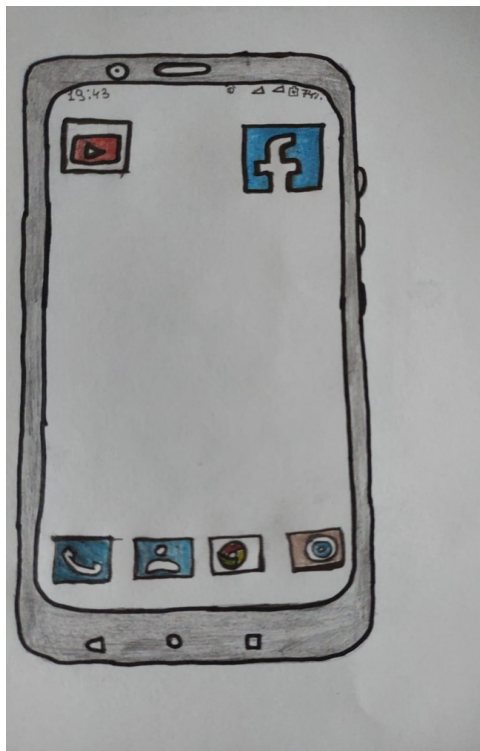
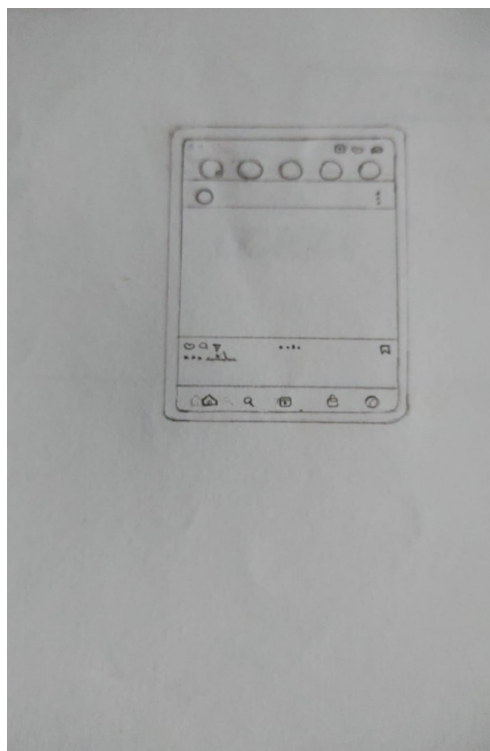
Infraestrutura necessária:

- Conexão com Internet
- Definir plataforma(s) alvo do software

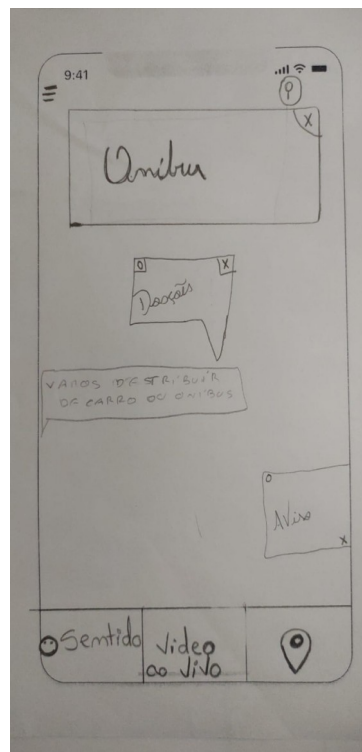
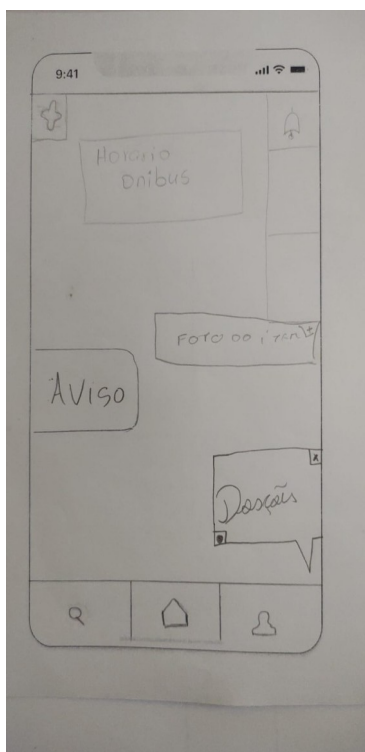
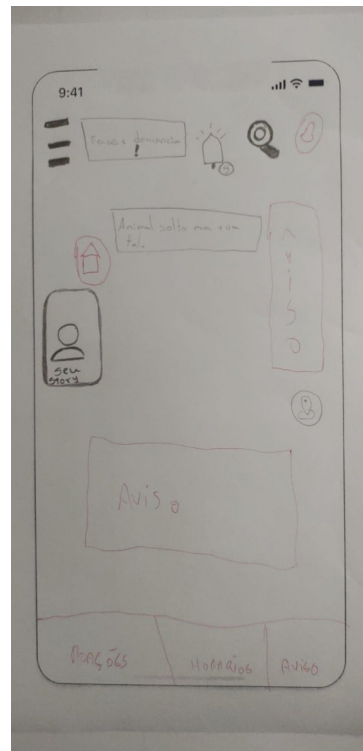
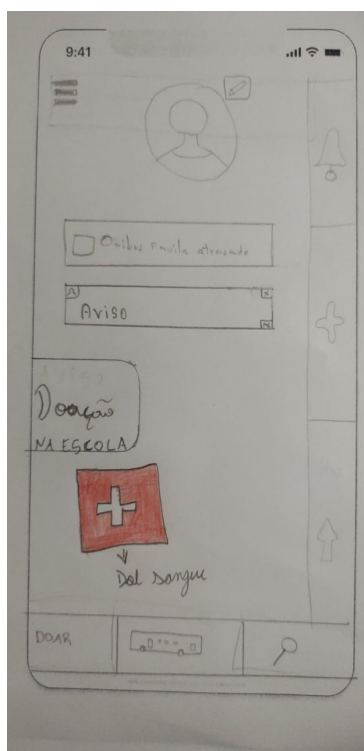
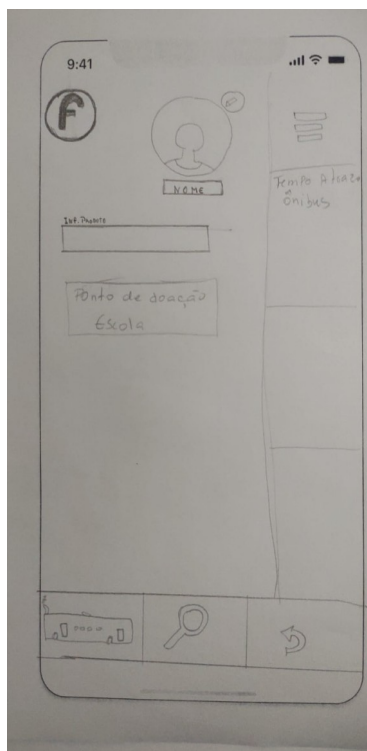
Definir infraestrutura para manutenção dos dados, seja *online* ou *offline*

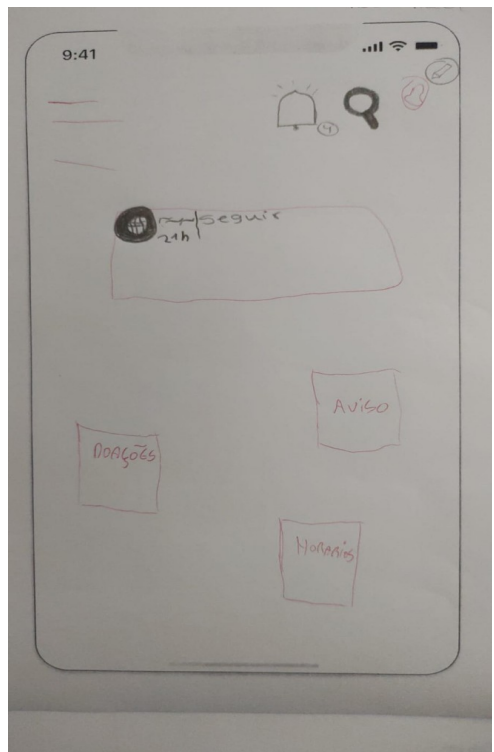
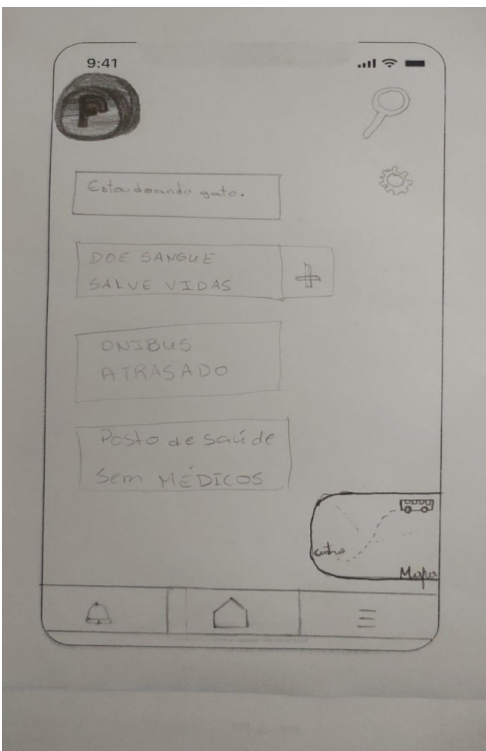
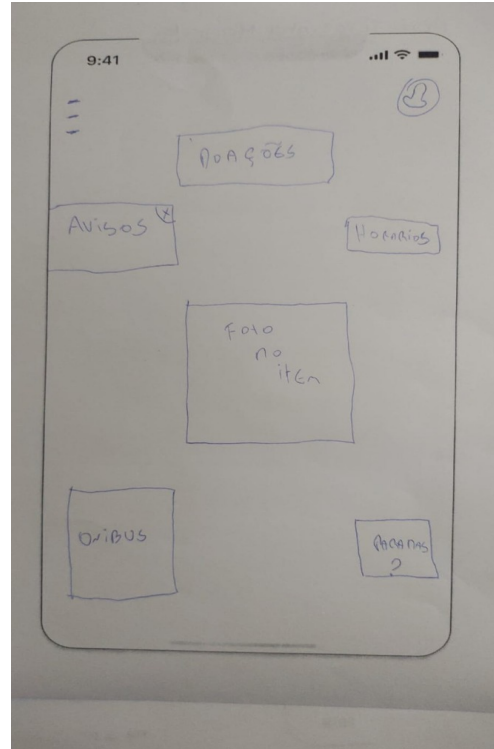
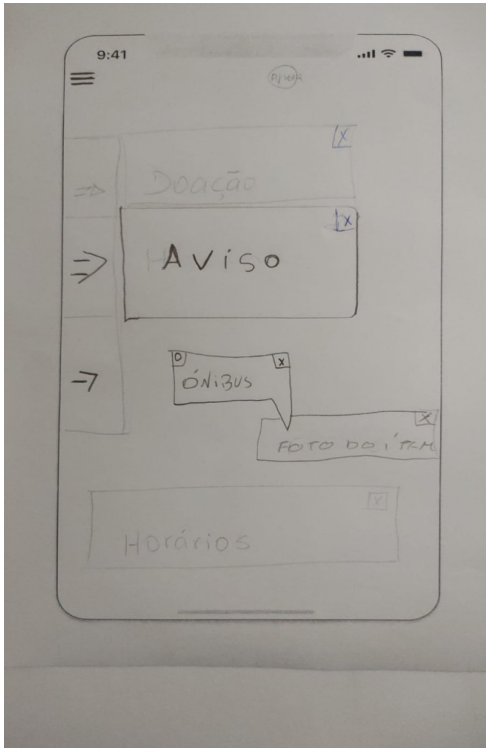
APÊNDICE H – Protótipos em papel 09/05





APÊNDICE I – Protótipos em Papel 16/05





APÊNDICE J – Documento de Requisitos

Documento de requisitos

1. Controle de Versão

Data	Versão	Descrição	Autor
10/05/2022	1.0	Versão inicial	Michel
16/05/2022	1.1	Versão validada com professora	Amanda, Michel
23/05/2022	1.2	Documento revisado	Michel

1. Glossário

Aplicação: Quando diz-se sistema lê-se software.

CRAS: Centro de Referência de Assistência Social.

TCC: Trabalho de Conclusão de Curso

1. Introdução

A aplicação, objeto deste documento de requisitos, é proposta no contexto de um projeto de Trabalho de Conclusão de curso (TCC), que tem como objetivo propor uma abordagem socialmente responsável ao desenvolvimento de software, no qual o software produzido objetive a melhoria da sociedade e que a sua abordagem de desenvolvimento considere os aspectos éticos na sua qualidade, nos processos, métodos e nas ferramentas utilizadas. Dito isso, em colaboração com alunos do ensino médio de uma escola pública de Alegrete/RS, foi levantada a necessidade de uma aplicação que registre os dados e avisos de instituições públicas, apresente avisos gerais sobre o bairro, assim como registre bens para doação, de forma que seja criada uma interface onde os cidadãos de um mesmo bairro se ajudem.

São requisitos essenciais dessa aplicação:

- Manter os dados de funcionamento de postos de saúde, escolas e Centro de Referência de Assistência Social (CRAS) do bairro;
- Autenticar usuários que desejarem publicar/editar algo;

- Manter dados de itens disponíveis para doação, como forma de ser uma interface entre doador e donatário;
- Manter horários de ônibus que tenham itinerário no bairro.
- Publicação de avisos para a comunidade

1.Requisitos éticos

RE	Descrição
RE 1	Desenvolver software que satisfaça as especificações, seja testado e não diminua a qualidade de vida ou a privacidade, ou cause danos ao meio ambiente.
RE 2	Considerar questões de deficiência física, alocação de recursos, desvantagem econômica ou outros fatores que possam diminuir o acesso aos benefícios do software. (RNF - Acessibilidade)
RE 3	Não usar, se souber, software obtido ou mantido de forma ilegal ou aética.
RE 4	Manter privada qualquer informação confidencial obtida por intermédio do trabalho profissional, desde que essa confidencialidade seja consistente com o interesse público e com a legislação.
RE 5	Assegurar que o método adequado é usado por qualquer projeto em que trabalhe ou se proponha a trabalhar.
RE 6	Assegurar que as especificações do software no qual trabalhe foram bem documentadas, satisfaçam os requisitos dos usuários e tenham as aprovações apropriadas.
RE 7	Trabalhar para desenvolver software e documentos relacionados que respeitem a privacidade daqueles que serão afetados por eles.
RE 8	Considerar em todos os julgamentos técnicos as necessidades de apoiar e manter os valores humanos.
RE 9	Assegurar boa gestão para qualquer projeto no qual trabalhe, incluindo procedimentos efetivos para a promoção da qualidade e redução dos riscos.

1.Requisitos de usuário

5.1. Requisitos funcionais

RF01 - Publicar item para doação - possibilitar ao cidadão o registro de itens para doação.

RF02 - Publicar aviso para a comunidade - possibilitar ao cidadão registrar avisos gerais para a comunidade.

RF03 - Consultar informações, horários de atendimento e avisos gerais - possibilitar ao cidadão consultar informações de instituições públicas locais (Postos de Saúde, Escolas e CRAS).

RF04 - Consultar informações sobre horários de ônibus - possibilitar ao cidadão consultar informações sobre horários de ônibus.

RF05 - Gerenciar item para doação - possibilitar ao cidadão a seleção, a edição e a exclusão do item para doação.

RF06 - Autenticar usuário - oferecer algum mecanismo de identificação do cidadão no sistema.

RF07 - Gerenciar avisos - possibilitar ao cidadão selecionar, editar ou excluir avisos enviados.

RF08 - Gerenciar informações, dados de horários de atendimento e avisos - possibilitar a um moderador criar, selecionar, editar ou excluir informações, dados sobre horários de atendimento e avisos das instituições públicas.

RF09 - Gerenciar horários de ônibus - possibilitar a um moderador criar, selecionar, editar ou excluir os horários de ônibus.

RF10 - Denunciar conteúdo impróprio - possibilitar ao cidadão denunciar algum aviso com conteúdo inadequado.

RF11 - Analisar conteúdo - moderador deverá analisar o conteúdo denunciado como impróprio, de forma que possa remover e bloquear algum cidadão com comportamento inadequado na aplicação.

7 5.2. Requisitos não funcionais

Usabilidade

RNF 01 - As interfaces devem observar os princípios de usabilidade de Nielsen;

RNF 02 - Os elementos de interface devem ser consistentes;

Acessibilidade

RNF 03 - O *software* deve funcionar em diferentes dispositivos;

RNF 04 - O *layout* deve ser responsivo;

RNF 05 - As funcionalidades devem poder ser utilizadas com apoio de leitores de telas;

RNF 06 - Deve-se priorizar a adoção de fontes sem serifa;

RNF 07 - Deve-se oferecer contraste adequado, observando o fator mínimo de 4,5:1 de taxa de contraste entre texto e plano de fundo;

RNF 08 - Deve-se adotar imagens e texto para apresentar as funcionalidades do aplicativo;

Compatibilidade

RNF 09 - A autenticação deve ser realizada com conta Google.

Segurança/proteção

RNF 10 - A privacidade dos usuários deve ser respeitada, de modo que nenhum usuário tenha acesso a informações indevidas.

RNF 11 - Mensagens de erro que são mostradas ao usuário revelarão somente a informação necessária, sem vazamento de detalhes internos do sistema na mensagem de erro.

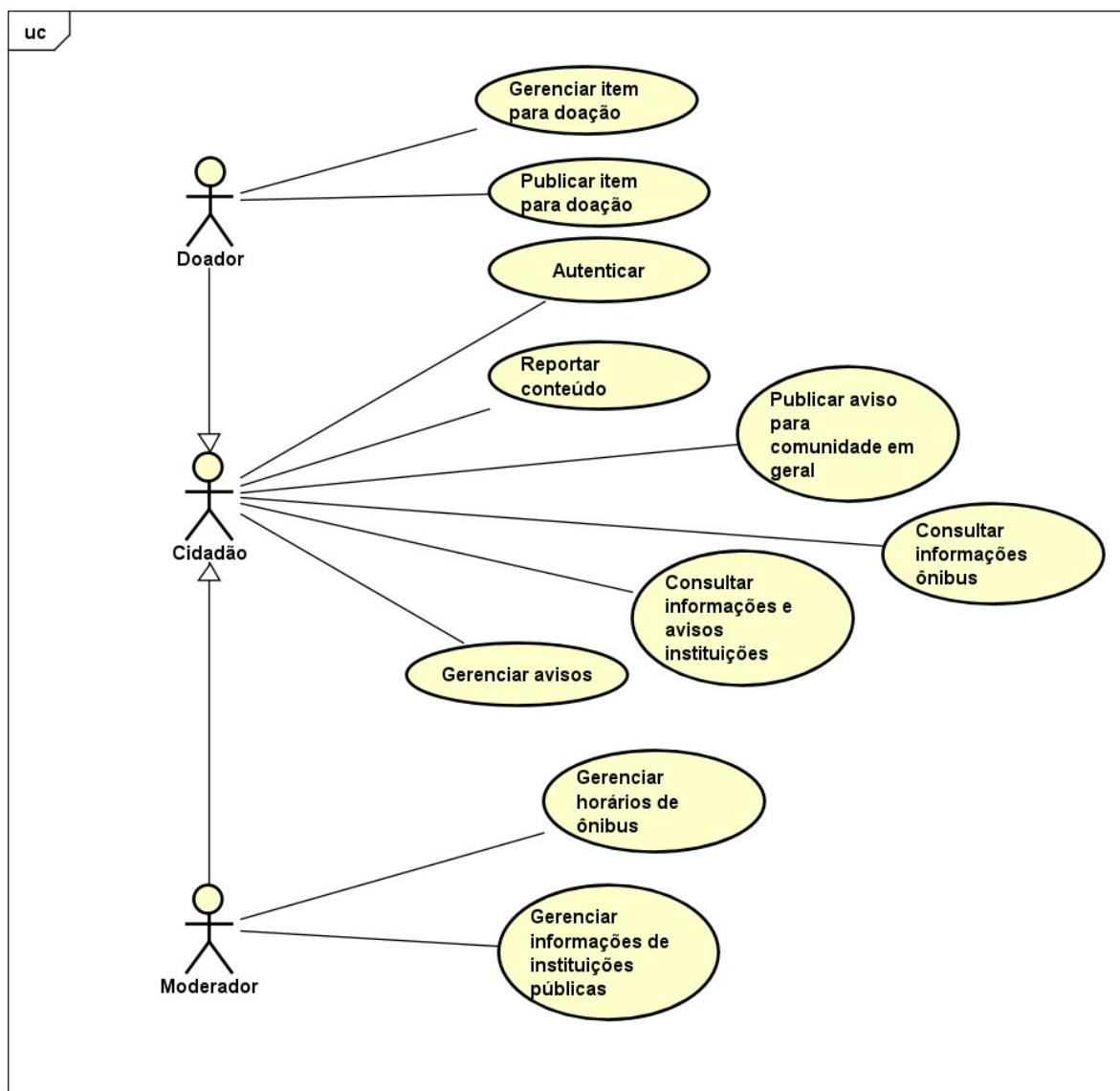
RNF 12 - A aplicação não deve solicitar ou manter dados sensíveis.

RNF 13 - Utilizar os dados recolhidos apenas para a finalidade informada ao usuário.

RNF 14 - Somente divulgar dado na aplicação com o consentimento do titular.

1. Modelos de sistema

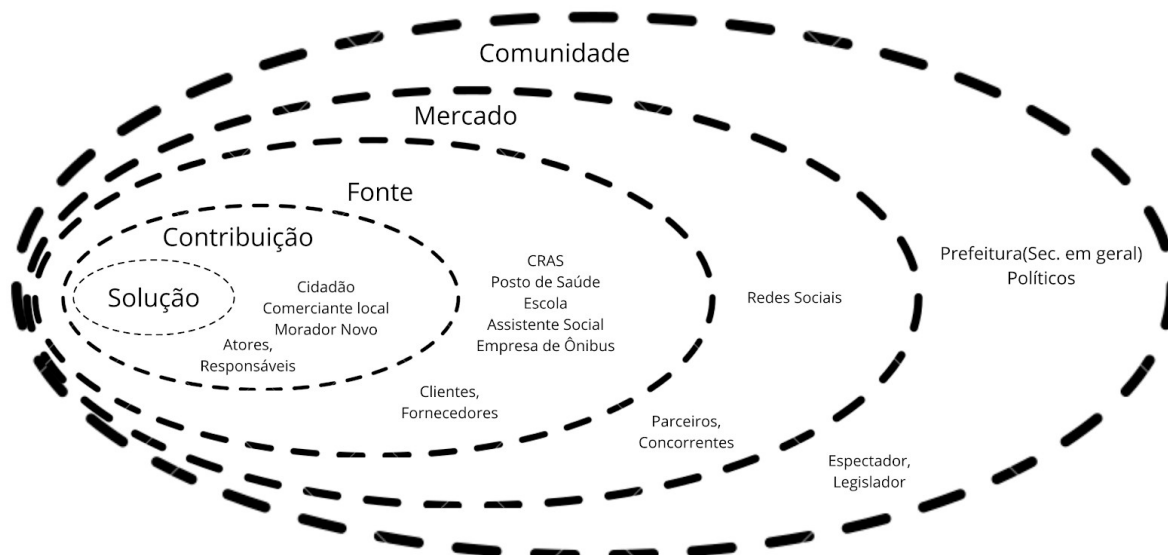
Figura 3 – Diagrama de Casos de Uso



Fonte: Autoria Própria

Apêndices e anexos

Partes Interessadas

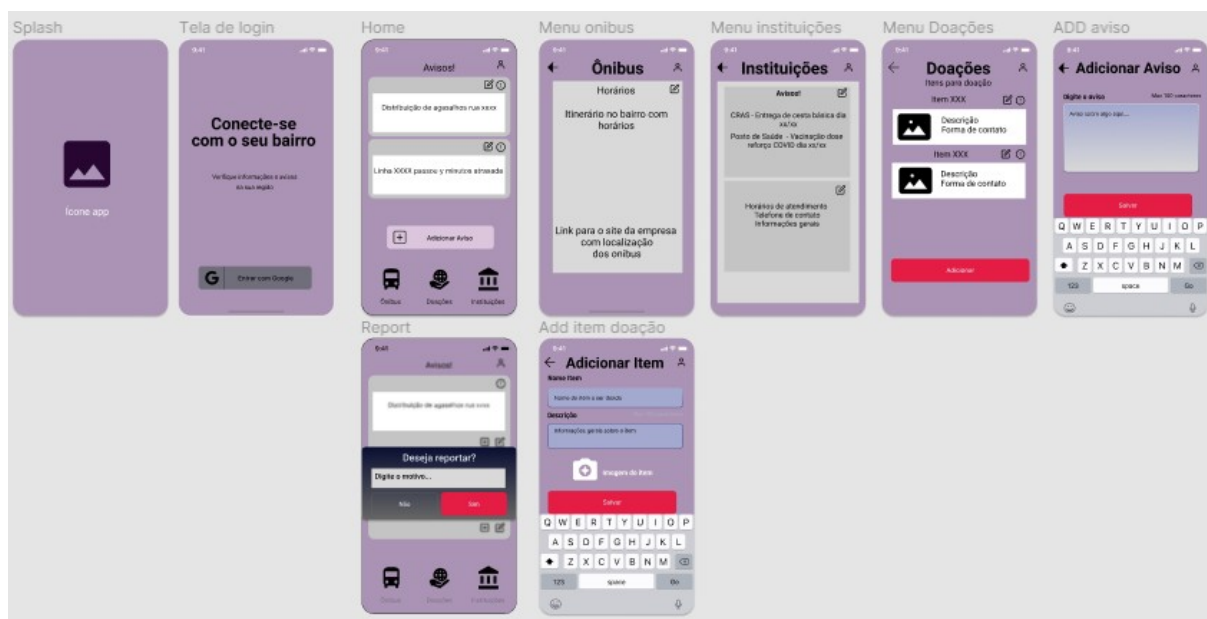


Quadro de Avaliação

Partes Interessadas	Problemas e Questões	Ideias e Soluções
Contribuição Atores Responsáveis	<ol style="list-style-type: none"> 1. Como será realizado o registo de solicitações para a prefeitura? 2. De que modo ajuda um morador novo no bairro? 	<ol style="list-style-type: none"> 1. Sistema vinculado com a prefeitura ou um cidadão protocola as todas as solicitações 2. Ajuda como forma de um guia para o cidadão que não conhece o bairro, visto que terá dados das instituições públicas e do comércio local
Fonte Clientes Fornecedores	<ol style="list-style-type: none"> 1. Quem atualiza informações sobre as 	<ol style="list-style-type: none"> 1. Próprio funcionário ou um cidadão se informa e atualiza

	instituições públicas	os dados
<p>Mercado</p> <p>Parceiros Concorrentes</p>	<p>1. Qual a vantagem do software com relação a publicar em redes sociais?</p> <p>2. Qual a vantagem de registrar solicitação na aplicação?</p>	<p>1. Vantagem da aplicação ser voltada para o público da própria região</p> <p>2. Redução de burocracia, podendo simplesmente registrar a solicitação na aplicação, sem precisar ir na prefeitura presencialmente</p>
<p>Comunidade</p> <p>Espectador Legislador</p>	<p>1. Qual a vantagem para a prefeitura?</p>	<p>1. Centralização das solicitações de uma determinada região</p>

APÊNDICE K – Mock-up primeira versão



APÊNDICE L – Roteiro avaliação heurística

INSTRUÇÕES PARA AVALIAÇÃO

1. Acessar os protótipos do aplicativo através do link: [Protótipos](#)
2. Clicar em play na parte superior
3. Realizar a seguintes atividade

AVALIAÇÃO DE USABILIDADE *ONLINE*

ESTADO INICIAL: Tela de Splash

TAREFAS:

- Tarefa 1: Acessar o aplicativo
 - Observar as informações da tela de Splash e Login
 - Autenticar através do Google
- Tarefa 2: Acessar Menus e retornar
 - Acessar as telas Ônibus, Doações e Instituições
 - Voltar para a tela inicial
- Tarefa 3: Adicionar aviso
 - Identificar na tela a opção de adicionar aviso e clicar
 - Clicar em Salvar aviso
- Tarefa 4: Reportar conteúdo impróprio
 - Identificar a opção de reportar conteúdo e clicar em “Sim”
- Tarefa 5: No menu Doações, adicionar item para doação
 - Identificar a opção para adicionar item na tela Doações e clicar
 - Na tela de adicionar item, verificar as informações e clicar em “Salvar”

Tempo esperado para a execução das tarefas: 5 min

AVALIAÇÃO

Listagem de problemas

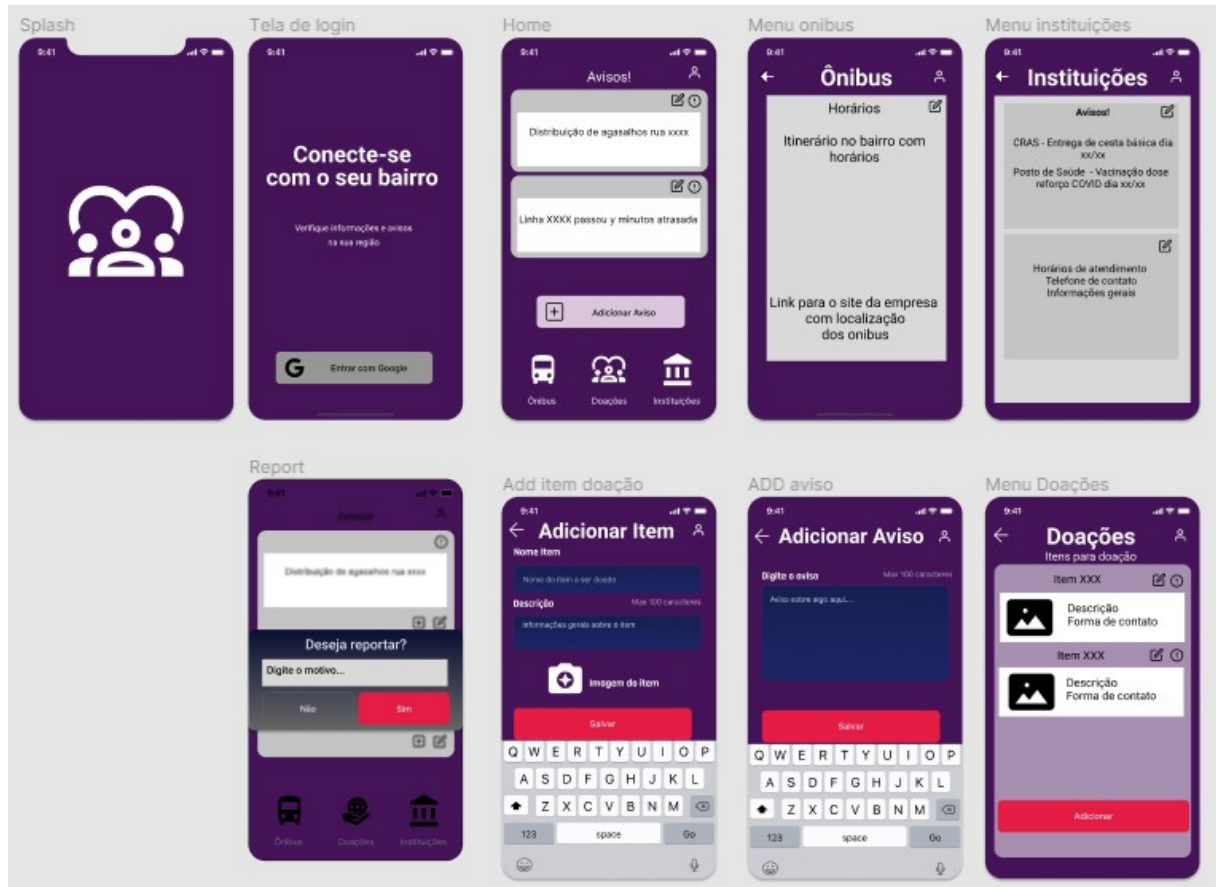
Aplicar questionário SUS

Tempo esperado para resposta do questionário: 15min

TEMPOS E PARTICIPANTES**Avaliador:** Michel S. Marques

Participante	Início	Fim	Tempo total
Avaliador 1	18:00	19:40	1:40
Avaliador 2	18:00	19:40	1:40
Avaliador 3	18:00	19:40	1:40

APÊNDICE M – Mock-up segunda versão



APÊNDICE N – Avaliação de usabilidade SUS

Avaliação de Usabilidade - SUS

Data: 06/06/2022

Questões SUS	Discordo Totalment e 1	2	3	4	Concordo Totalmente 5
Eu acho que eu gostaria de utilizar este sistema frequentemente.	•	•	•	•	•
Eu acho o sistema desnecessariamente complexo.	•	•	•	•	•
Eu acho o sistema fácil de usar.	•	•	•	•	•
Eu acho que eu precisaria da ajuda de um técnico para eu conseguir utilizar o sistema.	•	•	•	•	•
Eu acho que as várias funcionalidades do sistemas estão bem integradas.	•	•	•	•	•
Eu acho que há muita inconsistência no sistema.	•	•	•	•	•
Eu acho que a maioria das pessoas aprende facilmente a utilizar esse sistema.	•	•	•	•	•
Eu acho o sistema muito incômodo de usar.	•	•	•	•	•
Eu me senti muito confiante em utilizar o sistema.	•	•	•	•	•
Eu precisei aprender muito antes de utilizar esse sistema.	•	•	•	•	•

Liste os aspectos mais negativos do protótipo:

Liste os aspectos mais positivos do protótipo:

APÊNDICE O – Matriz priorização de requisitos

Priorização de Requisitos

Data: 10/06/2022

Os requisitos devem ser priorizados de acordo com a matriz apresentada na Figura 1, onde:

1. Urgente e importante
2. Urgente, mas não importante
3. Importante, mas não urgente
4. Nem urgente, nem importante

Figura 1. Matriz de Eisenhower



Fonte: <https://i0.wp.com/blog.softensistemas.com.br/wp-content/uploads/2019/02/matriz-eisenhower.jpg?resize=700%2C498&ssl=1>

RF01 - Publicar item para doação - possibilitar ao cidadão o registro de itens para doação.

- 1 - Urgente e importante

- ~~2 - Urgente, mas não importante~~
- ~~3 - Importante, mas não urgente~~
- ~~4 - Nem urgente, nem importante~~

RF02 - Publicar aviso para a comunidade - possibilitar ao cidadão registrar avisos gerais para a comunidade.

- ~~1 - Urgente e importante~~
- ~~2 - Urgente, mas não importante~~
- 3 - Importante, mas não urgente
- ~~4 - Nem urgente, nem importante~~

RF03 - Consultar informações, horários de atendimento e avisos gerais - possibilitar ao cidadão consultar informações de instituições públicas locais (Postos de Saúde, Escolas e CRAS).

- ~~1 - Urgente e importante~~
- ~~2 - Urgente, mas não importante~~
- 3 - Importante, mas não urgente
- ~~4 - Nem urgente, nem importante~~

RF04 - Consultar informações sobre horários de ônibus - possibilitar ao cidadão consultar informações sobre horários de ônibus.

- 1 - Urgente e importante
- ~~2 - Urgente, mas não importante~~
- ~~3 - Importante, mas não urgente~~
- ~~4 - Nem urgente, nem importante~~

RF05 - Gerenciar item para doação - possibilitar ao cidadão a seleção, a edição e a exclusão do item para doação.

- 1 - Urgente e importante
- ~~2 - Urgente, mas não importante~~
- ~~3 - Importante, mas não urgente~~
- ~~4 - Nem urgente, nem importante~~

RF06 - Autenticar usuário - oferecer algum mecanismo de identificação do cidadão no sistema.

- 1 - Urgente e importante

- ~~2 - Urgente, mas não importante~~
- ~~3 - Importante, mas não urgente~~
- ~~4 - Nem urgente, nem importante~~

RF07 - Gerenciar avisos - possibilitar ao cidadão selecionar, editar ou excluir avisos enviados.

- ~~1 - Urgente e importante~~
- ~~2 - Urgente, mas não importante~~
- 3 - Importante, mas não urgente
- ~~4 - Nem urgente, nem importante~~

RF08 - Gerenciar informações, dados de horários de atendimento e avisos - possibilitar a um moderador criar, selecionar, editar ou excluir informações, dados sobre horários de atendimento e avisos das instituições públicas.

- ~~1 - Urgente e importante~~
- ~~2 - Urgente, mas não importante~~
- 3 - Importante, mas não urgente
- ~~4 - Nem urgente, nem importante~~

RF09 - Gerenciar horários de ônibus - possibilitar a um moderador criar, selecionar, editar ou excluir os horários de ônibus.

- 1 - Urgente e importante
- ~~2 - Urgente, mas não importante~~
- ~~3 - Importante, mas não urgente~~
- ~~4 - Nem urgente, nem importante~~

RF10 - Denunciar conteúdo impróprio - possibilitar ao cidadão denunciar algum aviso com conteúdo inapropriado.

- ~~1 - Urgente e importante~~
- ~~2 - Urgente, mas não importante~~
- ~~3 - Importante, mas não urgente~~
- 4 - Nem urgente, nem importante

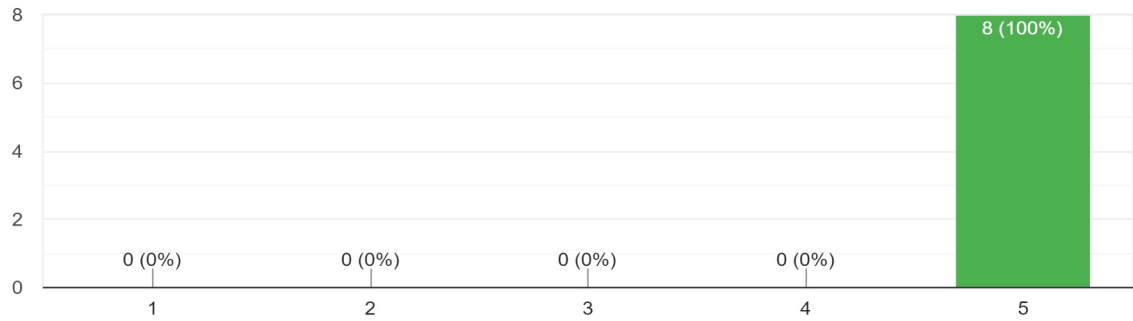
RF11 - Analisar conteúdo - moderador deverá analisar o conteúdo denunciado como impróprio, de forma que possa remover e bloquear algum cidadão com comportamento inadequado na aplicação.

- ~~1 - Urgente e importante~~
- ~~2 - Urgente, mas não importante~~
- ~~3 - Importante, mas não urgente~~
- 4 - Nem urgente, nem importante

APÊNDICE P – Resumo de respostas da avaliação SUS - 1º ciclo de desenvolvimento

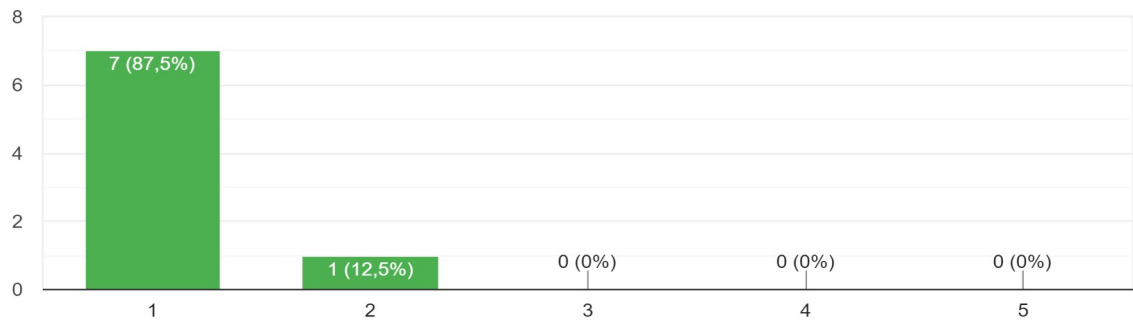
Eu acho que eu gostaria de utilizar este sistema frequentemente.

8 respostas



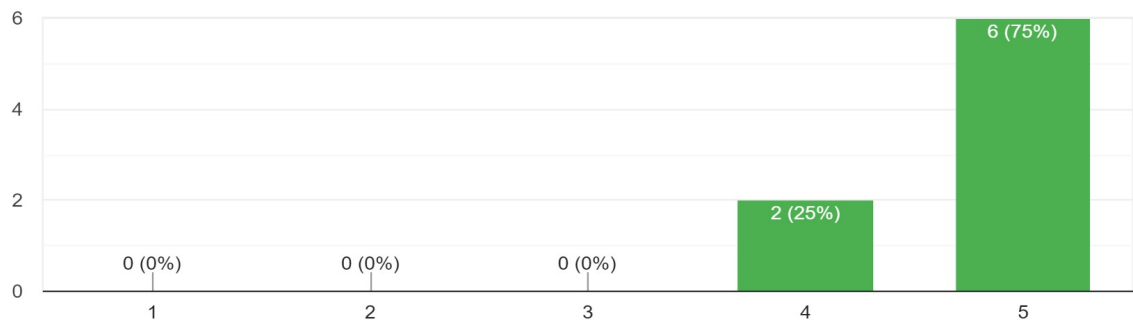
Eu acho o sistema desnecessariamente complexo.

8 respostas



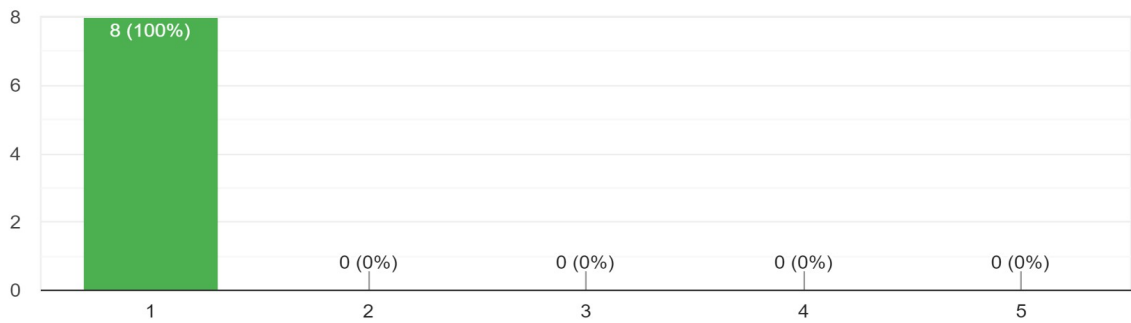
Eu acho o sistema fácil de usar.

8 respostas



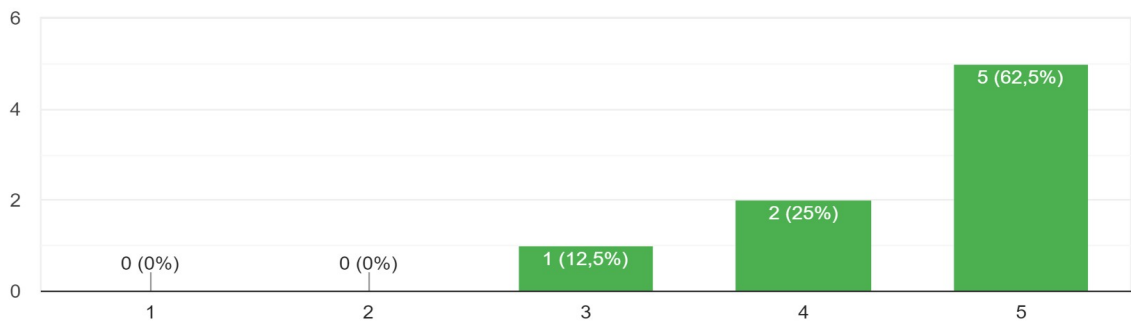
Eu acho que eu precisaria da ajuda de um técnico para eu conseguir utilizar o sistema.

8 respostas



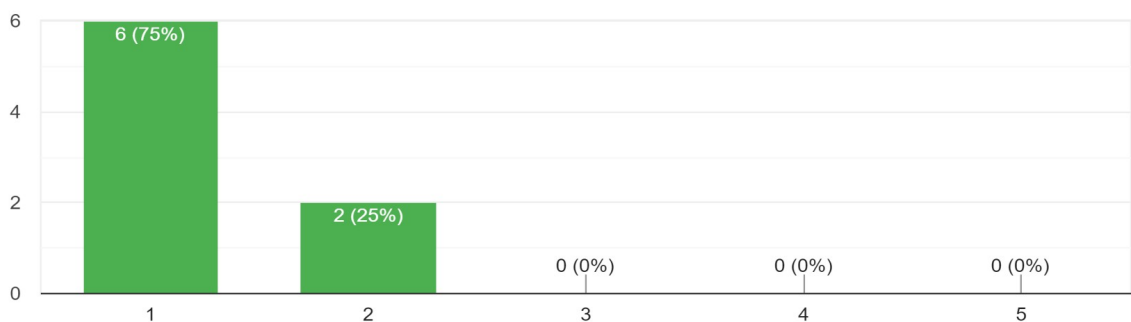
Eu acho que as várias funcionalidades do sistemas estão bem integradas.

8 respostas



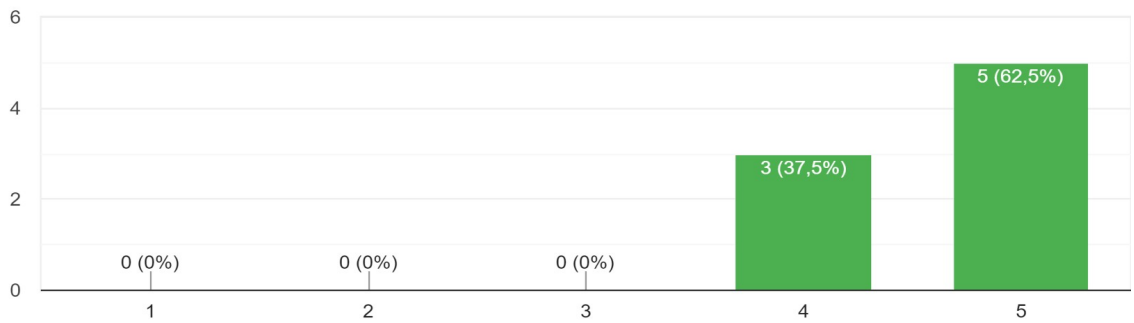
Eu acho que há muita inconsistência no sistema.

8 respostas



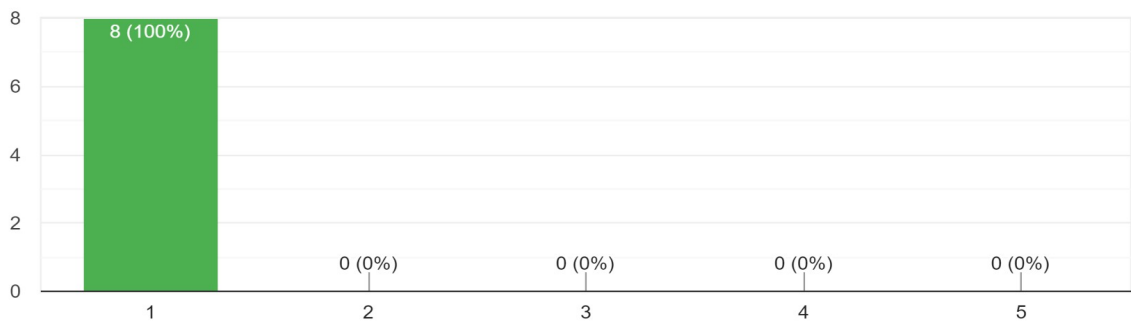
Eu acho que a maioria das pessoas aprende facilmente a utilizar esse sistema.

8 respostas



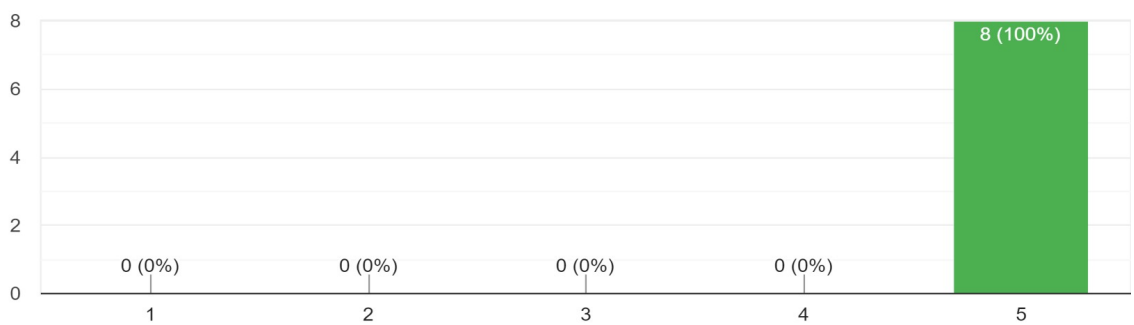
Eu acho o sistema muito incômodo de usar.

8 respostas



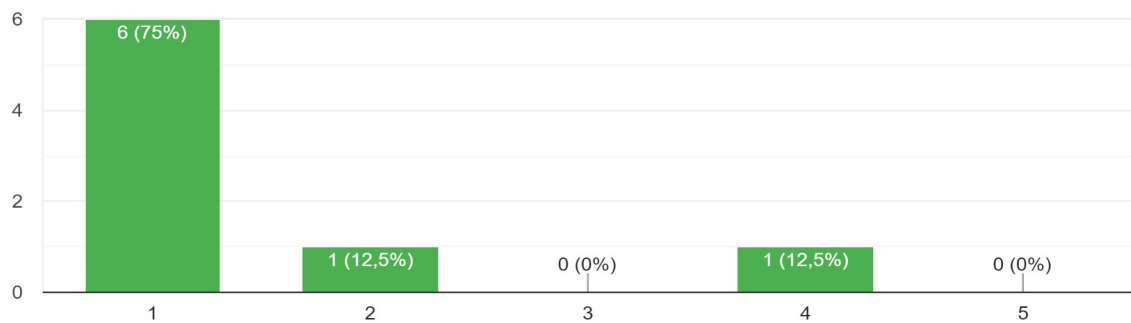
Eu me senti muito confiante em utilizar o sistema.

8 respostas



Eu precisei aprender muito antes de utilizar esse sistema.

8 respostas



Liste os aspectos mais negativos da plataforma:

- Infelizmente alguns idosos que não tem celular para usar o aplicativo
- Quadrado branco e letras pretas

Liste os aspectos mais positivos da plataforma:

- Cores
- Funcionalidade do ônibus é muito necessária pra quem depende do mesmo
- Ícones grandes
- Facilita muito no dia a dia e ajuda o próximo
- Prático

APÊNDICE Q – Documento de Validação

Validação do Produto Desenvolvido

Data: 11/07/2022

Com relação ao produto desenvolvido, responda:

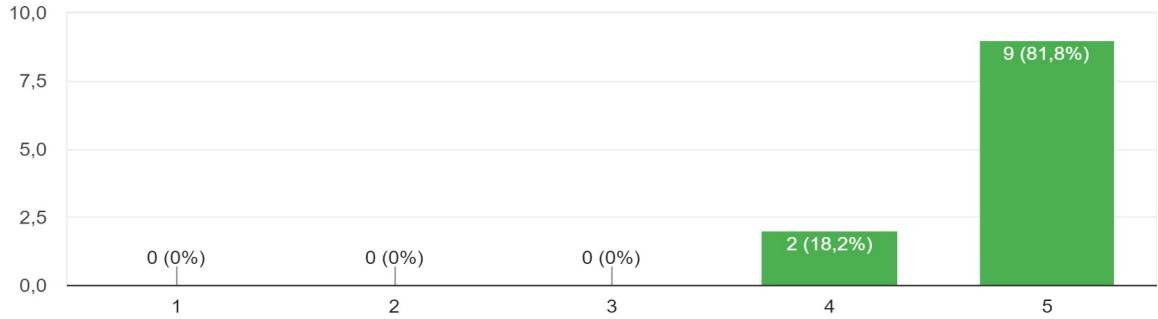
Questão	Discordo Totalmente 1	2	3	4	Concordo Totalmente 5
A autenticação está de acordo com o que foi idealizado.					
As funcionalidades relacionadas aos ônibus atendem ao que foi proposto inicialmente.					
As funcionalidades relacionadas às doações atendem ao que foi proposto inicialmente.					
As funcionalidades relacionadas às instituições atendem ao que foi proposto inicialmente.					
As funcionalidades relacionadas aos avisos atendem ao que foi proposto inicialmente.					
Os requisitos éticos são atendidos.					

Sugestões de melhorias:

APÊNDICE Q – Resultado Validação

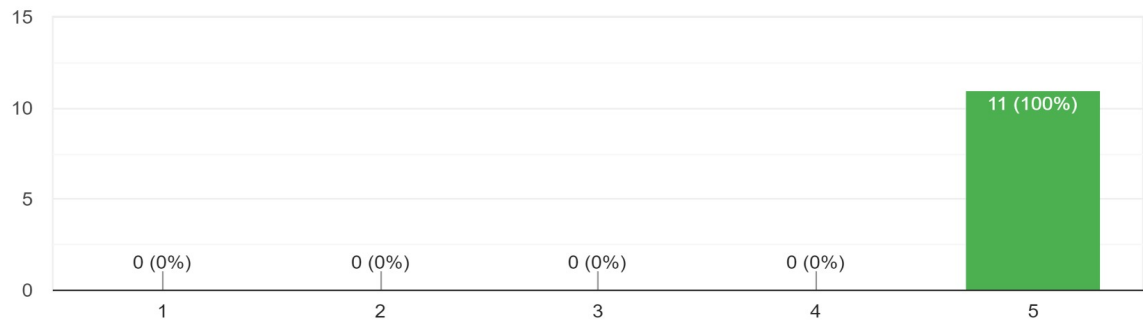
A autenticação está de acordo com o que foi idealizado.

11 respostas



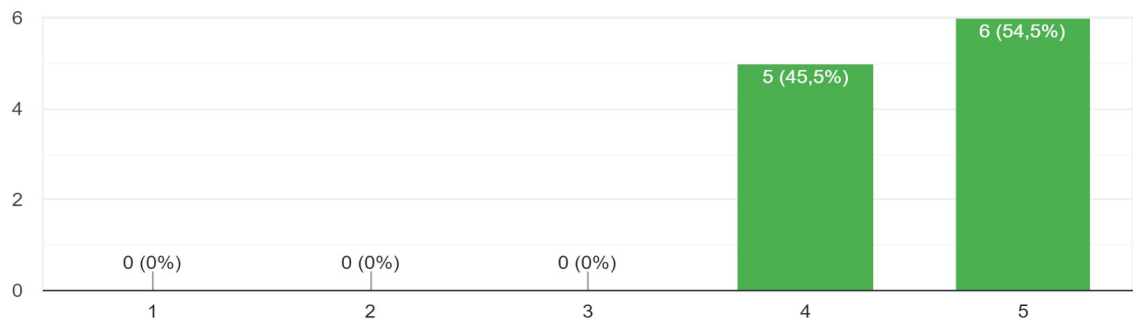
As funcionalidades relacionadas aos ônibus atendem ao que foi proposto inicialmente.

11 respostas



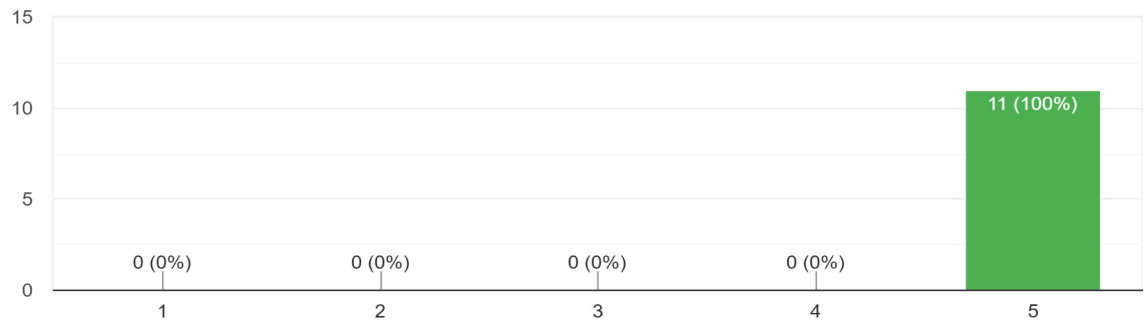
As funcionalidades relacionadas às doações atendem ao que foi proposto inicialmente.

11 respostas



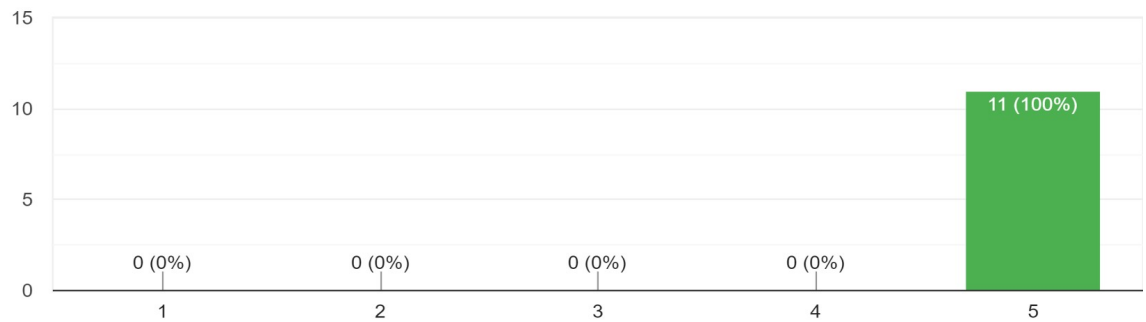
As funcionalidades relacionadas às instituições atendem ao que foi proposto inicialmente.

11 respostas



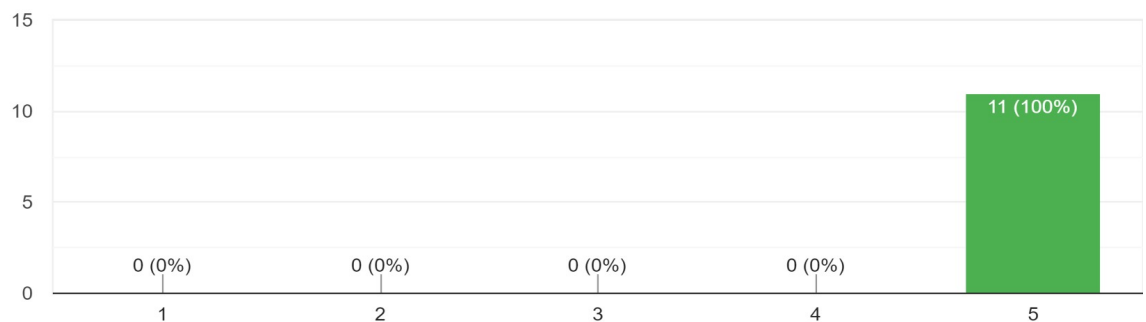
As funcionalidades relacionadas aos avisos atendem ao que foi proposto inicialmente.

11 respostas



Os requisitos éticos são atendidos.

11 respostas



APÊNDICE S – Termo de uso

Termos de Uso

Termos e condições gerais de uso do aplicativo ComuniApp

O aplicativo ComuniApp foi produzido como resultado do Trabalho de Conclusão de Curso de Michel Saucedá Marques do curso de Engenharia de Software da Universidade Federal do Pampa (Unipampa), apresentado em 03 de agosto de 2022.

O aplicativo caracteriza-se pela prestação dos seguintes serviços: Publicação de avisos para a comunidade, publicação de itens para doação, publicação de informações das instituições públicas do bairro Favila e entorno e pela visualização dos ônibus em tempo real em um mapa.

Ao utilizar o aplicativo o usuário aceita integralmente as presentes normas e compromete-se a observá-las, sob o risco de aplicação das penalidades cabíveis.

A aceitação deste termo é imprescindível para o acesso e para a utilização de quaisquer serviços fornecidos. Caso não concorde com as disposições deste instrumento, o usuário não deve utilizá-los.

É de responsabilidade do usuário:

- a) a correta utilização do aplicativo, dos serviços oferecidos, prezando pela boa convivência, pelo respeito e cordialidade entre os usuários;
- b) pelo cumprimento e respeito ao conjunto de regras disposto nesse Termo de Condições Geral de Uso, na respectiva Política de Privacidade e na legislação nacional e internacional.

O desenvolvedor poderá, a qualquer momento, advertir, suspender ou cancelar a conta do usuário:

- a) que violar qualquer dispositivo do presente Termo;
- b) que descumprir os seus deveres de usuário;
- c) que tenha qualquer comportamento fraudulento, doloso ou que ofenda a terceiros.

Caso você tenha dúvidas sobre este Termo de Uso ou sobre nossas práticas relacionadas, entre em contato conosco através do e-mail michelmarques.aluno@unipampa.edu.br.

APÊNDICE T – Política de privacidade

Política de Privacidade

A presente Política de Privacidade contém informações sobre coleta, uso, armazenamento, tratamento e proteção dos dados pessoais dos usuários do aplicativo ComuniApp, com a finalidade de demonstrar absoluta transparência quanto ao assunto e esclarecer a todos interessados sobre os tipos de dados que são coletados, os motivos da coleta e a forma como os usuários podem gerenciar ou excluir as suas informações pessoais.

Esta Política de Privacidade aplica-se a todos os usuários do aplicativo ComuniApp e integra os termos e condições gerais de uso do aplicativo, produzidos como resultados do Trabalho de Conclusão de Curso de Michel Saucedá Marques do curso de Engenharia de Software da Universidade Federal do Pampa (Unipampa), apresentado em 03 de agosto de 2022.

O presente documento foi elaborado em conformidade com a Lei Geral de Proteção de Dados Pessoais (Lei 13.709/18).

Informações que coletamos ou recebemos

Os dados pessoais do usuário são recolhidos pelo aplicativo da seguinte forma:

- Quando o usuário acessa a aplicação: é registrado o nome do usuário, e-mail e data de aceitação dos termos de uso e privacidade. A partir deles, podemos identificar o usuário e o visitante, além de garantir uma maior segurança e bem-estar às suas necessidades. Esses dados não podem ser acessados pelos demais usuários, sendo utilizados apenas para a identificação de ações do usuário dentro da aplicação.
- Quando um usuário publica algo no aplicativo: as informações sobre interação são registradas para garantir uma melhor experiência ao usuário. Estes dados são coletados em caso de publicação de item para doação, como o contato e o identificador do usuário.
- Por intermédio de terceiro: o aplicativo recebe dados do Google, quando um usuário faz login com o seu perfil desse site. A utilização desses dados é autorizada previamente pelos usuários junto ao terceiro em questão.

Os dados podem ser removidos ou anonimizados a pedido do usuário, excetuando os casos em que a lei oferecer outro tratamento.

Os dados pessoais não podem ser compartilhados com terceiros.

Ao utilizar os serviços e fornecer as informações pessoais no aplicativo, o usuário está consentindo com a presente Política de Privacidade.

Reservamos o direito de modificar essa Política de Privacidade a qualquer momento, então, é recomendável que o usuário revise-a com frequência.

Caso você tenha dúvidas sobre esta Política de Privacidade ou sobre nossas práticas relacionadas, entre em contato conosco através do e-mail michelmarques.aluno@unipampa.edu.br.