



**UNIVERSIDADE FEDERAL DO TOCANTINS  
CÂMPUS UNIVERSITÁRIO DE PALMAS  
CURSO DE CIÊNCIA DA COMPUTAÇÃO**

**DATAGUARDIAN - SISTEMA PARA AUXÍLIO DE CONSELHOS  
TUTELARES NO REGISTRO E TRATAMENTO DE INFORMAÇÕES**

**RODRIGO MAGALHÃES DIAS DA SILVA**

**PALMAS (TO)**

**2018**

RODRIGO MAGALHÃES DIAS DA SILVA

DATAGUARDIAN - SISTEMA PARA AUXÍLIO DE CONSELHOS TUTELARES NO  
REGISTRO E TRATAMENTO DE INFORMAÇÕES

Trabalho de Conclusão de Curso II apresentado  
à Universidade Federal do Tocantins para  
obtenção do título de Bacharel em Ciência da  
Computação, sob a orientação do(a) Prof.(a)  
D.Sc. Ary Henrique Morais de Oliveira.

Orientador: D.Sc. Ary Henrique Morais de  
Oliveira

PALMAS (TO)

2018

RODRIGO MAGALHÃES DIAS DA SILVA

DATAGUARDIAN - SISTEMA PARA AUXÍLIO DE CONSELHOS TUTELARES NO  
REGISTRO E TRATAMENTO DE INFORMAÇÕES

Trabalho de Conclusão de Curso II apresentado à UFT – Universidade Federal do Tocantins – Câmpus Universitário de Palmas, Curso de Ciência da Computação foi avaliado para a obtenção do título de Bacharel e aprovada em sua forma final pelo Orientador e pela Banca Examinadora.

Data de aprovação: 13 / 6 / 2018

Banca Examinadora:

---

Prof. Ary Henrique Morais de Oliveira D.Sc.

---

Prof. Thiago Magalhães de Brito Rodrigues M.Sc.

---

Prof. Warley Gramacho da Silva D.Sc.

**Dados Internacionais de Catalogação na Publicação (CIP)**  
**Sistema de Bibliotecas da Universidade Federal do Tocantins**

---

- S586d Silva, Rodrigo Magalhães Dias da.  
Dataguardian - Sistema para auxílio de Conselhos Tutelares no registro e tratamento de informações. / Rodrigo Magalhães Dias da Silva. – Palmas, TO, 2018.  
105 f.  
Monografia Graduação - Universidade Federal do Tocantins – Câmpus Universitário de Palmas - Curso de Ciências da Computação, 2018.  
Orientador: Ary Henrique Morais de Oliveira  
1. DATAGUARDIAN. 2. Conselho Tutelar. 3. Sistema Web. 4. Segurança de Dados. I. Título

**CDD 004**

---

TODOS OS DIREITOS RESERVADOS – A reprodução total ou parcial, de qualquer forma ou por qualquer meio deste documento é autorizado desde que citada a fonte. A violação dos direitos do autor (Lei nº 9.610/98) é crime estabelecido pelo artigo 184 do Código Penal.

**Elaborado pelo sistema de geração automática de ficha catalográfica da UFT com os dados fornecidos pelo(a) autor(a).**

*Dedico este trabalho em memória  
de meu querido irmão Miguel  
Magalhães*

## AGRADECIMENTOS

Obrigado Deus, por ter me proporcionado confiança, tempo, saúde e dedicação para realização deste trabalho. Graças ao Senhor, pude alcançar este tão desejado objetivo em minha vida. Agradeço à minha família, a qual levo comigo no coração desde e para todo o sempre. Obrigado pai, obrigado mãe, este trabalho foi desenvolvido pensando também em vocês. Quero agradecer também, a paciência, o apoio e a força que minha namorada, Fernanda Silva Nunes entregou-me nos momentos de fraqueza e de indecisão; obrigado meu bem por estar ao meu lado e ter se dedicado em me ajudar. Obrigado meus amigos que pouco se importam com este objetivo concluído, mas que sempre poderei contar e recorrer quando precisar. Faço um agradecimento especial aos meus professores, os quais me ensinaram, e sem dúvidas, me trouxeram a este momento. Tenho muito orgulho do corpo docente do curso de Ciência da Computação desta universidade. Por fim, porém com um grande grau de importância, agradeço ao meu orientador, o professor Dr. Ary Henrique Moraes de Oliveira, que me orientou, me aconselhou e exigiu de minha parte a dedicação e o estudo necessário para concluir esta etapa, saiba que o senhor foi fundamental para que eu concluísse este trabalho. Obrigado a todos os envolvidos e a todos que estiveram ao meu lado direta ou indiretamente, mas sempre torcendo pelo o meu sucesso.

## RESUMO

Este trabalho procura descrever as diretrizes, os conceitos e características do sistema *web* “DATAGUARDIAN”, que tem como propósito auxiliar os Conselheiros Tutelares em suas tarefas rotineiras, como o recolhimento e tratamento de denúncias realizadas contra os direitos das crianças e adolescentes. Sua principal motivação é informatizar as unidades conselheiras, proporcionando mais eficiência e eficácia na obtenção e gerenciamento das informações obtidas, visto que no ambiente atual, tais informações encontram-se apenas em formato impresso, ou seja, vulneráveis e sem garantia de sua integridade. Tarefas como consulta de processos e documentação, se tornam muito cansativas uma vez que, não há uma ferramenta que torne este procedimento mais eficiente. O projeto “DATA-GUARDIAN” é a proposta com objetivo de impedir a duplicidade e discrepância dos dados, garantir a segurança das informações e aumentar a produtividade nos Conselhos Tutelares.

**Palavra-chave:** DATAGUARDIAN. Conselho Tutelar. Sistema web. Segurança de Dados.

## ABSTRACT

This work looks to describes how the guidelines, concepts and characteristics of the "DATAGUARDIAN" web system, whose objective is to assist the Tutelary Counselors in their routine tasks, such as the collection and treatment of grievance made against the rights of children and adolescents. Its main motivation is computerize the advisory units, providing more efficiency and effectiveness in managing of the information obtained, since in the current environment, such information is only found in printed format, that is, vulnerable and without guarantee of its integrity. Tasks as query processes and documents, become very tiring, since, there is not a tool that makes this procedure more efficient. The "DATAGUARDIAN" project is the proposal with the objective to prevent data duplication and discrepancy, ensure information security and increase productivity in Tutelary Councils.

**Keywords:** DATAGUARDIAN. Guardianship Counselors. Web System. Data Security.

## LISTA DE FIGURAS

Figura 1.1 – Localização das Unidades Conselheiras no município de Palmas, Tocantins. . . . .	19
Figura 1.2 – Armários contendo as pastas com os casos registrados na Unidade Conselheira da Região Centro e Sul II. . . . .	20
Figura 2.1 – Ciclo da Metodologia <i>Scrum</i> . Fonte: (SISTEMAS, 2014). . . . .	25
Figura 2.2 – Exemplo de Backlog do Produto. Fonte: Adaptado de (PRIKLADNICKI; WILLI; MILANI, 2013). . . . .	26
Figura 2.3 – Definição dos papéis no Scrum. . . . .	26
Figura 2.4 – Backlog da Sprint. Fonte: Adaptado de (PRIKLADNICKI; WILLI; MILANI, 2013). . . . .	27
Figura 2.5 – Planejamento para testes de usabilidade. . . . .	30
Figura 2.6 – Como funciona uma avaliação heurística de usabilidade. . . . .	31
Figura 2.7 – Exemplo de problemas encontrados a partir de avaliação heurística. Fonte: (PRATES; BARBOSA, 2003). . . . .	33
Figura 2.8 – Formato de Resposta do SUS. Fonte: (TEIXEIRA, 2015). . . . .	34
Figura 2.9 – Demonstração do SCORE do SUS. Fonte: (SAURO, 2011b). . . . .	34
Figura 3.1 – Telas do sistema SIPIA. Fonte: (SIPIA, ). . . . .	37
Figura 3.2 – Tela do Módulo de Conselhos Tutelares do SUASnet. Fonte: (SUASNET, ). . . . .	38
Figura 3.3 – Telas do aplicativo Proteja Brasil. Fonte: (MOBILE, a). . . . .	39
Figura 4.1 – Fluxo de desenvolvimento do “DATAGUARDIAN” . . . . .	42
Figura 4.2 – Ambiente de Desenvolvimento do sistema “DATAGUARDIAN” . . . . .	43
Figura 4.3 – Fluxo de Trabalho Kanban. . . . .	45
Figura 4.4 – Ficha de Atendimento Utilizada no Conselho Tutelar de Palmas . . . . .	47
Figura 4.5 – Tela de Cadastro da Denúncia no “DATAGUARDIAN” . . . . .	47
Figura 4.6 – Diagrama de Caso de Uso Geral do sistema DATAGUARDIAN. . . . .	50

Figura 4.7 – Modelagem das classes de controle de acesso e seus relacionamentos	52
Figura 4.8 – Padrão de Arquitetura de Software MVC	52
Figura 4.9 – Diagrama de Classe do “DATAGUARDIAN”	53
Figura 4.10 – Diagrama de Classe do “DATAGUARDIAN”	54
Figura 4.11 – Diagrama de Atividade Cadastro Casos Antigos	55
Figura 5.1 – Tela de login do sistema “DATAGUARDIAN”	58
Figura 5.2 – Tela de Cadastro de Usuários “DATAGUARDIAN”	59
Figura 5.3 – Tela de Cadastro de Crianças e Adolescentes	59
Figura 5.4 – Tela de Registro de Atendimento e Aplicação de Medidas Protetivas	60
Figura 5.5 – Tela de Histórico de Atividades Extras “DATAGUARDIAN”	61
Figura 5.6 – Gráfico gerado a partir do teste de performance usando a ferramenta jMeter	62
Figura 5.7 – Teste de caixa preta utilizando o Selenium	63
Figura 5.8 – DATAGUARDIAN sendo monitorado pelo Netdata	64
Figura 5.9 – Gráfico gerado a partir das respostas do checklist de usabilidade.	65
Figura 5.10 – Gráfico gerado a partir das respostas da análise heurística.	66
Figura 5.11 – Gráfico gerado a partir da pontuação obtida através do questionário SUS.	68
Figura 1 – RF001 - Cadastro de Usuários	77
Figura 2 – RF002 - Aplicar Perfis aos Usuários	77
Figura 3 – RF003 - Cadastro de Responsáveis	78
Figura 4 – RF004 - Cadastro de Crianças e Adolescentes	78
Figura 5 – RF005 - Registrar Denúncia	78
Figura 6 – RF005 - Acompanhamento de Denúncias	79
Figura 7 – RF007 - Registro de Casos e Reincidências	79
Figura 8 – RF008 - Aplicar Medidas Protetivas	79
Figura 9 – RF009 - Registro de Solicitações	80
Figura 10 – RF010 - Registro de Atividades Extras	80
Figura 13 – RNF003 - Manutenção de Usuários	80
Figura 11 – RNF001 - Extensão de Relatórios	81
Figura 12 – RNF002 - Disponibilidade do Sistema	81

Figura 14 – RNF004 - Utilização de captcha para consulta de denúncias . . . . .	81
Figura 15 – RNF005 - Responsividade . . . . .	81
Figura 16 – RNF006 - Padrões de Projeto . . . . .	81
Figura 17 – Caso de Uso - Cadastro Responsável . . . . .	82
Figura 18 – Especificação Caso de Uso - Cadastro de Responsáveis . . . . .	82
Figura 19 – Caso de Uso - Cadastro Criança e Adolescente . . . . .	83
Figura 20 – Especificação Caso de Uso - Cadastro Criança e Adolescente . . . . .	83
Figura 21 – Caso de Uso - Registro de Denúncia . . . . .	84
Figura 22 – Especificação Caso de Uso - Registro de Denúncia . . . . .	84
Figura 23 – Caso de Uso - Caso/Reincidência . . . . .	84
Figura 24 – Especificação Caso de Uso - Caso/Reincidência . . . . .	85
Figura 25 – Caso de Uso - Solicitações Diversas . . . . .	85
Figura 26 – Especificação Caso de Uso - Registro de Solicitações . . . . .	85
Figura 27 – Caso de Uso - Atividade Extra . . . . .	86
Figura 28 – Especificação Caso de Uso - Atividades Extras . . . . .	86
Figura 29 – Diagrama de Atividade - Cadastro de Responsável . . . . .	87
Figura 30 – Diagrama de Atividade - Cadastro de Criança e Adolescente . . . . .	87
Figura 31 – Diagrama de Atividade - Registro de Denúncias . . . . .	87
Figura 32 – Diagrama de Atividade - Cadastro de Caso/Reincidência e Aplicação de Medidas Protetivas . . . . .	88
Figura 33 – Fotos capturadas em treinamentos realizados no Conselho Tutelar de Palmas. Fonte: (Produção do Próprio Autor) . . . . .	89
Figura 34 – Selecionar Funcionalidade . . . . .	90
Figura 35 – O sistema oferece ao usuário duas opções para prosseguir no cadastro . . . . .	90
Figura 36 – O usuário pode pesquisar denúncias já cadastradas a partir dos filtros . . . . .	90
Figura 37 – O sistema pergunta se a denúncia é referente a uma criança reincidente ou não . . . . .	91
Figura 38 – Se a criança/adolescente for reincidente, o sistema oferece opções para pesquisar tal pessoa . . . . .	91
Figura 39 – Tela onde pode-se pesquisar crianças e adolescentes pelo nome . . . . .	91
Figura 40 – Tela onde é possível cadastrar uma nova criança ou adolescente . . . . .	92

Figura 41 – Formulário a ser preenchido com os dados da denúncia . . . . .	92
Figura 42 – Formulário a ser preenchido com os dados da Criança / Adolescente	93
Figura 43 – Formulário a ser preenchido com os dados de endereço da Criança / Adolescente. É possível cadastrar um histórico de endereços . . .	93
Figura 44 – Formulário a ser preenchido com os dados do Atendimento. . . . .	94
Figura 45 – Caixa de diálogo onde o usuário seleciona os direitos que foram ameaçados ou violados. . . . .	94
Figura 46 – Formulário para entrada das narrativas dos envolvidos no caso. . .	95
Figura 47 – Caixa de diálogo para seleção das medidas protetivas a serem apli- cadas. . . . .	95
Figura 48 – Formulário a ser preenchido com os dados da pasta . . . . .	96
Figura 49 – Adicionar crianças e adolescentes à pasta . . . . .	96
Figura 50 – Formulário para o cadastro das crianças . . . . .	97
Figura 51 – Tabela de Anexos inseridos na pasta. . . . .	97
Figura 52 – Questionário SUS . . . . .	103

## SUMÁRIO

1	INTRODUÇÃO . . . . .	16
1.1	Descrição do Problema . . . . .	17
1.2	Justificativa . . . . .	20
1.3	Objetivos . . . . .	21
1.3.1	Objetivos Específicos . . . . .	21
1.4	Estrutura do Trabalho . . . . .	22
2	FUNDAMENTAÇÃO TEÓRICA . . . . .	23
2.1	Metodologia de Desenvolvimento de Software . . . . .	23
2.2	Metodologia de Desenvolvimento Ágil . . . . .	24
2.3	Usabilidade . . . . .	28
2.4	Testes de <i>Software</i> . . . . .	35
3	TRABALHOS RELACIONADOS . . . . .	37
3.1	SIPIA . . . . .	37
3.2	SUASnet . . . . .	38
3.3	Proteja Brasil . . . . .	39
3.4	Tradução e adaptação de <i>software</i> para o auxílio na identificação de maus-tratos em crianças e adolescentes . . . . .	40
4	METODOLOGIA . . . . .	41
4.1	Desenvolvimento do Sistema . . . . .	41
4.2	Ambiente Computacional . . . . .	42
4.3	Técnicas . . . . .	44
4.4	Diagramas UML . . . . .	49
4.5	Ferramentas Utilizadas . . . . .	56

5	RESULTADOS E DISCUSSÃO . . . . .	58
5.1	O <i>Software</i> . . . . .	58
5.2	Teste de Performance . . . . .	61
5.3	Teste de Segurança . . . . .	62
5.4	Monitoramento do Sistema . . . . .	64
5.5	Resultados da Avaliação Baseada no Especialista . . . . .	65
5.6	Aplicação do Questionário SUS . . . . .	67
6	CONCLUSÃO E TRABALHOS FUTUROS . . . . .	70
6.1	Sugestões para Trabalhos Futuros . . . . .	71
	REFERÊNCIAS . . . . .	72
	APÊNDICE A - REQUISITOS FUNCIONAIS E NÃO FUNCIONAIS .	77
	APÊNDICE B - DIAGRAMAS DE CASO DE USO . . . . .	82
	APÊNDICE C - DIAGRAMAS DE ATIVIDADE . . . . .	87
	APÊNDICE D - FOTOS DOS TREINAMENTOS REALIZADOS . . . . .	89
	APÊNDICE E - TAREFA 1: CADASTRO DENÚNCIA . . . . .	90
	APÊNDICE F - TAREFA 2: CADASTRO DE CRIANÇAS E ADOLES- CENTES . . . . .	93
	APÊNDICE G - TAREFA 3: REGISTRO DE ATENDIMENTO . . . . .	94
	APÊNDICE H - TAREFA 4: CADASTRO DE PASTAS ANTIGAS . .	96
	APÊNDICE I - PERFIL DOS AVALIADORES . . . . .	98
	APÊNDICE J - PERFIL DOS USUÁRIOS PARTICIPANTES DOS TES- TES DE USABILIDADE . . . . .	99
	APÊNDICE K - COMPILAÇÃO ANÁLISE HEURÍSTICA . . . . .	100
	ANEXO A - QUESTIONÁRIO DE TESTE DE USABILIDADE - SUS	103
	ANEXO B - TEMPLATE DO QUESTIONÁRIO PARA AVALIAÇÃO HEURÍSTICA DE USABILIDADE . . . . .	104
	ANEXO C - CHECKLIST PARA AVALIAÇÃO HEURÍSTICA . . . . .	105

## 1 INTRODUÇÃO

O Brasil é uma república federativa composta por 5.570 municípios (BRASIL, 2013), onde foram identificados 5.906 Conselhos Tutelares, de acordo com (REPÚBLICA, 2012). A figura do Conselho Tutelar teve início no dia 13 de julho de 1990 juntamente ao Estatuto da Criança e do Adolescente (ECA), instituído pela Lei 8.069 (REPÚBLICA, 1990). Segundo o ECA (DIGIÁCOMO; DIGIÁCOMO, 2010) os Conselhos Tutelares são órgãos municipais destinados a zelar pelo cumprimento dos direitos da criança e do adolescente (artigos 131 a 140). Além disso, são entidades vitalícias, ou seja, uma vez criado não podem ser extintos. São autônomas em suas decisões, mas não possuem poder jurisdicional, isso é, não julga nem aplica medidas judiciais.

O Conselho Tutelar é composto por cinco membros eleitos pela comunidade para um mandato de quatro anos, cuja responsabilidade é não somente acompanhar as crianças e os adolescentes, mas também, garantir em conjunto, seus direitos quando estes forem ameaçados ou violados pela sociedade, pelo Estado, pais, responsável ou em razão de sua própria conduta. Devido ao seu caráter de fiscalização, todos os entes de proteção, família, comunidade e o próprio Estado, o Conselho goza de autonomia funcional, não sendo subordinado a outros órgãos do Estado. Apesar desta autonomia, o mesmo possui os limitantes funcional, prestação de serviços à comunidade de acordo com o conjunto de atribuições definidos no ECA, e territorial, ou seja, local onde pode atuar. Nos casos onde atuam mais de um Conselho Tutelar, os conflitos de competência entre os Conselhos são resolvidos pelo Conselho Municipal dos Direitos da Criança e do Adolescente (CMDCA), a luz das disposições da Lei municipal (BRASILEIRINHO, 2016).

De acordo com o ECA, são atribuições do Conselho Tutelar:

1. atender as crianças e adolescentes nas hipóteses previstas nos arts. 98 e 105, aplicando as medidas previstas no art. 101, I a VII;
2. atender e aconselhar pais ou responsável, aplicando as medidas previstas no art.129, I a VII;
3. promover a execução de suas decisões, podendo para tanto: a) requisitar serviços públicos nas áreas de saúde, educação, serviço social, previdência, trabalho e segurança; b) representar junto à autoridade judiciária nos casos de descumprimento injustificado de suas deliberações;
4. encaminhar ao Ministério Público notícia de fato que constitua infração administrativa ou penal contra os direitos da criança e do adolescente;
5. encaminhar à autoridade judiciária os casos de sua competência;

6. providenciar a medida estabelecida pela autoridade judiciária, dentre as previstas no art. 101, de I a VI, para o adolescente autor do ato infracional;
7. expedir notificações;
8. requisitar certidões de nascimento e de óbito de criança ou adolescente quando necessário;
9. assessorar o Poder Executivo local na elaboração da proposta orçamentária para planos e programas de atendimento dos direitos da criança e do adolescente;
10. representar, em nome da pessoa e da família, contra a violação dos direitos previstos no art. 220, §3º, inciso II, da Constituição Federal;
11. representar ao Ministério Público, para efeito das ações de perda ou suspensão do pátrio poder.

A partir da criação do Estatuto da Criança e do Adolescente (ECA), em 1990, tornou-se real garantir a proteção integral dos menores de 18 anos, sendo que só após o ECA é que “[...] a infância e adolescência adquiriram status de sujeitos de direitos” (PEREZ; PASSONE, 2010). Assim, diversas diretrizes foram definidas para assegurar a proteção e o desenvolvimento saudável desta população, como pode ser observado no artigo 3º do ECA, onde diz que “[...] a criança e o adolescente gozam de todos os direitos fundamentais inerentes à pessoa humana [...]” (REPÚBLICA, 1990). Posteriormente, no artigo 4º, convoca toda a sociedade enquanto responsável pela efetivação desses direitos nas esferas referentes a aspectos como saúde, alimentação, educação, lazer, cultura, dignidade e respeito (CALZA, 2014). E o artigo 5º, que trata dos variados tipos de violência e opressão contra crianças e adolescentes como exploração, negligência, discriminação, violência e crueldade.

O ECA também declara que, uma vez que tais direitos sejam violados ou ameaçados, medidas protetivas devem ser aplicadas (Artigo 98). E nesse sentido, entra as atividades do Conselho Tutelar, que deve ser comunicado obrigatoriamente sempre que determinar caso de suspeita ou confirmação de maus-tratos, deste modo, tanto a população quanto o próprio Conselho, trabalham em conjunto para garantir a proteção de crianças e adolescentes através de denúncias, investigações e aplicação de medidas protetivas.

## 1.1 Descrição do Problema

A notificação dos casos de maus-tratos é essencial para o enfrentamento da violência, uma vez que possibilita a interrupção desta e desencadeia medidas de proteção, além de gerar informações para avaliação da situação local e da necessidade de investimentos

públicos (DESLANDES et al., 2011). A notificação é o ponto inicial para que uma determinada situação de violência seja registrada no sistema de justiça (DOBKE; SANTOS; DELL'AGLIO, 2010). Todavia, mesmo que o Artigo 13º do ECA preconize que quaisquer casos de suspeita ou confirmação de maus-tratos deva ser obrigatoriamente comunicados ao Conselho Tutelar, essa tarefa ainda é um desafio a ser alcançado (DOBKE; SANTOS; DELL'AGLIO, 2010).

No trabalho realizado por Gonçalves e Ferreira (GONÇALVES; FERREIRA, 2002) afirma que, embora seja evidente a legislação quanto à obrigatoriedade da notificação em casos de violência intrafamiliar, não deixa de existir pouca orientação e esclarecimento aos profissionais de saúde envolvidos. Já (DESLANDES et al., 2011) mencionam a precária estrutura física e a carência de profissionais capacitados nos Conselhos Tutelares, cuja função é primordial para o manejo desses casos.

Muitos trabalhos que tem como objetivo efetuar tomadas de decisões acerca de maus-tratos contra crianças foram desenvolvidos em diversos países, inclusive o Brasil. Na Holanda por exemplo, foi implementada uma metodologia denominada ORBA, que visa tornar o processo de decisão mais sistemático e eficiente (KWAADSTENIET et al., 2013). Nos Estados Unidos, um *software* denominado *Computer Assisted Maltreatment Inventory* (CAMI) foi desenvolvido por (DILILLO et al., 2010) com a finalidade de mensurar a história de maus-tratos contra crianças e registrar o grau de severidade e a frequência em que ocorrem.

Atualmente, no Brasil, o governo federal fornece o Sistema de Informação para Infância e Adolescência (SIPIA) (SIPIA, 1998) para atender as demandas dos Conselhos Tutelares. O SIPIA tem como finalidade tratar as informações sobre a promoção e defesa dos direitos fundamentais previstos no ECA, e possibilitar a produção de conhecimentos específicos referentes as situações concretas de violações aos direitos, e ainda, suas respectivas medidas de proteção.

Apesar da existência do sistema SIPIA, o mesmo ainda não é amplamente utilizado. Isso pode ser comprovado ao se acessar a própria página do site <sup>1</sup>, na parte de estatísticas de violação por Estado, onde apenas 20 Estados contêm informações, sendo que o Estado do Tocantins não é encontrado nesta relação. Por esta razão, este trabalho visa contribuir com o tema em âmbito nacional, enquanto campo de estudo e prática, concentrou-se no município de Palmas, Tocantins.

## **Registro e Tratamento de Informações no Conselho Tutelar de Palmas**

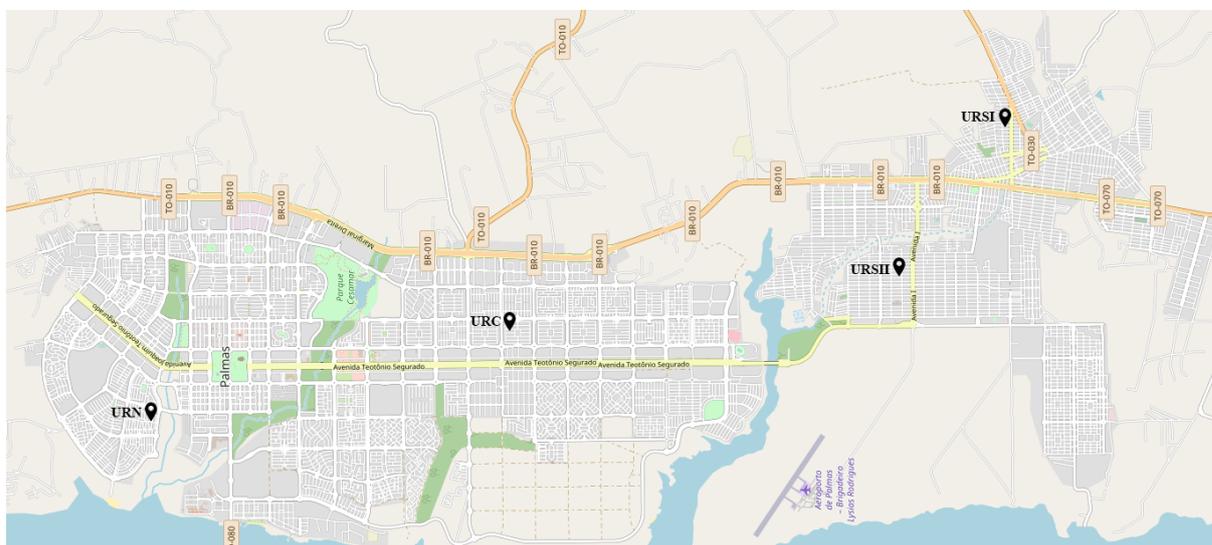
O Conselho Tutelar de Palmas é composto por quatro unidades conselheiras, onde cada unidade possui em sua totalidade cinco conselheiros e cinco suplentes, um para cada conselheiro, atuando em prol dos direitos das crianças e adolescentes previstos no ECA.

---

<sup>1</sup>[http://www.sipia.gov.br/CT/?x=h\\*llWBpL-NrFqpcDV4LbGg](http://www.sipia.gov.br/CT/?x=h*llWBpL-NrFqpcDV4LbGg)

A imagem 1.1 apresenta o mapa de Palmas indicando a localidade de cada unidade.

**Figura 1.1 – Localização das Unidades Conselheiras no município de Palmas, Tocantins.**



O agir do Conselho Tutelar ocorre em sua grande maioria por meio de denúncias, estas são feitas quando reconhecido fatos que configurem ameaça ou violação de direitos de crianças e adolescentes (TELEFÔNICA, 2016).

Recebida a denúncia, o Conselho Tutelar deve apurá-la imediatamente, sendo esta realizada no local da ocorrência. É recomendado que a incumbência seja realizada por pelo menos dois conselheiros, a fim de evitar entendimento distorcido ou parcial da situação que está sendo apurada.

Constatada a veracidade de uma denúncia, e sendo ela totalmente ou parcialmente procedente, o Conselho Tutelar tem em suas mãos um caso, para estudo, encaminhamento e acompanhamento (TELEFÔNICA, 2016). Cada caso é um caso e deve ser tratado como tal, tendo atendimento personalizado visando suas particularidades.

Em 2015, o programa jornalístico Bom Dia Tocantins, realizou uma reportagem<sup>2</sup> nos Conselhos Tutelares de Palmas, e constatou que o trabalho dos conselheiros eram precavidos de uma forma que otimizasse e auxiliasse em seus serviços, como por exemplo, um sistema computacional capaz de promover uma integração, além de permitir a sincronização de todas as informações registradas nas diversas unidades conselheiras, uma vez que, no atual modelo, um Conselho Tutelar não é capaz de reconhecer os atendimentos e processos efetuados por outra unidade.

Os conselheiros tutelares organizam os casos em pastas físicas, como pode ser visto na imagem 1.2, onde cada família é representada por uma pasta composta por seus aten-

<sup>2</sup>URL do vídeo da reportagem: <https://youtu.be/bRLBD3Wf-w8>

dimentos. Ao receber uma denúncia, seja ela de qual natureza for (disque 100, entidades governamentais ou não governamentais, parentes e etc), os conselheiros procuram saber se aquela criança/adolescente — que teve seu direito ameaçado ou violado — é reincidente, decidindo assim, se o atendimento que está sendo realizado será somado à sua respectiva pasta ou se uma nova será criada.

**Figura 1.2 – Armários contendo as pastas com os casos registrados na Unidade Conselheira da Região Centro e Sul II.**



De acordo com os dados do Conselho Tutelar de Palmas, no ano de 2017, mais de 2700 novas denúncias foram registradas juntando todas as unidades conselheiras, sem contar denúncias de casos reincidentes. No ano de 2018, até o mês de maio, cerca de 900 denúncias já foram apuradas. Todas essas informações foram registradas em formato impresso e guardadas em pastas físicas.

Segundo os Conselheiros, consultar casos existentes demandam bastante tempo e paciência, já que na atual realidade, não há um mecanismo ou ferramenta que possibilite agilizar este processo. Problemas como esses, podem ser resolvidos utilizando um *software*, assim como, viabilizar o compartilhamento de informações entre as unidades e interromper a duplicidade de dados.

## 1.2 Justificativa

Ao observar as possibilidades que sistemas de informação oferecem, tais como custo/benefício, segurança de dados, integração e considerável desempenho na realização de tarefas, verificou-se o grande potencial na utilização de *softwares web* em proporcionar o gerenciamento eficaz de informações. Partindo dessa premissa, de forma a auxiliar o Conselho Tutelar, foi desenvolvido e implantado no município de Palmas o sistema computacional intitulado “DATAGUARDIAN”, um sistema web de apoio à gestão em direitos humanos de crianças e adolescentes. Por ter sido desenvolvida em contato direto com

os conselheiros de Palmas, essa se torna uma ferramenta mais amigável, facilitando sua utilização e aceitação por parte dos usuários.

Como dito anteriormente, o dia a dia no Conselho Tutelar no cenário atual é precário no que se trata de meios que otimizem o gerenciamento das informações obtidas através das denúncias recebidas. O “DATAGUARDIAN” busca trazer benefícios na facilidade de efetuar consultas de casos cadastrados, geração de relatórios e manutenção dos dados.

Para demonstrar quantitativamente e qualitativamente as métricas do software, testes de disponibilidade, segurança e desempenho foram realizados, assim como avaliação de usabilidade com o propósito de avaliar os diferentes níveis de experiência do usuário com relação a fatores típicos como facilidade de aprendizado, facilidade de uso, eficiência de uso e produtividade.

Nesse contexto, a aplicação “DATAGUARDIAN” visa sobretudo agilizar os processos de registro de denúncias e aplicação de medidas protetivas aos casos atendidos no Conselho Tutelar de Palmas.

### 1.3 Objetivos

Este trabalho possui como objetivo geral, o desenvolvimento e a implantação de um sistema web que possibilite o registro e o tratamento de informações sobre a promoção e defesa dos direitos fundamentais previstos no ECA. A partir dele, tornar possível a sistematização das demandas dos Conselhos Tutelares, obter as informações desde o registro da denúncia até o estudo de caso, onde auxiliará o Conselheiro Tutelar a providenciar as medidas que levem ao ressarcimento do direito. Para tanto, os seguintes objetivos específicos foram traçados.

#### 1.3.1 Objetivos Específicos

1. Discriminar os processos realizados atualmente nos Conselhos Tutelares;
2. Identificar os problemas em tecnologia da informação atuais nos Conselhos Tutelares mirando em uma solução para suprir tais necessidades;
3. Analisar através de métricas a efetividade do desenvolvimento do software;
4. Implementar uma aplicação web para acompanhamento, atendimento, registros de denúncias e geração de relatórios analíticos referentes aos processos dos casos contidos no sistema;
5. Analisar através de métricas a implantação da solução do problema exposto neste trabalho;

6. Provar a proficiência do sistema no ambiente Conselho Tutelar no registro e tratamento das informações que previamente eram desempenhadas em documentos escritas à mão;
7. Identificar possíveis trabalhos futuros que podem acarretar na otimização do software desenvolvido.

#### 1.4 Estrutura do Trabalho

Esse trabalho foi dividido em seis capítulos. Neste primeiro capítulo são apresentados a contextualização do problema, os objetivos, tanto geral como específicos.

No capítulo 2 é apresentada a fundamentação teórica. É feita uma explanação a respeito dos conceitos utilizados para o desenvolvimento do sistema.

No capítulo 3 são apresentados os trabalhos relacionados: ferramentas que possuem características semelhantes ou que seguem o mesmo ramo de atividade do sistema tratado neste projeto, como também, trabalhos que possuem alguma semelhança.

O capítulo 4 descreve como foram aplicadas as técnicas que levaram ao desenvolvimento e implantação do *software*, como o levantamento de requisitos e os diagramas que serviram como base para a sua implementação, assim como as ferramentas que auxiliaram para sua concepção. É demonstrado também os métodos utilizados para a realização dos testes do sistema e sua avaliação de usabilidade.

O capítulo 5 é destinado aos resultados obtidos através dos testes realizados no sistema “DATAGUARDIAN”.

No último capítulo (6) Conclusões e Trabalhos Futuros, são observadas as conclusões obtidas com o desenvolvimento do trabalho e sugestões de trabalhos futuros.

## 2 FUNDAMENTAÇÃO TEÓRICA

O sistema “DATAGUARDIAN” foi desenvolvido para atender as demandas dos Conselhos Tutelares e implantado no Conselho Tutelar de Palmas com o objetivo de tornar mais eficiente e eficaz a forma como os conselheiros desempenham suas atividades.

Este capítulo apresenta os principais conceitos necessários para o desenvolvimento do sistema. Além disso, é explanado acerca de usabilidade em sistemas e de testes de *software* para fundamentar a finalidade deste trabalho.

### 2.1 Metodologia de Desenvolvimento de Software

De acordo com (REZENDE, 1997) uma metodologia pode ser descrita como uma abordagem organizada que visa atingir um objetivo através de passos pré-definidos. Trata-se de um roteiro, um processo dinâmico e iterativo para desenvolvimento estruturado de projetos, sistemas ou *softwares*, tendo como metas a qualidade, produtividade e efetividade de projetos.

Uma metodologia de *software* é mais que uma técnica e deve auxiliar o desenvolvimento de sistemas. A mesma pode ser entendida como um instrumento que possibilite um planejamento metódico, de maneira a atender às necessidades do cliente, com os recursos disponíveis e dentro de um prazo ideal.

A metodologia de desenvolvimento pode ser dividida em fases, também conhecidas como ciclo de vida de sistema ou processo de *software*. De maneira sintética, tais fases podem ser definidas como:

- Estudo preliminar: pode ser entendida como a visão preliminar do projeto. É nela que se faz a primeira definição dos requisitos funcionais, objetivos, abrangências, integrações, limitações, impactos e áreas envolvidas;
- Análise do sistema: nada mais é que a visão atual do sistema, relatando os requisitos funcionais atuais;
- Projeto lógico: é o desenho da lógica do projeto, definição de propostas de soluções e dos requisitos funcionais;
- Projeto físico: compreende a execução, confecção do programa e seus testes, além dos layouts das entradas e saídas;
- Projeto de implantação: trata-se da implantação dos sistema, treinamento e capacitação do cliente ao usuário.

Para o desenvolvimento do “DATAGUARDIAN” foi aplicada uma metodologia, onde através dela foi feita a análise dos requisitos após entrevistas e reuniões com os conselheiros tutelares do Conselho Tutelar de Palmas. No processo de desenvolvimento de *software*, o levantamento de requisitos compreensíveis por todas as partes envolvidas no desenvolvimento do sistema (clientes, analistas, desenvolvedores, etc.) é um fator básico e extremamente importante, evitando falhas no entendimento do problema a ser solucionado. A obtenção de requisitos dentro do contexto da organização deve ser realizada de forma adequada, com métodos, técnicas e ferramentas que deem suporte à etapa do processo de desenvolvimento. Para isso, dentro do contexto de Engenharia de Requisitos (ER), a representação dos requisitos tem papel fundamental na condução das demais atividades desse processo (FRANCETO, 2005).

Um requisito é uma característica do sistema ou a descrição de algo que o sistema é capaz de realizar para atingir os seus objetivos (PFLEEGER, 2004). São levantados durante a fase de concepção do projeto para que se obtenha uma visão geral do que será desenvolvido. Os requisitos levantados dizem o que o sistema deverá fazer, mas não como fazer. Para este fim, existem outros diagramas para descrever como as funcionalidades se comportam.

## 2.2 Metodologia de Desenvolvimento Ágil

Metodologias ágeis têm sido apontadas como uma alternativa às abordagens tradicionais para o desenvolvimento de software (SOARES, 2014). O motivo é que em metodologia ágil de *software* o ambiente das organizações são mais dinâmicos, os projetos são propensos a mais mudanças, onde os requisitos são passíveis de alterações. Equipes são pequenas, datas de conclusões de atividades são curtas e o desenvolvimento rápido é fundamental. Além disso, a preocupação com tempo gasto em documentação é consideravelmente menor, focando mais nas soluções de problemas de forma iterativa.

Esta seção tem como objetivo trazer os conceitos da metodologia ágil *Scrum*, a qual foi utilizada para a concepção do sistema proposto.

### O Manifesto Ágil

O termo “Métodos Ágeis” tornou-se popular em 2001 quando 17 especialistas em desenvolvimento de software representando diversas metodologias existentes como *XP*, *Scrum*, *Crystal* e outras estabeleceram princípios comuns a todas elas, resultando na criação da “Aliança Ágil”, surgindo então o Manifesto Ágil<sup>1</sup>, onde os conceitos chave são descritos a seguir (PRIKLADNICKI; WILLI; MILANI, 2013):

- **Indivíduos e interações** mais que processos e ferramentas;

---

<sup>1</sup><http://agilemanifesto.org/>

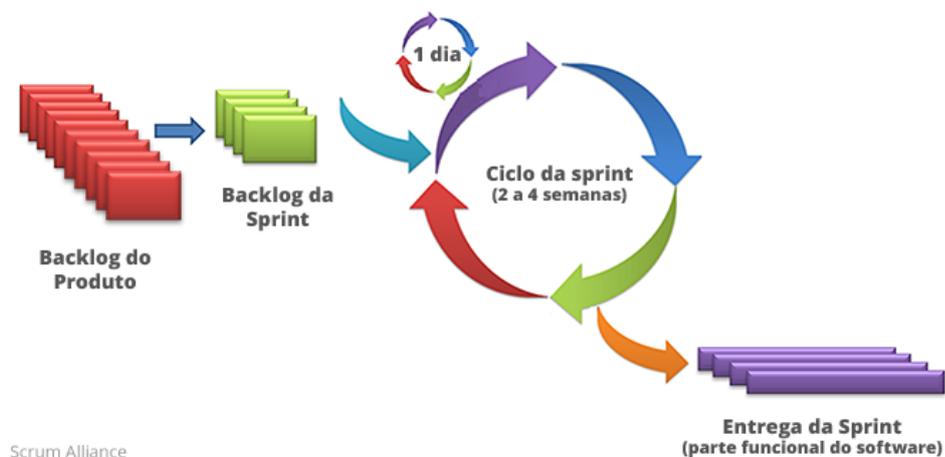
- **Software funcionando** mais que documentação abrangente;
- **Colaboração com cliente** mais que negociação de contratos; e
- **Responder a mudanças** mais que seguir um plano.

A engenharia de software ágil é fruto da combinação de uma filosofia e de um conjunto de diretrizes de desenvolvimento. A filosofia empregada encoraja a satisfação do cliente, além da entrega incremental do *software* logo de início. As equipes de projeto devem ser pequenas, com alta motivação e podendo aplicar métodos informais na edificação do projeto. Soma-se a isso, uma simplicidade global do desenvolvimento. O conjunto de diretrizes, por sua vez, enfatizam a entrega em contraposição à análise e ao projeto, além da comunicação ativa e contínua entre desenvolvedores e clientes. Nesse ponto vale ressaltar que as atividades de análise e projeto não são totalmente descartadas, nem desencorajadas nessa metodologia (PRESSMAN, 2006).

### Metodologia *Scrum*

Seguindo os princípios do manifesto ágil, a metodologia *Scrum* têm como intuito guiar as atividades estruturais de desenvolvimento de *software*. É importante entender que o *Scrum* não se trata de um processo previsível, definindo o que fazer em toda circunstância. Em problemas complexos onde não é possível prever com exatidão tudo que irá ocorrer, o *Scrum* é aplicado de forma a tornar o projeto mais flexível, permitindo que ajustes sejam realizados certificando que os objetivos sejam atingidos (SCHWABER, 2004).

Figura 2.1 – Ciclo da Metodologia *Scrum*. Fonte: (SISTEMAS, 2014).



A figura 2.1 apresenta o ciclo de atividades que compõem a metodologia *Scrum*. O *Scrum* baseia o progresso de desenvolvimento em iterações chamadas de *Sprints*. Em cada

*sprint*, o trabalho é priorizado a partir de uma lista de requisitos, chamada de *Backlog do Produto* (PRIKLADNICKI; WILLI; MILANI, 2013). Assim, o desenvolvimento das funcionalidades são realizados de acordo com sua superioridade e valor para o cliente. Ao finalizar cada *sprint*, um conjunto de funcionalidades prontas são entregues. A figura 2.2 demonstra um exemplo simplificado de um *Backlog de Produto*.

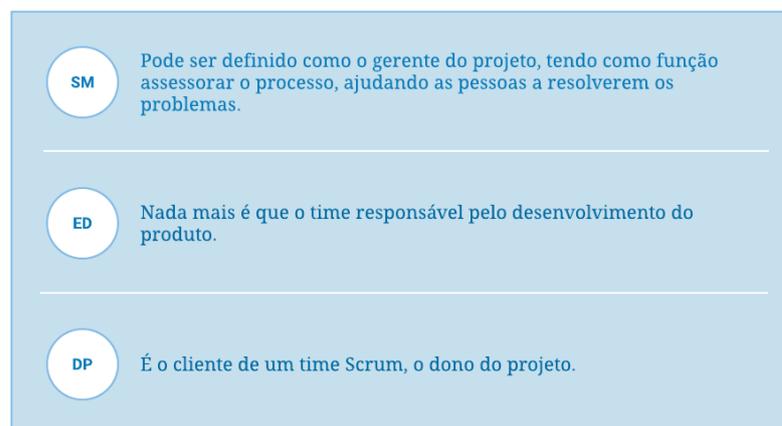
**Figura 2.2 – Exemplo de Backlog do Produto. Fonte: Adaptado de (PRIKLADNICKI; WILLI; MILANI, 2013).**

Backlog do Produto: Twitter
Login de usuários já cadastrados
Cadastrar novo usuário
Tuitar
[...]

### Papéis do *Scrum*

No *Scrum* não há simplesmente um responsável pelo projeto. Sua gestão é dividida em três papéis<sup>2</sup>: Dono do Produto (DP), *Scrum Master* (SM) e a Equipe de Desenvolvimento (ED) (PRIKLADNICKI; WILLI; MILANI, 2013), conforme imagem 2.3.

**Figura 2.3 – Definição dos papéis no Scrum.**



### *Backlog da Sprint*

O *Backlog da Sprint* é gerado a partir das histórias retiradas do *Backlog do Produto* e que serão implementadas durante o *Sprint*. O seu objetivo é tornar visível o trabalho

<sup>2</sup>Vale ressaltar que papéis não são cargos.

necessário para que a Equipe de Desenvolvimento atinja a meta da Sprint.

**Figura 2.4 – Backlog da Sprint. Fonte: Adaptado de (PRIKLADNICKI; WILLI; MILANI, 2013).**

Backlog da Sprint		
META DA SPRINT: USUÁRIOS PODERÃO ESTAR NO TWITTER		
Login de usuários já cadastrados	Ativar login com usuário	Montar layout do box de login
	Mostrar plano de segurança	Testar integrado
	Estruturar log	Revisar código
	Criar comportamento de login	Inserir hint explicativo de funcionamento
	Atualizar documentação técnica	(...)
Cadastrar novo usuário	Criar tabelas no banco de dados	Estruturar persistência
	Definição sobre uso de templates	Quando validado, ativar usuário
	Escrever teste	Definir padrões para cadastro
	Atualizar documentação técnica	Validar e-mail cadastrado
	Testar integrado	(...)

Para isso, os membros da equipe podem tanto adicionar como remover tarefas caso julguem necessário (PRIKLADNICKI; WILLI; MILANI, 2013). A imagem 2.4 mostra o possível resultado do *Backlog da Sprint* planejado a partir dos dois primeiros itens do *Backlog do Produto* demonstrados na imagem 2.2.

## Incremento do Produto

Ao final de cada *Sprint*, a Equipe de Desenvolvimento entrega um incremento do produto, resultado do que foi produzido durante a *Sprint* (PRIKLADNICKI; WILLI; MILANI, 2013).

Para se atingir tal objetivo, realiza-se uma reunião para validação daquilo que será entregue (*Sprint Review*), onde o cliente pode verificar se o objetivo do *Sprint* foi realmente alcançado. Logo após, é realizado apenas para o time, uma reunião (*Sprint Retrospective*), onde o *Sprint* é avaliado sob a perspectiva de processo, time ou produto, identificando os acertos e os erros a fim de otimizar o processo (RUBIN, 2012).

Com base no que foi apresentado é possível concluir que o *Scrum* é um modelo de implementação simplificado, contendo poucas especificações e artefatos, sendo seu maior desafio a adaptação às necessidades do cliente, do time e da empresa. Devido a tais características, o *Scrum* foi escolhido como método de gerenciamento do projeto.

## 2.3 Usabilidade

De acordo com (BENYON, 2011a) a usabilidade sempre foi a principal busca da interação humano-computador (IHC). A definição original de usabilidade é de que os sistemas devem ser fáceis de aprender, flexíveis e devem despertar nas pessoas uma boa atitude (BENYON, 2011a). A ISO (International Organization for Standardization)<sup>3</sup> por meio da norma 9241-210 (STANDARDIZATION, 2010), define usabilidade como “[...] a medida na qual um produto pode ser usado por usuários específicos para alcançar objetivos específicos com efetividade, eficiência e satisfação em um determinado contexto de uso.”.

Através de uma avaliação de usabilidade pode-se verificar se o sistema realmente atende aos requisitos e se comporta da forma que lhe é esperado pelo usuário, permitindo assim, conhecer o seu grau de usabilidade.

Segundo (BENYON, 2011a), um sistema com alto grau de usabilidade terá as seguintes características:

- será eficiente no sentido de que as pessoas poderão fazer coisas mediante uma quantidade adequada de esforço;
- será eficaz no sentido de que conterá as funções e o conteúdo de informações adequadas e organizadas de forma apropriada;
- será fácil aprender como fazer as tarefas e será fácil lembrar como fazê-las após algum tempo;
- será seguro de operar na variedade de contextos em que será usado; e
- terá um alto grau de utilidade no sentido de que realizará tarefas que as pessoas querem que sejam realizadas.

Para alcançar a usabilidade, o design deve ser realizado centrado no humano<sup>4</sup> e adotar uma abordagem de design na qual a avaliação seja fundamental (BENYON, 2011a).

### Princípios de Design

No decorrer dos anos muitos princípios de design foram desenvolvidos gerando certos padrões e diretrizes em sistemas interativos, como por exemplo o comando “Desfazer”, botão “Voltar” em um site, ou botões ou campos de formulários de sites acinzentados informando ao usuário que tal opção está desabilitada. Don Norman em seu livro O Design do Dia-a-Dia (NORMAN, 2006), apresenta seis princípios de design que ajudam a

---

<sup>3</sup><https://www.iso.org/standard/52075.html>

<sup>4</sup>Ser centrado no humano é colocar as pessoas em primeiro lugar; é projetar sistemas interativos que favoreçam as pessoas e dos quais elas possam usufruir.(BENYON, 2011b)

entender por que alguns produtos satisfazem os consumidores, enquanto outros os deixam completamente frustrados, que são eles:

- **Visibilidade** — as funções mais visíveis são aquelas em que o usuário provavelmente será capaz de saber o que fazer em seguida. Do contrário, quando as funções estão fora de vista, tornam-se mais difíceis de encontrar, e conseqüentemente de saber como usá-las. É mais fácil reconhecer coisas do que lembrar delas.
- **Retorno (*feedback*)** — o *feedback* é a informação retornada em razão das ações realizadas pelo usuário. O retorno constante e consistente intensificará a sensação de controle.
- **Restrições** — proporcionar restrições de forma que as pessoas não tentem fazer coisas inadequadas. Em especial, as pessoas devem ser impedidas de realizar erros graves por meio de restrições.
- **Mapeamento** — termo técnico que significa o relacionamento entre duas coisas, neste caso, entre os controles e seus movimentos, e os resultados dessa relação no mundo. O controle é aprimorado se houver um mapeamento claro entre os controles e seus efeitos.
- **Consistência** — operações similares com elementos similares para realizar tarefas semelhantes. Em particular, ser consistente é utilizar das características de design e métodos-padrão de trabalho.
- **Affordance** — o design deve ser criado de forma clara para que se saiba para o quê os objetos servem. Refere-se ao atributo de um objeto que permite que as pessoas saibam como usá-lo, por tão óbvio que é, ou pelo seu visual sugerir que é fisicamente possível. Por exemplo, botões propiciam ser apertados, cadeiras propiciam que se sente nelas e campos de formulário propiciam serem preenchidos.

## Teste de Usabilidade

A usabilidade refere-se à qualidade da interação de sistemas com os usuários. Algumas destas características relevantes são a facilidade de aprendizado do sistema, as facilidades de uso, a satisfação do usuário, flexibilidade e a produtividade.

Antes de afirmar que um *software* está pronto para ser utilizado, é importante saber se ele apoia adequadamente os usuários nas suas tarefas e no ambiente em que será utilizado. Da mesma forma que testes de funcionalidade são necessários para verificar a robustez da implementação, a avaliação de interface é essencial para analisar a qualidade de uso de um *software* (KARAT, 1990).

O teste de usabilidade é feito executando experimentos com os usuários, e o valor de cada um destes fatores é medido, para tentar obter alguns fatores de interesse. Para isto, toma o usuário como base em seus estudos, e com isto o design conclui-se os valores alcançados são satisfatórios ou não. Os testes identificarão potenciais problemas uma vez que uma versão estável da tecnologia esteja disponível (BENYON, 2011c). O importante é que revelando os problemas ainda haja tempo de resolvê-los. A imagem 2.5 demonstra uma estratégia para realizar testes de usabilidade em um sistema interativo.

**Figura 2.5 – Planejamento para testes de usabilidade.**



A primeira etapa se dá pela realização dos testes, depois são levantados os problemas encontrados. Posteriormente esses problemas são priorizados para serem solucionados, e suas possíveis soluções são determinadas, seguindo das devidas implementações para a realização de novos testes. Este ciclo é reproduzido até que todos os problemas de usabilidade sejam eliminados.

### **Avaliação Heurística**

O método de avaliação heurística é um método analítico que visa identificar problemas de usabilidade conforme um conjunto de heurísticas ou diretrizes (*guidelines*) (NIELSEN, 1994). Este método não envolve usuários, e deve ser realizado por avaliadores especialistas. Em geral, recomenda-se que 3 a 5 especialistas<sup>5</sup> realizem uma avaliação heurística. A imagem 2.6 mostra o procedimento que deve ser realizado para a avaliação heurística.

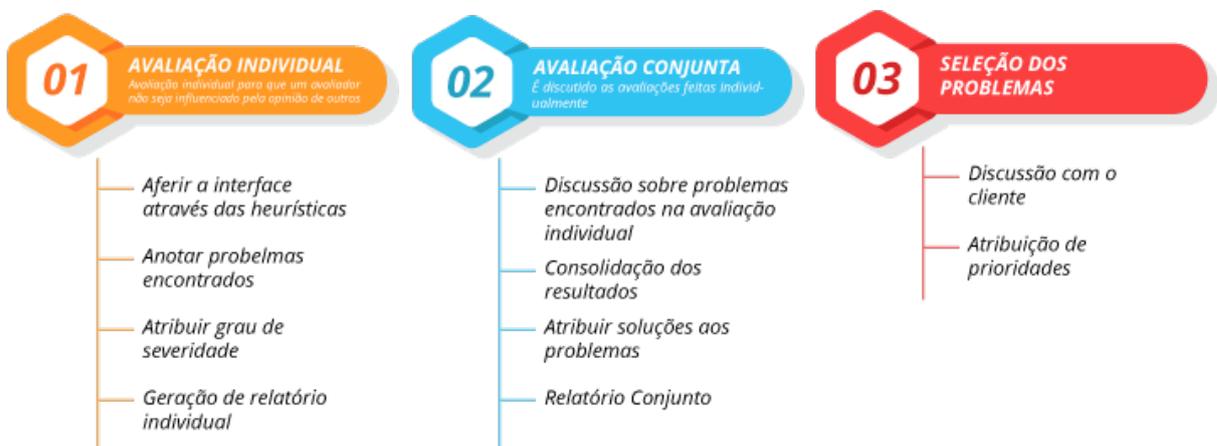
Na primeira fase acontecem sessões curtas (1 a 2 horas) de avaliação individual, pois desta forma, um avaliador não é influenciado por opiniões alheias. No decorrer de cada sessão, o avaliador percorre a interface inspecionando os diversos elementos de interface e

<sup>5</sup>Nielsen chegou a esse número baseado em uma pesquisa que levava em consideração o número de avaliadores, o número de erros encontrados e o valor gasto.(NIELSEN, 2000)

comparando-os com a lista de heurísticas de usabilidade. Na segunda fase, cada avaliador tem acesso aos relatórios individuais elaborados na etapa anterior, pode-se expressar seu julgamento sobre os problemas apontados pelos outros avaliadores. Ao final, deve-se gerar um relatório unificado e consolidado sobre os problemas encontrados. Na terceira e última fase, é realizada uma análise de custo/benefício das correções aos problemas encontrados. Esta análise deve levar em consideração não apenas a gravidade dos problemas, como também os prazos, orçamentos e a capacitação da equipe de desenvolvimento.

**Figura 2.6 – Como funciona uma avaliação heurística de usabilidade.**

### Como a avaliação heurística funciona?



Jakob Nielsen criou o que são chamados de: 10 heurísticas de Nielsen (NIELSEN, 1993). Cada elemento de interface (ou conjunto de elementos) deve ser analisado para verificar sua conformidade com cada uma das seguintes heurísticas:

- **Visibilidade de qual estado estamos no sistema** — é responsabilidade do sistema informar o que está acontecendo em tempo real para o usuário.
- **Correspondência entre o sistema e o mundo real** — em relação ao mundo real pode-se considerar sons, visual e o tom de escrita que o usuário utiliza para se comunicar. Esta heurística é bastante implementada quando utiliza-se uma seta, ícones e a cor vermelha para elementos negativos.
- **Liberdade de controle fácil pro usuário** — fornecer aos usuários alternativas e “saídas de emergências”; possibilidades de desfazer aquilo que foi feito, com exceção das regras que vão contra o negócio ou interferem em outra funcionalidade.
- **Consistência e padrões** — é importante manter a consistência e padrão visual (texto, cor, desenho do elemento, som e etc).

- **Prevenções de erros** — tentar evitar que erros aconteçam, informando ao usuário sobre as consequências de suas ações, ou se possível, impedindo ações que levariam a uma situação de erro.
- **Reconhecimento em vez de memorização** — o usuário não tem obrigação de decorar qual foi o caminho que ele fez para chegar a determinado lugar. Uma boa estratégia é o uso de *breadcrumb*<sup>6</sup>.
- **Flexibilidade e eficiência de uso** — oferecer aos usuários aceleradores e caminhos alternativos para realizarem a mesma tarefa. Deixar uma boa experiência do sistema desde o usuário mais leigo até o mais avançado.
- **Estética e design minimalista** — toda informação extra deixada para o usuário pode na verdade adicionar mais uma dúvida, ou seja, tornar o layout e o conteúdo o mais simples e direto possível.
- **Ajude os usuários a reconhecerem, diagnosticarem e recuperarem-se de erros** — as mensagens de erros tem que ser claras e próximas do conteúdo ou ação que causou o erro.
- **Ajuda e documentação** — devem ser fáceis de buscar, focados no domínio e na tarefa do usuário, devem listar passos concretos a serem efetuados para atingir seus objetivos.

Para cada heurística avaliada e constatado problema, deve-se definir a localização do problema — na interface — e sua gravidade. Com relação à localização, o problema pode ter sido encontrado nos seguintes locais:

- em um único local na interface;
- em dois ou mais locais na interface, casualmente;
- na estrutura geral da interface, de forma sistemática; ou
- pode ser algo inexistente, ou seja, precisa ser incluído na interface.

Quanto a gravidade do problema, é calculada por cada especialista como uma combinação dos fatores **frequência**, **impacto** e **persistência do problema**. Seu valor pode ser definido atribuindo uma nota que varia de 0 a 4, conforme a tabela 2.1.

O quadro 2.7 mostra uma exemplo de relatório gerado a partir de uma avaliação heurística em um site de quadro de avisos elaborado por (PRATES; BARBOSA, 2003).

---

<sup>6</sup>indicam a localização atual do usuário e quais são os níveis superiores da estrutura

Tabela 2.1 – Níveis de gravidade dos problemas

Gravidade	Descrição
0	Não é encarado necessariamente como um problema de usabilidade
1	Problema estético. Não necessita ser corrigido, a menos que haja tempo disponível
2	Problema menor de usabilidade. Baixa prioridade para sua correção
3	Problema maior de usabilidade. Alta prioridade em sua correção
4	Catástrofe. Imperativo corrigi-lo o mais rápido possível

Figura 2.7 – Exemplo de problemas encontrados a partir de avaliação heurística. Fonte: (PRATES; BARBOSA, 2003).

<p><b>Problema:</b> O usuário não conseguirá entender que o texto “privativo da comunidade” lhe dá acesso a um espaço com mais funcionalidades do que aquele em que ele se encontra.</p> <p><b>Heurística violada:</b> correspondência entre o sistema e o mundo real</p> <p><b>Explicação:</b> Embora na sede da ASCR tenha alguns espaços que normalmente só são acessíveis por membros da comunidade, o usuário não utiliza a palavra “privativo” no seu cotidiano e não saberá a que ela se refere.</p> <p><b>Gravidade:</b> 4 – catastrófico. O usuário não conseguirá acessar as funcionalidades que estão disponíveis apenas para membros, como por exemplo ler avisos específicos ao trabalho em que está envolvido, ou criar um novo aviso.</p>
<p><b>Problema:</b> O texto “Quadro geral” não transmite a idéia do que está sendo visualizado</p> <p><b>Heurística violada:</b> reconhecimento</p> <p><b>Explicação:</b> O que está sendo mostrado na seção denominada Quadro Geral são os avisos do Quadro de Avisos que foram colocados em destaque</p> <p><b>Gravidade:</b> 3 – grave. Como os usuários na sua maioria têm pouca experiência com informática, pode não ficar claro para eles que os avisos no Quadro geral são aqueles selecionados para estarem em destaque e podem aparecer também em outras seções. Isto pode comprometer o entendimento do usuário sobre como utilizar o Quadro de Avisos.</p>

### SUS (*System Usability Scale*)

Dentre os métodos para medir usabilidade de sistemas interativos, destaca-se o SUS<sup>7</sup>, uma ferramenta que oferece uma visão global das avaliações subjetivas de usabilidade. Originalmente criado por John Brooke em 1986 (BROOKE, 1986a), o SUS é uma das escalas de avaliação de usabilidade padronizadas mais conhecidas, com referências em mais de 1300 artigos e publicações (AFFAIRS, 2013).

O questionário consiste de 10 perguntas, e para cada uma delas o usuário assinala sua resposta em uma escala Likert<sup>8</sup> de 1 a 5, onde 1 significa Discordo Completamente e 5 significa Concordo Completamente, como pode-se observar na figura 2.8.

<sup>7</sup><https://www.usability.gov/how-to-and-tools/methods/system-usability-scale.html>

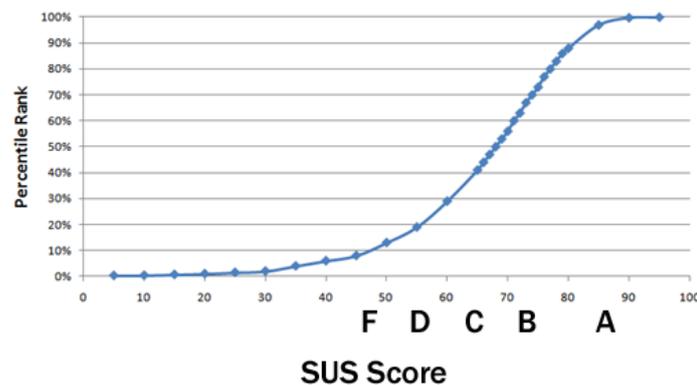
<sup>8</sup>é um tipo de escala de resposta psicométrica usada habitualmente em questionários, e é a escala mais usada em pesquisas de opinião (WIKIPÉDIA, )

**Figura 2.8 – Formato de Resposta do SUS. Fonte: (TEIXEIRA, 2015).**

Strongly Disagree 1	2	3	4	Strongly Agree 5
○	○	○	○	○

O resultado do SUS é a soma da contribuição individual de cada item. Para as afirmativas ímpares (1, 3, 5, 7 e 9) deve-se subtrair 1 à resposta do usuário, ao passo que para as afirmativas pares (2, 4, 6, 8 e 10) a pontuação é 5 menos a resposta do usuário, ou seja, se o participante tiver respondido 2, é contabilizado 3, caso tenha respondido 4, contabiliza-se 1. Depois de obter a pontuação de cada item, realiza o somatório de todas as pontuações e multiplica-se o resultado por 2,5 (BROOKE, 1986b). Desta forma, o resultado obtido será um índice de satisfação do utilizador (que varia de 0 a 100).

**Figura 2.9 – Demonstração do SCORE do SUS. Fonte: (SAURO, 2011b).**



De acordo com Jeff Sauro em sua pesquisa (SAURO, 2011a), foram analisados dados de mais de 5.000 usuários em 500 avaliações de usabilidade diferentes. Esses dados mostraram que o SUS é uma medida confiável e válida de usabilidade.

A pontuação média do SUS de todos os 500 estudos é 68. Uma pontuação do SUS acima de 68 seria considerada acima da média e qualquer coisa abaixo de 68 estaria abaixo da média. O gráfico 2.9 mostra como os níveis percentuais se associam às pontuações do SUS e das letras. A letra “F” representa a pior nota, enquanto a letra “A”, a melhor nota.

No Capítulo de Metodologia serão apresentados como foram realizados os testes de usabilidade, tanto a avaliação heurística como a aplicação do questionário SUS. Posteriormente, no capítulo de Resultados e Discussão, serão apresentados os resultados obtidos com a finalidade de expor os pontos positivos e negativos quanto à interatividade do usuário com o sistema desenvolvido.

## 2.4 Testes de *Software*

O *software* desempenha um importante papel no apoio aos negócios das empresas, sendo em algumas delas parte intrínseca. Essa importância tende a crescer em um mundo em que as atividades e produtos tendem a depender cada vez mais do *software* (RIOS; MOREIRA, 2013).

Muitas vezes os usuários acabam abandonando o site, insatisfeitos com a sua usabilidade e/ou funcionalidade, causando enormes prejuízos à imagem da organização ou financeiros. As razões é que as aplicações estão ficando cada vez mais complexas. Atualmente, um *software* tem que se integrar com muitos ambientes e possui inúmeros componentes que precisam se comunicar entre si. Desenvolver sistemas de informação está cada vez mais difícil e, como consequência disso, a atividade de testar está cada vez mais complexa (RIOS; MOREIRA, 2013).

O teste do *software* é o processo com o objetivo de avaliar o seu comportamento baseado no que foi especificado (RIOS; MOREIRA, 2013). Os testes também servem para se obter medidas de requisitos não funcionais do software, tais como confiabilidade ou desempenho, usando-se técnicas estatísticas apropriadas (MARTINS, 2005).

### Desempenho de *Software*

Entre os diferentes tipos de testes existentes, destacam-se os testes de desempenho ou teste de performance. Esse teste consiste em avaliar a capacidade de resposta, robustez, disponibilidade, confiabilidade e escalabilidade de uma aplicação, conforme a quantidade de conexões simultâneas, avaliando seu desempenho em alta carga de trabalho e considerando seu comportamento em circunstâncias normais (TESTARME, 2015).

Dentre os diferentes tipos de testes de desempenho, pode-se considerar os mais relevantes:

- Teste de carga – visam verificar o comportamento do sistema com uma quantidade de usuários (reais ou virtuais) previamente estabelecida. Busca-se estabelecer um número estimado de usuários o mais próximo das condições reais possível;
- Teste de *stress* – o teste de stress tem como objetivo explorar os limites da aplicação, aumentando-se a carga indefinidamente.

Buscando avaliar a capacidade e disponibilidade em situações extremas ou somente verificar o tempo de resposta de determinada funcionalidade da aplicação, torna-se necessário a utilização de ferramentas que permitem a simulação do uso e a devida documentação dos resultados obtidos. A realização dos testes em aplicações *web* podem ser alcançadas utilizando das diversas ferramentas existentes, dentre as quais pode-se citar:

- *PageSpeed Tools*– pontua as falhas identificadas pelo teste e sugere as possíveis soluções para contorná-las. (<<https://developers.google.com/speed/pagespeed/insights/>>. Acessado em: 02 Março 2018)
- *GTmetrix*– possibilita a escolha do país/cidade que serão feitos os testes, além do tipo de navegador, e permite ainda a comparações entre dois sites. Algumas recomendações de alterações são apresentadas ao final do teste. (<<https://gtmetrix.com/>>. Acessado em: 02 Março 2018)
- *Apache JMeter*– é um *software* de código aberto, desenvolvido na linguagem Java, utilizado para a realização de testes comportamentais, sendo possível mensurar o desempenho do conteúdo testado. Ele possui a capacidade de simular alta carga em um servidor (grande quantidade de usuários e iterações), grupo de servidores, redes ou objeto para aferir sua robustez ou para avaliar o desempenho sob diferentes tipos de carga. (<<http://jmeter.apache.org/>>. Acessado em: 02 Março 2018)

## Segurança de *Software*

O Teste de Segurança tem como objetivo garantir que o funcionamento da aplicação esteja exatamente como especificado. Verifica também se o *software* se comporta adequadamente mediante as mais diversas tentativas ilegais de acesso, visando possíveis vulnerabilidades. Para isso, testa se todos os mecanismos de proteção embutidos na aplicação de fato a protegerão de acessos indevidos (ELIZA; LAGARES, 2013).

É muito comum que as aplicações se tornem alvo de sujeitos que buscam provocar ações que possam prejudicar ou, até mesmo, beneficiar pessoas. Em função de situações como estas, o Teste de Segurança propõe demonstrar se a aplicação faz exatamente o que deve fazer ou se a aplicação não faz o que não deve ser feito (ELIZA; LAGARES, 2013).

Conforme Bastos (BASTOS et al., 2007), esses testes são necessários para garantir a confidencialidade das informações e a proteção dos dados contra o acesso indevido de terceiros. Para Bartié (BARTIÉ, 2002), ataques à segurança de *software* podem ter origens internas ou externas, aumentando ainda mais o grau de atenção que se deve dar para esses tipos de testes.

Um ponto determinante em segurança de *software* é a segurança dos dados. Portanto, é essencial que políticas de salvaguarda sejam aplicadas nas organizações, como rotinas de *backup* visando a prevenção de incidentes e extravio destas informações.

Pode-se perceber a importância da realização de testes de *software* para avaliação da qualidade e eficiência de aplicações. Tendo os parâmetros de testes bem definidos, a confiabilidade dos mesmos tende a aumentar, elevando-se a classificação positiva das aplicações pelos usuários (JÚNIOR, 2017).

### 3 TRABALHOS RELACIONADOS

Neste capítulo serão apresentadas algumas iniciativas e ferramentas de tratamento de informações pertinente a órgãos de assistência social e que também apresentam características semelhantes ao sistema proposto. Serão expostos suas principais características, como pontos positivos e limitações. Serão apresentados também, artigos e trabalhos realizados cujo o foco possui afinidade com o este trabalho de conclusão de curso.

#### 3.1 SIPIA

Desenvolvido em 1997, o SIPIA é um sistema nacional de registro e tratamento de informações sobre a garantia e defesa dos direitos fundamentais preconizados no ECA. Tal sistema é composto pelos módulos: Módulo I SIPIA CT – violação de direitos a partir de denúncias coletadas por conselhos tutelares; Módulo II SIPIA SINASE – atendimento ao adolescente em conflito com a lei; Módulo III – inserção familiar e adoção nacional e internacionais e Módulo IV – cadastro dos conselhos tutelares e fundos para infância e adolescência dos municípios brasileiros. Apesar de bastante completo o mesmo ainda não foi amplamente implantado em todos os Estados do país, como pode ser observado na própria página do sistema, mais especificamente em estatísticas por Estado <sup>1</sup>(SIPIA, 1997).

Figura 3.1 – Telas do sistema SIPIA. Fonte: (SIPIA, ).

A imagem apresenta duas telas do sistema SIPIA. A tela à esquerda é o menu principal, com o título 'SIPIA - Sistema de Informações para Infância e Adolescência' e 'Módulo para Conselhos Tutelares'. O menu contém links para 'Página Inicial', 'Sobre o Sipi', 'Estatísticas', 'Denúncia On-line' e 'Observatório Nacional'. Abaixo, há uma seção de 'Acesso ao SIPIA-CT Web' com campos para 'Usuário:' e 'Senha:', botões 'Esqueci minha senha', 'Entrar' e 'Limpar', e links para 'Fale conosco' e 'Versão 2.4.0'. A tela à direita é a interface de registro de uma denúncia, com o título 'Denúncia'. Ela contém campos para 'Data de registro:' (17/05/2017 17:04:56), 'Identidade Provisória:', 'Origem da denúncia:' (menu suspenso), 'Descrição da ocorrência:' (campo de texto com 'Mínimo 30 caracteres'), 'Local da ocorrência:', 'Ponto de Referência:', 'UF/Município:' (MA - Alto Alegre do Maranhão), 'Bairro:' (CENTRO), 'Data da ocorrência:', 'Conselho Tutelar:' (Conselho Tutelar de Aldeia), e um campo 'Digite o texto:' com o logotipo 'api'. Botões 'Limpar', 'Voltar' e 'Salvar' estão no canto inferior direito. Abaixo da interface, há uma barra de estatísticas com os seguintes dados: 'Denúncias On-line no mês: 49', 'Total de Denúncias no mês: 5070', 'Conselhos Tutelares Ativos: 5043' e 'Visitas ao Portal SipiCT: 22174279'. Na base da tela, há logos da 'Observatório Nacional dos Direitos de Criança e do Adolescente', 'CONANDA Conselho Nacional de Políticas de Criança e do Adolescente', 'Secretaria de Direitos Humanos' e 'BRASIL PAÍS RICO E PAÍS SEM POBREZA'.

<sup>1</sup>[http://www.sipia.gov.br/CT/?x=h\\*llWBpL-NrFqpcDV4LbGg](http://www.sipia.gov.br/CT/?x=h*llWBpL-NrFqpcDV4LbGg)

## Pontos Positivos

- Registro externo de denúncias online;
- Acompanhamento de situação da denúncia;
- Simplicidade para consultar denúncias registradas;
- Módulo de treinamento para conselheiros.

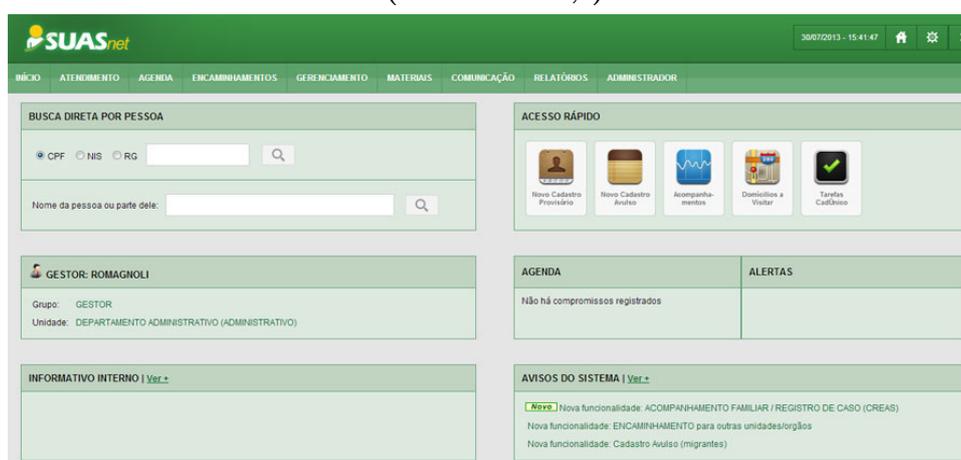
## Pontos Negativos

- Simples e com baixo número de funcionalidades;
- Interface antiquada se comparado com sistemas web mais atuais;
- Não possui *layout* responsivo.

### 3.2 SUASnet

O SUASnet<sup>2</sup> é uma ferramenta para o Gerenciamento da Assistência Social de municípios, integrando suas unidades e agilizando o dia-a-dia do atendimento ao público (SUASNET, 2006). Por se tratar de um sistema privado, as características descritas aqui foram retiradas do próprio site do sistema.

Figura 3.2 – Tela do Módulo de Conselhos Tutelares do SUASnet. Fonte: (SUASNET, ).



<sup>2</sup>SUASnet - <https://www.suasnet.com.br/site/?p=home>

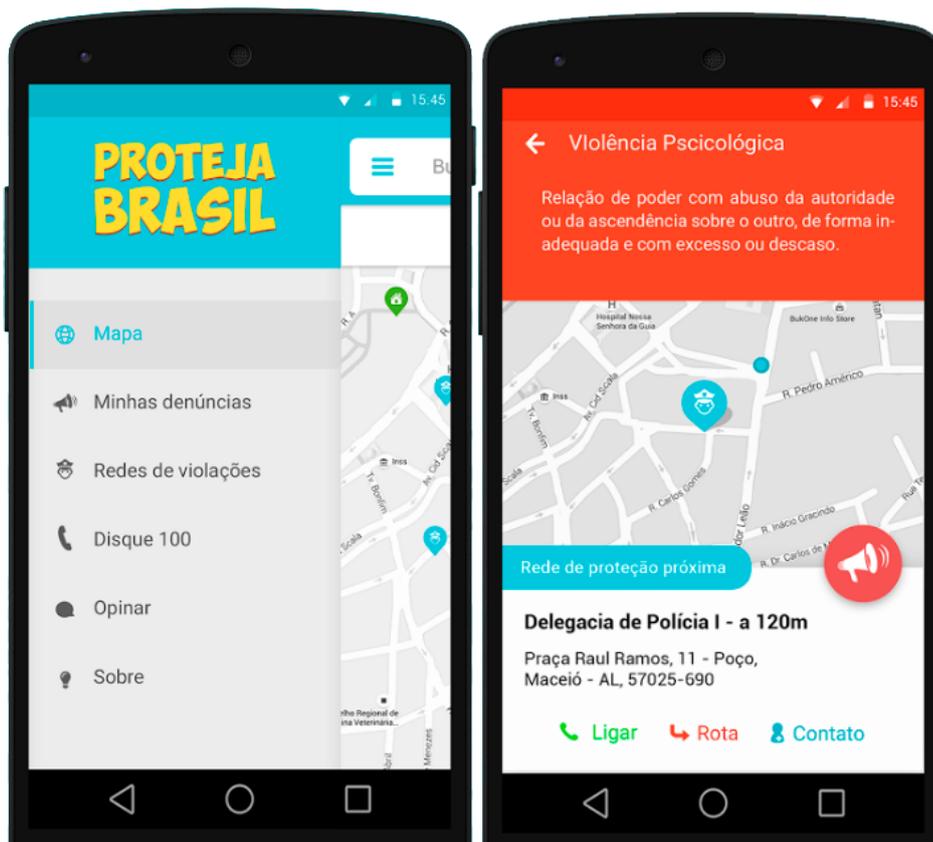
## Pontos Positivos

- Integração com CadÚnico;
- Registro de Atendimentos, Casos e Encaminhamentos;
- Simplicidade para consultar denúncias registradas.

### 3.3 Proteja Brasil

O Proteja Brasil é um aplicativo para *smartphones* e *tablets* criado para facilitar denúncias de violência contra crianças e adolescentes. Por meio dele, é possível obter os telefones e endereços de delegacias, conselhos tutelares e outras instituições do sistema de garantia de direitos que estão mais próximos do usuário (MOBILE, b).

Figura 3.3 – Telas do aplicativo Proteja Brasil. Fonte: (MOBILE, a).



## Pontos Positivos

- Aplicativo de fácil manuseio e bastante intuitivo;
- Visualização de delegacias, órgãos governamentais e outras unidades de apoio em forma de mapa;

- Diversas categorias de violação de direitos disponíveis;
- Possibilidade de criação de rotas da posição do usuário até a unidade que deseja ir;
- Disponível para todos os estados do Brasil.

### **Pontos Negativos**

- Alguns dados das delegacias e conselhos estão desatualizados;
- Não há opções de relatórios;
- Não há possibilidade de consultar status das denúncias realizadas;

### **3.4 Tradução e adaptação de *software* para o auxílio na identificação de maus-tratos em crianças e adolescentes**

O objetivo deste trabalho foi fazer a tradução e a adaptação do conteúdo do software “Módulo de Apoyo a la Gestión del Riesgo Social en la Infancia y la Adolescencia (MSGR)” para o contexto brasileiro. O *software* visa auxiliar profissionais de diversas áreas a identificar e a como proceder diante de casos de suspeitas de maus-tratos contra crianças e adolescentes. Foram realizados dois estudos empíricos (CALZA, 2014).

O primeiro estudo refere-se à tradução e à adaptação dos itens presentes no software. Enquanto o segundo, é determinado pela aplicação piloto do sistema, já traduzido e adaptado (CALZA, 2014). O primeiro estudo foi dividido em duas fases: (1) tradução dos itens, através do método *backtranslation*; e (2) validação por juízes e grupos de discussão, a fim de encontrar problemas de concordância e propor ações recomendadas para cada tipo de gravidade. Já, o segundo estudo, participaram 16 adultos com idades entre 22 e 55 anos. Estes foram instruídos a responder a três estudos de caso fictícios, apontando as dúvidas de compreensão dos itens, além de suas opiniões quanto ao manejo e a utilidade do software (CALZA, 2014).

Os resultados obtidos pela pesquisa indicaram dificuldades de compreensão de algumas palavras, especialmente entre os participantes que tinham somente o Ensino Médio completo. Também houve dúvidas quanto à ambiguidade de itens, além de tentativas de indução de respostas. No artigo é discutido aspectos relativos a facilitar o entendimento das questões e ao uso do software, além da importância deste para o auxílio na notificação das suspeitas.

A conclusão se dá pela relevância da divulgação dos conhecimentos científicos acerca da identificação de maus tratos para a população, assim como o desenvolvimento de novas ferramentas para a utilização na proteção dos direitos da infância e adolescência (CALZA, 2014).

## 4 METODOLOGIA

Este capítulo descreve como foram aplicados os métodos e ferramentas para o desenvolvimento do sistema “DATAGUARDIAN”, definidas previamente no Capítulo 2 (Fundamentação Teórica). É identificado a população de usuários afetada pela implantação do *software*, assim como, quais usuários participaram do treinamento do sistema e quais participaram do teste de usabilidade. Também são apresentados as ferramentas utilizadas, assim como suas versões e motivações para seu uso. Serão apresentados os diagramas Unified Modeling Language (UML) elaborados a partir do levantamento de requisitos que serviram como base para a produção desse *software*.

Este capítulo foi distribuído em quatro seções explicando as etapas percorridas para atingir o objetivo final deste projeto: Ambiente Computacional, Desenvolvimento do Sistema, Técnicas, Diagramas UML e Ferramentas Utilizadas.

Em Ambiente de Computacional são apresentados a maneira como o ambiente foi organizado e preparado para realizar a implementação do *software*, e sua publicação em modo produção. Em Desenvolvimento do Sistema, descreve-se os conceitos utilizados para a criação do sistema, e apresenta as ferramentas que auxiliaram à sua consumação. Na seção Técnicas, são expostos as utilizações práticas dos conceitos referentes ao teste e avaliação de usabilidade. Na seção referente à Diagramas UML, é apresentado o Diagrama de Caso de Uso, Diagrama de Classe e Diagrama de Atividades. A última seção detalha todas as ferramentas utilizadas para a elaboração do *software* e o que motivou na escolha de cada uma.

### 4.1 Desenvolvimento do Sistema

Para o desenvolvimento do sistema “DATAGUARDIAN” foram utilizadas técnicas de programação orientada a objetos juntamente com padrões de projeto para o ganho considerável de produtividade. Vale ressaltar o uso da metodologia *Scrum* para implementação deste trabalho.

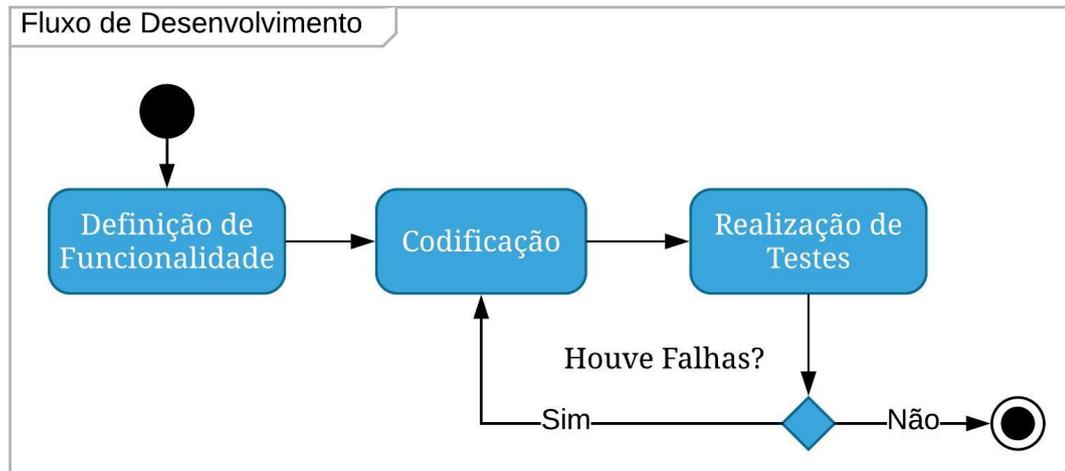
O fluxo de desenvolvimento das funcionalidades do sistema é representado na figura 4.1 onde o processo é iniciado com a definição da funcionalidade a ser implementada, seguindo de sua codificação. Os testes são realizados durante e depois de finalizado a tarefa, voltando sempre para a etapa de codificação ao reconhecer uma eventual falha.

É importante frisar o uso do Apache<sup>TM</sup> Subversion® (SVN)<sup>1</sup>, um Sistema de Controle de Versão, este utilizado por todo o ciclo de desenvolvimento do sistema com a finalidade de preservar o código fonte escrito, e manter os terminais de desenvolvimento continuamente sincronizados. Outras ferramentas para versionamento de códigos são bas-

---

<sup>1</sup><https://subversion.apache.org/>

Figura 4.1 – Fluxo de desenvolvimento do “DATAGUARDIAN”



tante conhecidas e utilizadas, como o Git Hub e o Mercurial, entretanto, a escolha do SVN se deu pela afinidade e facilidade de uso.

No processo de desenvolvimento de *software* a falta de definição de critérios para tomadas de decisões pode impactar profundamente na qualidade do *software* e na sua aceitação dentro do contexto para o qual foi proposto. Por este motivo foram adotadas boas práticas de programação integradas a metodologia *Scrum* utilizada para concepção do sistema conforme verificado na figura 2.1.

## 4.2 Ambiente Computacional

Esta seção demonstra os ambientes computacionais utilizados para efetuar a implementação e implantação do sistema proposto. Foi realizada uma pesquisa para determinar quais tecnologias e conjunto de ferramentas de desenvolvimento de *software* viabilizariam a criação de um ambiente de programação flexível para a conclusão do *software* proposto neste trabalho.

### Ambiente de Desenvolvimento

Com a finalidade de desenvolver o sistema “DATAGUARDIAN”, foi preparado um ambiente de desenvolvimento embasado na série de especificações Java EE<sup>2</sup> (Java Enterprise Edition) focando no desenvolvimento *web*. A linguagem de programação padrão utilizada foi a linguagem Java, trabalhando em conjunto com *frameworks* que simplificam e aumentam o nível de produtividade de implementação do sistema, como o JSF<sup>3</sup>,

<sup>2</sup>JavaEE - <http://www.oracle.com/technetwork/java/javaee/overview/index.html>

<sup>3</sup>JSF - <https://javaserverfaces.java.net/>

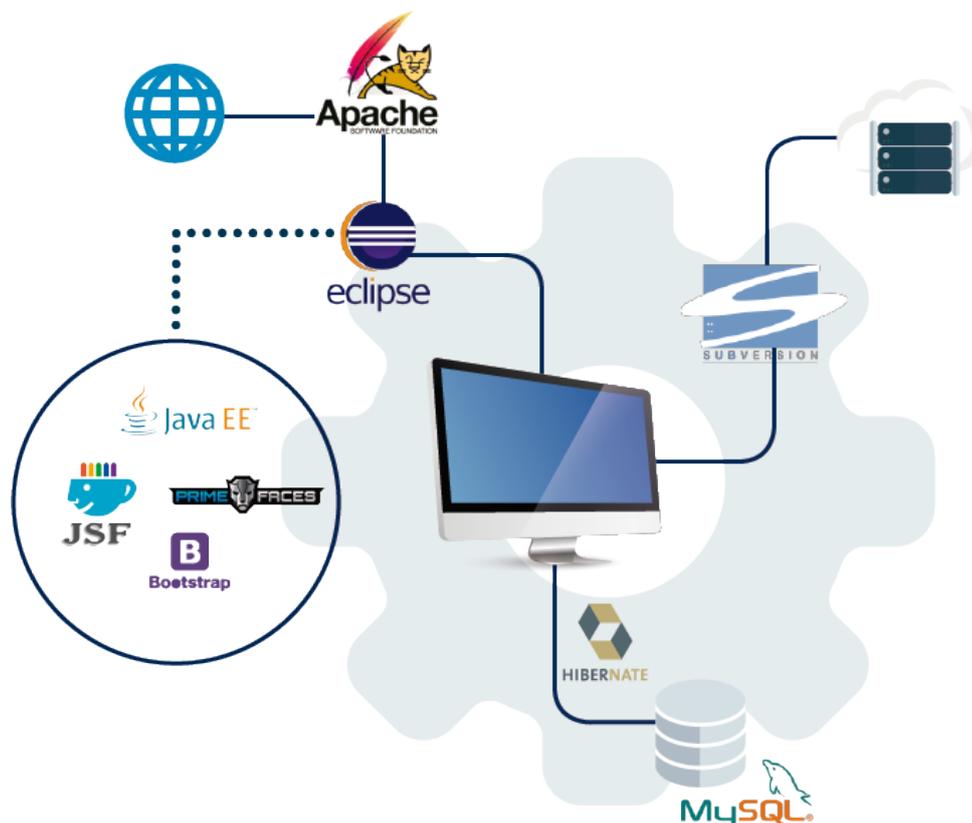
PrimeFaces<sup>4</sup> e JPA<sup>5</sup> com Hibernate<sup>6</sup> para o mapeamento objeto-relacional das entidades.

Foram utilizados dois ambientes de desenvolvimento integrado (IDE), Eclipse IDE<sup>7</sup>, como editor de código fonte, depurador e compilador padrão, e o iReport Designer, voltado para a criação dos *layouts* de relatório.

O Apache Tomcat<sup>8</sup> foi utilizado como servidor *web*, sendo bastante estável com todas as características que um container comercial de aplicações *web* possui. É o intermediador entre sistema cliente-servidor.

Como sistema de controle de versão, o SVN foi escolhido, pois facilita o trabalho em equipe, possibilitando que o sistema seja implementado de qualquer um dos pontos ligados ao desenvolvimento do projeto. Sua hospedagem localiza-se em um conjunto de repositórios remotos de acesso privado nos servidores da empresa Integrator<sup>9</sup>.

**Figura 4.2 – Ambiente de Desenvolvimento do sistema “DATAGUARDIAN”**



A figura 4.2 demonstra como funciona o ambiente de desenvolvimento usado na construção do sistema “DATAGUARDIAN”. No computador de desenvolvimento estão instalados as ferramentas para implementação, como a Eclipse IDE composta pelas de-

<sup>4</sup>PrimeFaces - <https://www.primefaces.org/>

<sup>5</sup>JPA - <http://www.oracle.com/technetwork/java/javaee/tech/persistence-jsp-140049.html>

<sup>6</sup>Hibernate: <http://hibernate.org/>

<sup>7</sup>Eclipse – <https://www.eclipse.org/ide/>

<sup>8</sup>Tomcat - [tomcat.apache.org/](http://tomcat.apache.org/)

<sup>9</sup><https://www.integrator.com.br/>

pendências necessárias para o desenvolvimento da aplicação, o banco de dados necessário para que seja possível simular o ambiente de produção, e o (JDK) *Java Development Kit*<sup>10</sup> para implementação e compilação de códigos em JAVA.

Na máquina do desenvolvedor está instalado o servidor *web container* Apache Tomcat, responsável por receber todas as requisições *Hypertext Transfer Protocol* (HTTP) do sistema *web*.

O SVN é utilizado para efetuar o controle de versão do código fonte. Todos os repositórios estão disponíveis em um *Cloud Virtual Private Servers* (VPS), mantido pela empresa Integrator Host do Brasil<sup>11</sup>, onde podem ser acessados pelos programadores envolvidos no projeto que tenham suas respectivas permissões de acesso.

## Ambiente de Produção

O sistema “DATAGUARDIAN” foi implantado em um servidor da própria Prefeitura do Município de Palmas, visando mais segurança, escalabilidade e robustez. A aplicação pode ser acessada a partir da *intranet*<sup>12</sup> da Prefeitura de Palmas, um portal corporativo destinado a promover integração entre os seus diversos serviços de TI, porque assim limita o acesso de pessoas não autorizadas. Todas as quatro unidades conselheiras de Palmas, estão ligadas à essa intranet.

O ambiente instalado no servidor disponibilizado pela Prefeitura de Palmas, é basicamente a mesma do ambiente de desenvolvimento, sendo diferente apenas no quesito *hardware*, com o objetivo de proporcionar um melhor desempenho visto o tráfego de dados e acessos simultâneos ao sistema.

Após o sistema ter sido implantado em ambiente produtivo, o mesmo foi adicionado às rotinas de *backup*, estas coordenadas pelo departamento de tecnologia da informação da prefeitura de Palmas. A realização dos *backup's* são feitas diariamente no período noturno, semanalmente aos fins de semana e mensalmente nos últimos dias de cada mês. Foi adicionado às instruções de *backup* o banco de dados do sistema e os diretórios de arquivos.

### 4.3 Técnicas

Esta seção apresenta como foi empregado o modelo de desenvolvimento ágil *Scrum*, os testes de software realizados, as maneiras utilizadas para a coleta de dados e os métodos utilizados para efetuar as avaliações de usabilidade (avaliação heurística baseados em especialistas e avaliação empírica baseados em participantes).

---

<sup>10</sup>JDK – <http://www.oracle.com/technetwork/java/javase/downloads/index.html>

<sup>11</sup><https://www.integrator.com.br/>

<sup>12</sup>A intranet é uma rede de computadores privada que assenta sobre a suíte de protocolos da Internet, porém, de uso exclusivo de um determinado local, como, por exemplo, a rede de uma empresa, que só pode ser acessada pelos seus utilizadores ou colaboradores internos.

## Desenvolvimento Ágil (*Scrum*)

Utilizar a metodologia *Scrum* foi fundamental para a organização e gerenciamento das atividades necessárias para a conclusão do desenvolvimento do sistema. O motivo de sua escolha se deu pelos benefícios que o *Scrum* oferece, como por exemplo, sua adaptabilidade onde ele se mostra aberto a mudanças durante o processo de desenvolvimento e o contato direto com o cliente ocasionando um *feed back* contínuo. Outro benefício, é uma menor preocupação com documentação abrangente, resultando na implementação mais ágil e eficiente. Em conjunto com as práticas ágil *Scrum* foi utilizado o modelo de organização de tarefas chamado *Kanban* (LEFFINGWELL, 2011), usado para visualizar e padronizar o fluxo de trabalho.

De forma prática, o quadro *Kanban* tem suas tarefas divididas em três colunas: À fazer, Em desenvolvimento e Finalizado. No entanto, dependendo do tamanho, da estrutura e dos objetivos da equipe, o mapeamento das colunas podem aumentar para atender ao processo exclusivo da equipe.

**Figura 4.3 – Fluxo de Trabalho Kanban.**



A figura 4.3 demonstra a utilização do quadro Kanban, onde cada tarefa é representada por um *post-it*. Para um melhor acompanhamento das demandas do *software*, foi utilizado a ferramenta *online* Trello<sup>13</sup>, uma ferramenta gratuita, flexível e visual de organizar tarefas.

Os *Product Backlog* foram formados a partir das atividades a serem implementadas. Um exemplo pode ser visualizado na tabela 4.1, onde as principais funcionalidades do "DATAGUARDIAN" são apresentadas.

A realização de todas as atividades que levaram ao desfecho deste projeto foram baseadas na metodologia de desenvolvimento *Scrum*, através do conceito de organizações dos *backlog* e das demandas a serem entregues ao final de cada *sprint*. É importante

<sup>13</sup>Trello: <https://trello.com/>

**Tabela 4.1 – Exemplo de Backlog do Produto**

<b>Backlog do Produto</b>	
<b>Id</b>	<b>Descrição</b>
1	Cadastro de crianças e adolescentes
2	Registro e acompanhamento de denúncias
3	Aplicação de medidas protetivas
4	Emitir termo de medidas aplicadas
5	Cadastro de casos antigos

ressaltar a presença dos conselheiros tutelares no trabalho em conjunto para definição das atividades a serem implementadas, como também, na avaliação das funcionalidades existentes.

### **Coleta de Dados**

Uma etapa muito importante para a concepção do *software* foi a coleta de dados, esta, realizada através de reuniões com os conselheiros tutelares e análise dos diversos relatórios e documentos que os conselheiros geram manualmente todos os dias.

Alguns documentos como o Termo de Aplicação de Medidas Protetivas e o relatório trimestral, auxiliaram para a construção de telas e modelagem do banco de dados. Para a construção da tela de Registro de Denúncia por exemplo, utilizou-se como base uma ficha (Imagem 4.4) onde os Conselheiros registram os atendimentos realizados nos Conselhos Tutelares. A partir dos dados presentes na ficha, percebeu-se a necessidade de criar um formulário contendo mais campos para inserção de mais informações, como pode-se observar na imagem 4.5.

Analisando os dados coletados e verificando as informações cuja necessidade possuía maior valor, foi possível definir os principais requisitos funcionais para o devido desenvolvimento do sistema.

Os requisitos levantados são demonstrados no Apêndice A, onde foram divididos em requisitos funcionais (RF) e não funcionais (RNF). Os requisitos funcionais se referem à requisições de funções que o sistema deverá atender ou realizar, ou seja, define o que o sistema deverá fazer (SOMMERVILLE, 2011). Quanto aos não funcionais, seu objetivo é atender a requisitos do sistema que de fato não se referem a funcionalidades do negócio, mas que fazem parte do escopo do sistema (SOMMERVILLE, 2011).

Figura 4.4 – Ficha de Atendimento Utilizada no Conselho Tutelar de Palmas

CONSELHO TUTELAR DOS DIREITOS DAS CRIANÇAS E DOS ADOLESCENTES  
LEI FEDERAL 8.069/90 E LEI MUNICIPAL 1.553/2008  
PALMAS – ESTADO DO TOCANTINS  
Região Norte

**FICHA DE ATENDIMENTO**

FICHA N° 2

1. CASO N° \_\_\_\_\_ 2. DATA \_\_\_\_/\_\_\_\_/\_\_\_\_ 3. HORARIO \_\_\_\_\_

4. DENUNCIANTE: \_\_\_\_\_

5. ENDEREÇO: \_\_\_\_\_

6. TIPO DE DENUNCIA: ( ) Pessoal ( ) Telefone ( ) Anônima

7. Nome: Criança (s) ou Adolescente \_\_\_\_\_ D/N  
\_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_

8. Endereço: \_\_\_\_\_

9. Telefone: \_\_\_\_\_

10. Filiação: \_\_\_\_\_

11. Responsável: \_\_\_\_\_

12. Narrativa da denúncia:  
\_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_

Figura 4.5 – Tela de Cadastro da Denúncia no “DATAGUARDIAN”

DATAGUARDIAN ONLINE ADMIN

Denúncia - Cadastro de Denúncia

Home / Cadastro de Denúncia

Atenção: A chave gerada para consultar esta DENÚNCIA depois de cadastrada é 201851/3

Unidade \* CONSELHO TUTELAR NORTE UF \* UF Cidade \* - CIDADE -

Identidade Provisória \* Origem da Denúncia \* Seleccione uma opção Agente-Violador \* Procurar

Local da Ocorrência \* Descrição da Ocorrência \*

Situação \* Pendente Data/Hora Provisível \* Data Registro 20/05/2018 19:45 Chave da Denúncia 20185-1/3

Observação

Selecione o Conselheiro que irá atender esta denúncia. N° CASO:

CONSELHEIRO 1 NORTE  CONSELHEIRO 3  CONSELHEIRO 4 NORTE  CONSELHEIRO 2 NORTE

Salvar Denúncia Limpar Campos Consultar Denúncia Imprimir Ficha

## Avaliação Baseada no Especialista

Esta avaliação se deu por duas etapas, a resolução de um *checklist* e avaliação heurística com base nos princípios de Nielsen. A avaliação heurística refere-se a vários métodos nos quais uma pessoa que possui treinamento em IHC e *design* de interação

examina o *design* proposto para avaliar como ele se qualifica diante de uma lista de princípios, diretrizes ou “heurísticas” (BENYON, 2011c).

Para realizar a avaliação, dois avaliadores (Apêndice I) se voluntariaram para analisar algumas características de *design* do sistema e relacionar com as heurísticas escolhidas na tentativa de encontrar pontos positivos e negativos, neste último caso, determinando um grau de severidade e oferecendo soluções para aperfeiçoar o *design* do sistema. Um modelo de questionário foi desenvolvido para facilitar a avaliação e demonstrar os resultados obtidos. Também foi elaborado um *checklist* (Anexo B) baseado no mesmo desenvolvido por Caroline Krone (KRONE, 2013), pois a combinação da avaliação heurística com listas de conferências ajudam a alcançar as vantagens das duas abordagens (PÁDUA, 2012).

Por se tratar de um teste, uma versão de teste do sistema foi utilizada, não provocando quaisquer alterações no sistema em ambiente de produção. Para cada avaliador foi criado um *login* e senha para autenticação. Antes de efetuarem a avaliação, os avaliadores foram submetidos a um breve treinamento para se habituarem com o *software*. As atividades a serem realizadas no sistema e quais características de *design* seriam avaliadas ficaram a cargo dos próprios avaliadores não sendo definidos previamente critério algum.

Cada avaliador realizou sua inspeção individualmente, e durante a avaliação, não houve contato entre os dois, garantindo assim avaliações independentes e sem influência. Durante a avaliação, o avaliador percorre a interface do sistema diversas vezes, averiguando os diversos elementos da interface e comparando-os com a lista de heurísticas.

Os relatórios elaborados pelos especialistas com os problemas encontrados, o grau de relevância de cada um e suas possíveis soluções serão demonstrados no capítulo 5, assim como, esclarecido as soluções que já foram implementadas e os resultados alcançados com a resolução do *checklist*.

### **Teste de Usabilidade Utilizando o SUS (*System Usability Scale*)**

No capítulo referente à Fundamentação Teórica, foi descrito os procedimentos para a realização do teste de usabilidade SUS, o qual se dá pela aplicação de um questionário. Também foi demonstrado a maneira de se realizar o cálculo de nível de facilidade de uso de sistemas, embasado em três fundamentais pontos: efetividade, eficiência e satisfação dos usuários.

O questionário elaborado para o teste de usabilidade é apresentado no Anexo A. Ele é composto por 10 afirmativas, onde cada uma possui cinco opções de resposta: Discordo Totalmente, Discordo, Neutro, Concordo e Concordo Totalmente. Dentre as opções, apenas uma pode ser escolhida como resposta de cada afirmativa. Ao final do questionário, o usuário tem a oportunidade de fazer algum comentário pessoal referente ao sistema, que também é levado em consideração na avaliação de usabilidade.

Um grupo de 25 pessoas responderam ao questionário, destes, 20 são Conselheiros

Tutelares e 5 são funcionários do departamento administrativo, onde dois trabalham na Unidade Conselheira da Região Sul II, outros dois na Unidade Conselheira da Região Centro e um na Unidade Conselheira da Região Norte. A faixa etária dos participantes variam entre 23 a 57 anos de ambos os sexos. Os perfis destes participantes é demonstrado de forma detalhada no Apêndice J. No capítulo 5 serão apresentados os resultados alcançados a partir da aplicação do questionário SUS, demonstrando de acordo com este teste, o nível de facilidade que obtiveram.

#### 4.4 Diagramas UML

Esta seção tem como objetivo expor os diagramas UML produzidos para a criação do sistema “DATAGUARDIAN”. Diagramas estes, desenvolvidos a partir do levantamento de requisitos que foram divididos entre funcionais e não funcionais determinando as principais funcionalidades do sistema. Com o propósito de se obter um *software* conforme o planejamento, os seguintes diagramas foram elaborados: Diagrama de Casos de Uso, Diagrama de Classes e Diagrama de Atividades.

##### Diagrama de Caso de Uso

O diagrama de Casos de Uso é produzido a partir dos requisitos levantados, seu objetivo é exemplificar a parte comportamental do sistema, uma visão dinâmica do que ocorre durante a execução de uma funcionalidade. O caso de uso não é somente composto pelo diagrama, mas muito mais importante é sua especificação, a descrição de seus cenários. Com base na especificação, transmite ao desenvolvedor o objetivo do diagrama. Segue os componentes deste diagrama:

- **Ator** - Aquele que executará a funcionalidade que está especificada no caso de uso.
- **Caso de Uso** - São os fluxos produzidos para que seja efetuado a funcionalidade fim determinada no diagrama.
- **Relacionamento** - Relações entre os casos de uso existentes no diagrama. Estes relacionamentos podem ou não existir. Os principais tipos de relacionamentos em um diagrama de caso de uso são: Inclusão (*include*), Extensão (*extend*) e Herança (*generalization*).

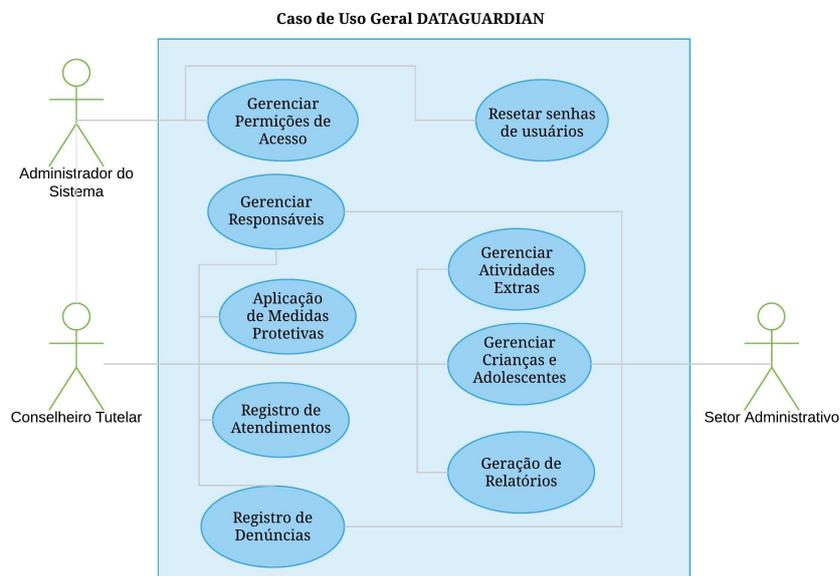
Os relacionamentos presentes em um diagrama de caso de uso podem ser explicados da seguinte forma:

- **Inclusão:** significa que ao executar um caso de uso “A”, o caso de uso “B” também será executado.

- **Extensão:** significa que ao executar um caso de uso “A”, o caso de uso “B” poderá (talvez não seja) executado.
- **Herança:** se um caso de uso “A” herda de um caso de uso “B”, ao executar “A”, será executado tudo que está especificado em “B” além do que está em “A”.

A fim de exemplificar melhor o funcionamento do fluxo de trabalho do sistema “DATAGUARDIAN”, foi criado um diagrama de caso de uso que demonstre de forma geral as principais atividades dos atores — Administrador, Conselheiro Tutelar e Setor Administrativo — conforme a figura 4.6.

**Figura 4.6 – Diagrama de Caso de Uso Geral do sistema DATAGUARDIAN.**



No diagrama, apresenta-se as principais atividades que podem ser exercidas por cada um dos atores mencionados, pode-se observar que o ator Conselheiro Tutelar é capaz de realizar praticamente todas as atividades, com exceção apenas de gerenciar os controles de acesso, o qual somente pode ser feito pelo ator Administrador. Já o ator Setor Administrativo, é capaz de realizar o cadastro de crianças e adolescentes, cadastro de responsáveis e o cadastro das denúncias que são recebidas no Conselho Tutelar.

No Apêndice B são demonstrados outros diagramas de caso de uso essenciais e suas especificações esclarecendo o caminho percorrido até que satisfaça o objetivo da funcionalidade.

## Diagrama de Classes

Em Engenharia de Software, o Diagrama de Classes é uma representação de estrutura e relacionamentos entre as classes que servem como modelos para os objetos. Este

diagrama é muito importante para o entendimento de como será organizado as classes no sistema dando um maior entendimento para os desenvolvedores.

A classe neste diagrama é representado por um retângulo que pode possuir três divisões compostas pelos seguintes elementos:

- Nome da Classe
- Atributos: Determina as características de uma classe.
- Operações: Representa atividades que objetos de uma determinada classe pode executar.
- Relacionamentos: Descreve um vínculo que ocorre, normalmente, entre os objetos de uma ou mais classes.

As imagens 4.7, 4.9 e 4.10 apresentam os diagramas de classes que serviram como base para o desenvolvimento do sistema “DATAGUARDIAN”. Para tornar mais simples o entendimento, o diagrama foi dividido nestas três imagens.

O diagrama ilustrado em 4.7 explica como é organizado as classes que formam o controle de acesso juntamente com seus relacionamentos. A classe Módulo representa todos os módulos existentes no sistema como por ex: Módulo Cadastro, Módulo Relatório e Modulo de Denúncia. A classe Perfil representa os perfis de usuário, sendo que para cada Módulo, existe pelo menos um Perfil associado. A classe Funcionalidade representa todas as telas cadastradas no sistema, onde sua visualização será permitida apenas se estiver cadastrada seguindo os parâmetros da classe PermissaoAcesso. A classe UsuarioModulo-Perfil diz quais perfis estão vinculados ao usuário cadastrado.

A imagem 4.9 representa as principais entidades persistentes no banco de dados, ou seja, qualquer mudança nestas classes como inclusão de um novo atributo, exigirão uma atualização obrigatória do banco de dados.

O padrão de arquitetura utilizado para a implementação do sistema é o *Model-View-Controller* (MVC), onde sua principal característica é separar as camadas de uma aplicação em diferentes níveis.

A camada *View*, é relacionada ao visual da aplicação. A camada *Model* consiste nas regras de negócio, enquanto a camada *Controller*, funciona como intermediador entre as duas outras camadas. Este padrão foi adotado, pois funciona em conjunto com o *framework* utilizado, e por ser muito difundido no cenário de aplicações *web*.

O diagrama representado pela figura 4.10 demonstra a maior parte lógica do *software*. Estão presentes as classes de *Controller*, *DAO*, *Filtro de Requisições*, entre outras. Dentre as classes exibidas nesta imagem, vale ressaltar a importância de algumas, como por exemplo a interface *GerenteModulo*, que assina os métodos de adicionar, editar e

Figura 4.7 – Modelagem das classes de controle de acesso e seus relacionamentos

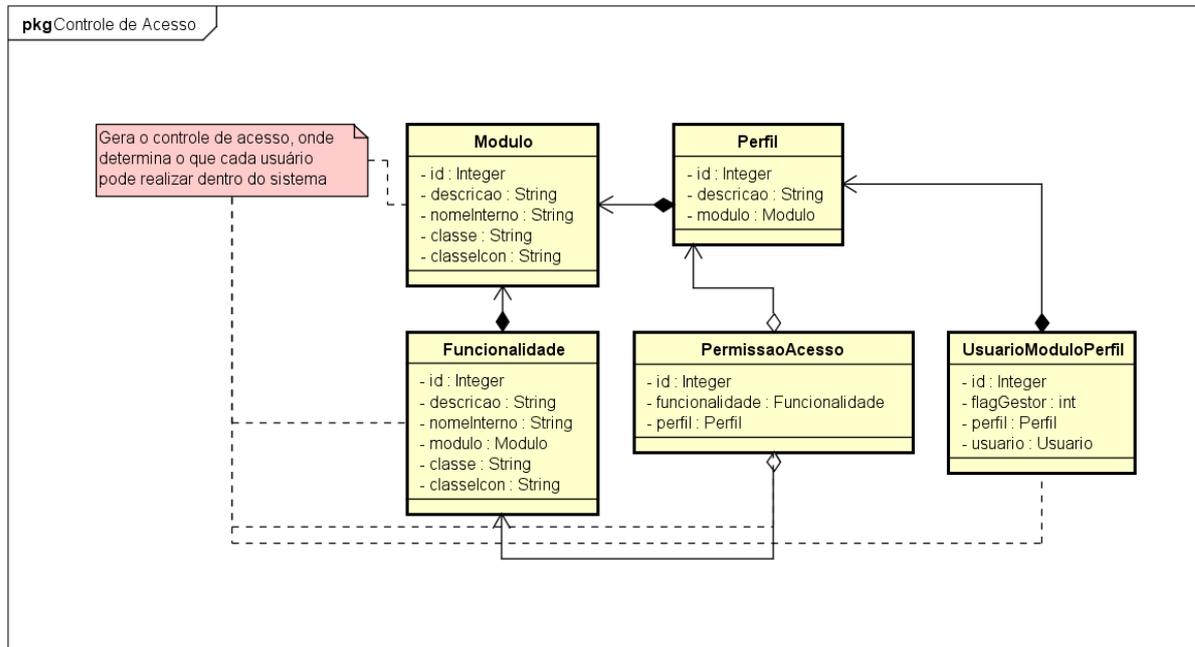
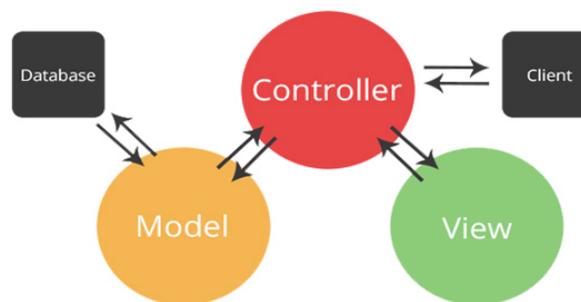


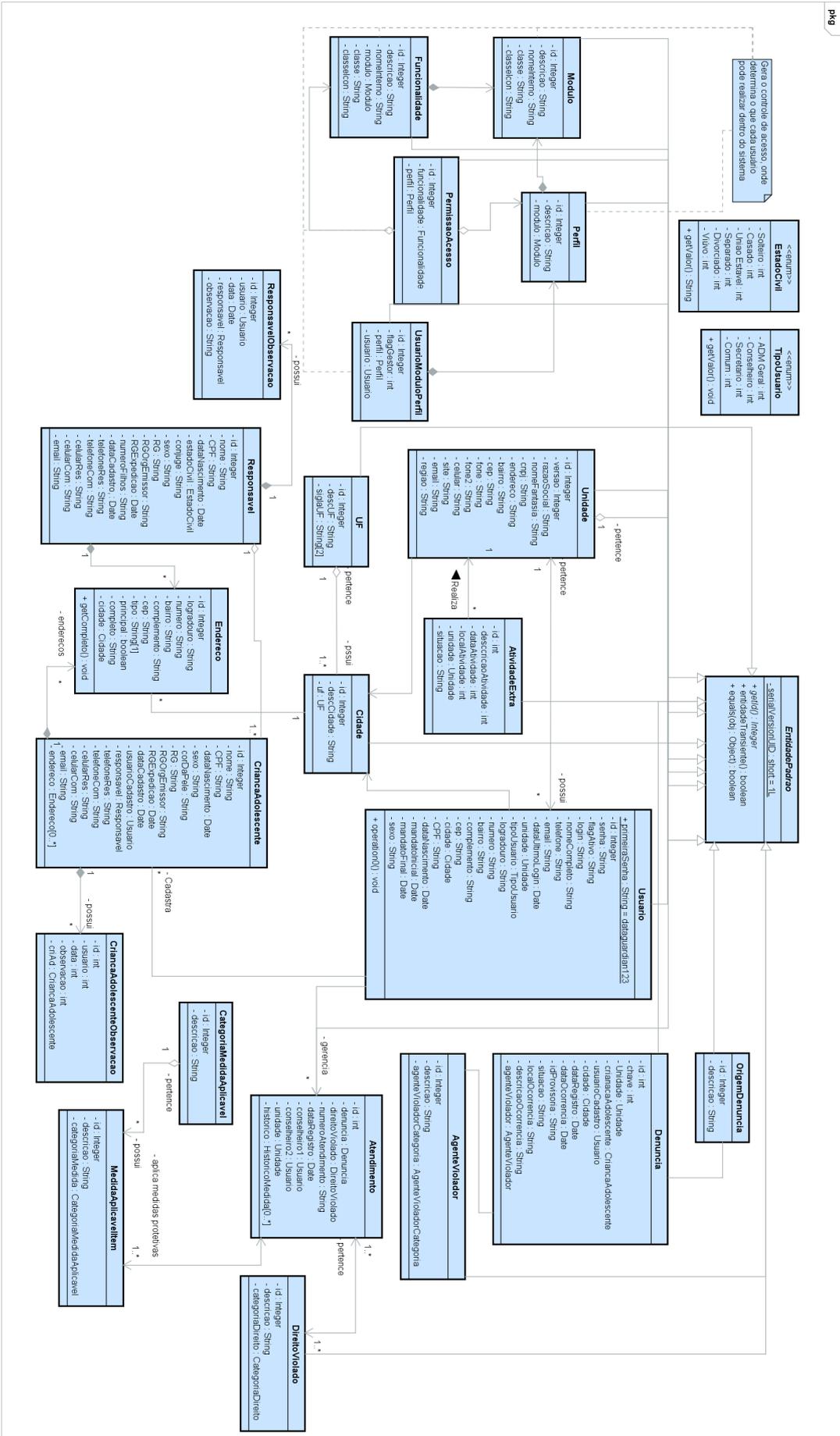
Figura 4.8 – Padrão de Arquitetura de Software MVC



excluir um objeto na base de dados. A classe *CadastroPadrao* que é responsável pela implementação destes métodos genéricos e a classe *FabricaTransacaoHibernate* que controla as *sessões*, abrindo, fechando e criando novas quando necessário.

A interface *GerenteArquivos* é responsável pela implementação dos métodos pertinentes ao tratamento de arquivos dentro do *software*, assim como o *GerenteCadastro* é responsável pela implementação dos métodos relacionados aos cadastros. Este padrão se mostra muito eficiente quando for necessário o desenvolvimento de novos módulos. Esta é uma das muitas vantagens do *Design Pattern Factory*, utilizado neste projeto para facilitar a inclusão de novas funcionalidades.

Figura 4.9 – Diagrama de Classe do “DATAGUARDIAN”



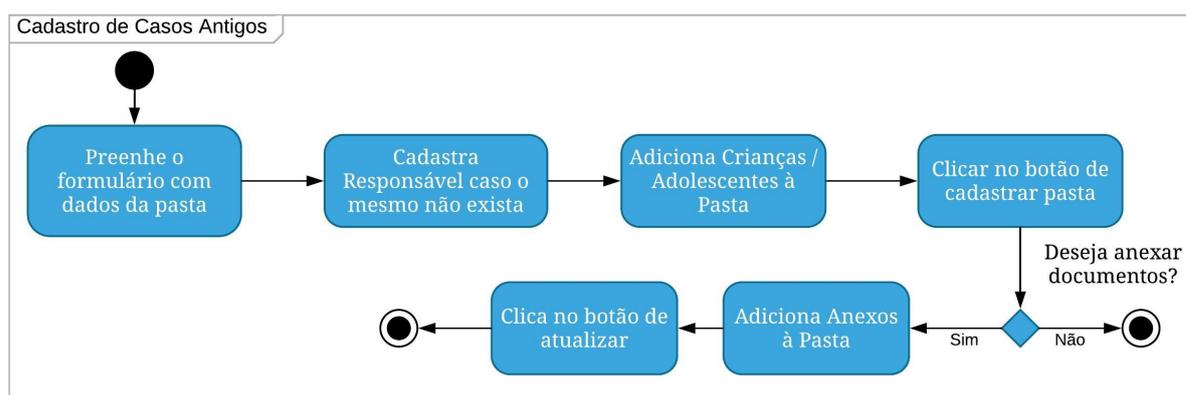


## Diagrama de Atividades

Foram desenvolvidos diagramas de atividades das principais funcionalidades do sistema com o objetivo de entender melhor como as mesmas se comportam. Na seção 5.1 são exibidas algumas capturas de tela do *software* demonstrando como são aplicados esses diagramas.

Uma funcionalidade fundamental do sistema é a opção de cadastro de pastas antigas, os casos que já foram registrados no Conselho Tutelar que são armazenados em armários dentro das unidades. Como foi levantado pelos próprios Conselheiros a necessidade de poder transferir as pastas físicas para dentro da base de dados do sistema, esta funcionalidade foi implementada. O diagrama de atividade pode ser visualizado na imagem 4.11. Os demais diagramas de atividade elaborados podem ser visualizados no Apêndice C.

**Figura 4.11 – Diagrama de Atividade Cadastro Casos Antigos**



A imagem 29 explica o processo para o cadastro de um Responsável, que consiste na validação dos dados inseridos a partir do formulário de cadastro. Sendo este efetivado com sucesso, o Responsável poderá ser usado para cadastrar Crianças e Adolescentes, cuja funcionalidade é representada pela imagem 30.

Na sequência, a imagem 31 representa como é realizado o registro de denúncias. As denúncias podem ser obtidas via telefone, disque 100, presencial, ou outros meios de comunicação que possibilitem o diálogo entre denunciante e Conselho Tutelar. Para o registro bem sucedido, todos os campos são de preenchimento obrigatório, com exceção do campo de observação. Ao finalizar, uma chave é gerada para que esta seja usada para consultas futuras. Por fim, uma ficha da denúncia cadastrada no formato PDF é aberta em uma janela externa, possibilitando o *download*, impressão ou apenas sua visualização.

A imagem 31 ilustra o processo para cadastrar um Caso ou Reincidência, demonstrando a obrigatoriedade das demais entidades vinculadas para que o registro seja realizado

sem erros. Mostra também como acontece a aplicação de Medidas Protetivas aos casos, tarefa de muito importância dentro dos conselhos tutelares.

#### 4.5 Ferramentas Utilizadas

Nesta seção é descrito as ferramentas utilizadas para o desenvolvimento do sistema “DATAGUARDIAN”, assim como a funcionalidade, a aplicação e a finalidade de cada uma. Segue a lista das principais ferramentas que auxiliaram para a implementação do *software*:

- A Linguagem de Programação Java<sup>14</sup> é atualmente, uma das linguagens mais utilizadas no mundo por programadores, e possui vasta documentação online, possibilitando grande facilidade para aprender e desenvolver soluções. Esta foi a linguagem utilizada para o desenvolvimento do sistema proposto neste trabalho, onde viabilizou considerável produtividade em sua implementação.
- JavaServer Faces (JSF)<sup>15</sup> é uma especificação Java para a construção de interfaces de usuário baseadas em componentes para aplicações *web*. A implementação JSF utilizada neste projeto é o Mojarra 2.2.
- O PrimeFaces<sup>16</sup> é um *framework* de código aberto para JSF que fornece mais de 100 componentes para serem usados em projetos JSF. São componentes sensíveis ao toque, estruturas de formulários para envio de informações, validações do lado do cliente, mecanismos de tema, dentre outros. Foi conveniente sua escolha pois diminui a preocupação do desenvolvedor com código HTML e JavaScript, além da grande diversidade de componentes oferecidos. A versão utilizada neste projeto foi a 6.0.
- Bootstrap<sup>17</sup> é um dos *framework's* HTML, CSS, e JS para o desenvolvimento de projetos responsivos. Ele possibilita o desenvolvimento *front-end web* mais rápido e mais fácil, adequando os projetos em que é utilizado para todos os navegadores e dispositivos. Por possuir documentação detalhada e ser descomplicado, foi utilizado no desenvolvimento dos *layouts* do “DATAGUARDIAN”. A versão aplicada foi a 3.3.
- O Java Persistence API (JPA)<sup>18</sup> é um *framework*, baseado em POJOS (Plain Old Java Objects) para persistir objetos Java. POJO são classes simples, que represen-

---

<sup>14</sup>JavaEE - <http://www.oracle.com/technetwork/java/javaee/overview/index.html>

<sup>15</sup>JSF - <https://javaserverfaces.java.net/>

<sup>16</sup>PrimeFaces - <http://www.primefaces.org/>

<sup>17</sup>Bootstrap - <http://getbootstrap.com.br/>

<sup>18</sup>JPA - <http://www.oracle.com/technetwork/java/javaee/tech/persistence-jsp-140049.html>

tam entidades a serem persistidas no banco de dados relacional em ambientes de desenvolvimento orientado a objeto. A implementação Hibernate<sup>19</sup> foi utilizada.

- Banco de Dados MySQL<sup>20</sup> é o banco de dados de código aberto mais conhecido no mundo. Com comprovado desempenho, confiabilidade e facilidade de uso, o MySQL tornou-se a principal opção de banco de dados para aplicativos baseados na Web (ORACLE, ). A ampla documentação e o alto número de desenvolvedores que trabalham com MySQL, incentivaram o seu uso neste trabalho. Versão usada 5.7.
- Apache Tomcat<sup>21</sup> é um *web container* que pode ser usado como servidor *web*. Desenvolvido pela Apache Software Foundation, é distribuído como software livre. Bastante utilizado por programadores java, e possuindo fácil configuração. A versão utilizada neste projeto é a 8.
- A IDE Eclipse<sup>22</sup> foi escolhida para o desenvolvimento deste sistema por se tratar de uma ferramenta leve, sofisticada e ideal para desenvolvimento de *softwares web*, além de ser totalmente gratuita.
- Para criação dos layouts dos relatórios, foi utilizado o IReport Designer<sup>23</sup>, ferramenta de código aberto fornecida pela Jaspersoft Community. Esta ferramenta proporciona o desenho de relatórios desde o mais simples até o mais complexo com diversas opções de formato e visualização. A versão 5.6 foi utilizado para o desenho dos relatórios do sistema tratado neste projeto.
- Foram utilizados dois sistemas operacionais no período de desenvolvimento do sistema: Windows<sup>24</sup> e Ubuntu OS<sup>25</sup>. São dois dos sistemas operacionais muito utilizados e funcionaram perfeitamente com as ferramentas aplicadas a este projeto.

## Considerações Finais Sobre o Capítulo

Nesta seção foram apresentados as técnicas utilizadas e como foram aplicadas no desenvolvimento do projeto. Considerou-se mostrar os ambientes de desenvolvimento, a maneira como foi empregada a metodologia *Scrum*, e os recursos para a coleta de dados. Foi demonstrado também os diagramas que levaram à concepção do *software* e as ferramentas utilizadas. Através dos conceitos aplicados, obteve-se uma base sólida para organização dos processos e serviços utilizados para implantação e manutenção do “DATAGUARDIAN”.

---

<sup>19</sup><http://hibernate.org/>

<sup>20</sup>MySQL - <https://www.mysql.com/>

<sup>21</sup>Tomcat - <http://tomcat.apache.org/>

<sup>22</sup><https://eclipse.org/>

<sup>23</sup>iReport - <http://community.jaspersoft.com/project/ireport-designer>

<sup>24</sup>Windows - <https://www.microsoft.com/pt-br/windows/>

<sup>25</sup>Ubuntu - <https://www.ubuntu.com/>

## 5 RESULTADOS E DISCUSSÃO

Este capítulo é dedicado aos resultados obtidos durante o desenvolvimento e após a implantação da ferramenta “DATAGUARDIAN”. São apresentadas telas de determinadas funcionalidades documentando o desfecho do *software* desenvolvido, como também, os resultados dos testes efetivados em cima do sistema.

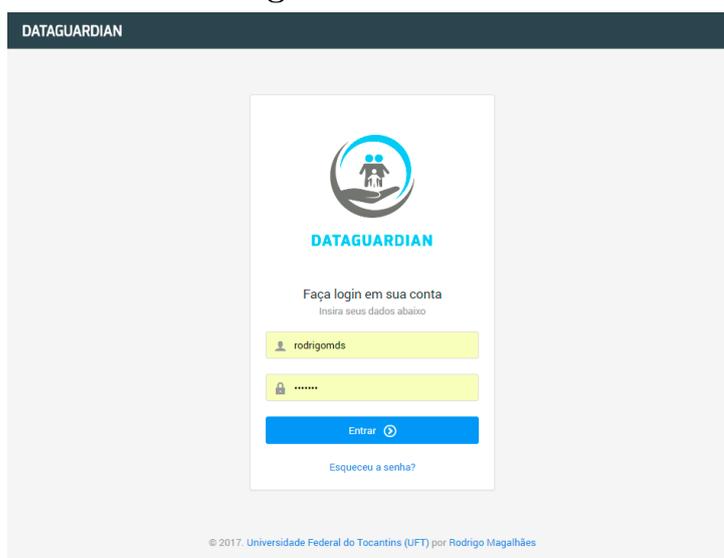
Foi considerado expor neste trabalho o teste de performance que convém avaliar e validar os limites operacionais de processamento do *software*, o teste de segurança, realizado a fim de encontrar vulnerabilidades no sistema, a aplicação do questionário SUS e avaliação heurística de usabilidade do sistema com o objetivo de mensurar a facilidade de sua utilização, a satisfação do usuário e o grau de produtividade na interação humano-computador.

### 5.1 O *Software*

A demonstração do *software* desenvolvido se dará através da captura de algumas de suas telas, sendo estas as principais a serem utilizadas pelos seus usuários. Serão apresentadas imagens de como o sistema se comporta nos diversos tamanhos de tela com o objetivo de mostrar a técnica de *responsividade* aplicada.

A figura 5.1 representa o primeiro contato do usuário com o sistema, onde é realizado o *login*.

Figura 5.1 – Tela de login do sistema “DATAGUARDIAN”



Na figura 5.2 observa-se o formulário para cadastramento dos usuários como administradores e conselheiros. Uma tabela com os usuários já cadastrados se encontra logo

abaixo, facilitando assim a seleção de algum membro do conselho para que seja efetuado quaisquer alterações.

**Figura 5.2 – Tela de Cadastro de Usuários “DATAGUARDIAN”**

Login	Nome	Telefone	Email	Situação
COORDENADOR	COORDENADOR			ATIVO
PAULO	PAULO			ATIVO

**Figura 5.3 – Tela de Cadastro de Crianças e Adolescentes**

O cadastro de Criança e Adolescente pode ser visto na figura 5.3. Este é o maior formulário existente no sistema devido a grande quantidade de informações que uma

Criança/Adolescente pode possuir. O Conselho Tutelar trabalha em prol da defesa dos direitos das crianças e adolescentes, e quando um ou mais desses direitos são violados, é crucial a obtenção do maior número de informações possíveis. Informações como dados pessoais, situação de saúde, situação familiar e escolar podem ser preenchidos a partir deste formulário.

A *responsividade* em ação pode ser vista na figura 5.4, onde são apresentadas as telas de Registro de Atendimento e Aplicação de Medidas Protetivas (da esquerda para direita). No sistema “DATAGUARDIAN”, o registro de atendimentos ocorre partindo de uma denúncia, e as medidas protetivas serão aplicadas após o estudo desse atendimento. Pode-se observar o campo onde vincula-se a denúncia previamente cadastrada ao atendimento que será realizado, da mesma forma, o campo onde vincula um atendimento para a aplicação das medidas protetivas.

**Figura 5.4 – Tela de Registro de Atendimento e Aplicação de Medidas Protetivas**

As medidas protetivas aplicadas podem ser alteradas ou adicionadas dependendo de cada situação. É atribuição dos conselheiros determinar quais medidas serão pertinentes a cada caso, e quais procedimentos tomar em condição de não atendimento destas medidas.

Por fim, temos o Histórico de Atividades Extras, funcionalidade representada pela imagem 5.5, que apesar de bem básica, é muito importante para o controle dos trabalhos

externos realizados pelos conselheiros, como por exemplo: palestras em escolas, conferências, cursos e *workshop's*.

**Figura 5.5 – Tela de Histórico de Atividades Extras “DATAGUARDIAN”**

The screenshot shows the 'DataGuardian - Atividades' interface. At the top, there is a navigation bar with a home icon and the text 'Home / Histórico de Atividades'. Below this is a search filter section with the following fields and buttons:

- Conselho Tutelar:** A dropdown menu with 'CONSELHO TUTELAR' selected.
- Categoria da Atividade:** A dropdown menu with 'Conferências' selected.
- Data Inicial:** An empty text input field.
- Data Final:** An empty text input field.
- Pesquisar:** A blue button with a magnifying glass icon.
- Pesquisar Atividades:** A blue button.
- Limpar Filtro:** A blue button.

Below the search filter is a table titled 'Atividades Realizadas' with the following columns: 'Cons. TUTELAR', 'DATA', 'CATEGORIA', and 'ATIVIDADE'. The table contains two rows of data:

Cons. TUTELAR	DATA	CATEGORIA	ATIVIDADE
CONSELHO TUTELAR	02/05/2017	Palestras Escolas	LOREM IPSUM DOLOR SIT AMET, CONSECTETUR ADIPISCING ELIT. DONEC EUISMOD MAURIS ERAT, EU COMMODO LEO PHARETRA NEC. AENEAN SEMPER.
CONSELHO TUTELAR	01/05/2017	Conferências	DUIS ELEIFEND, DOLOR QUIS AUCTOR FEUGIAT, NIBH IPSUM IACULIS TELLUS, QUIS PORTA ELIT ODIO NON MAURIS. DONEC A LUCTUS ODIO.

At the bottom of the table, there is a pagination control showing '< < 1 > >'.

## 5.2 Teste de Performance

Efetuar um teste de performance é fundamental para determinar como o sistema se comporta perante situações de grandes quantidades de requisições. Este foi o teste que se mostrou mais relevante em seu efeito. O mesmo foi alcançado usando a ferramenta Apache JMeter<sup>1</sup>.

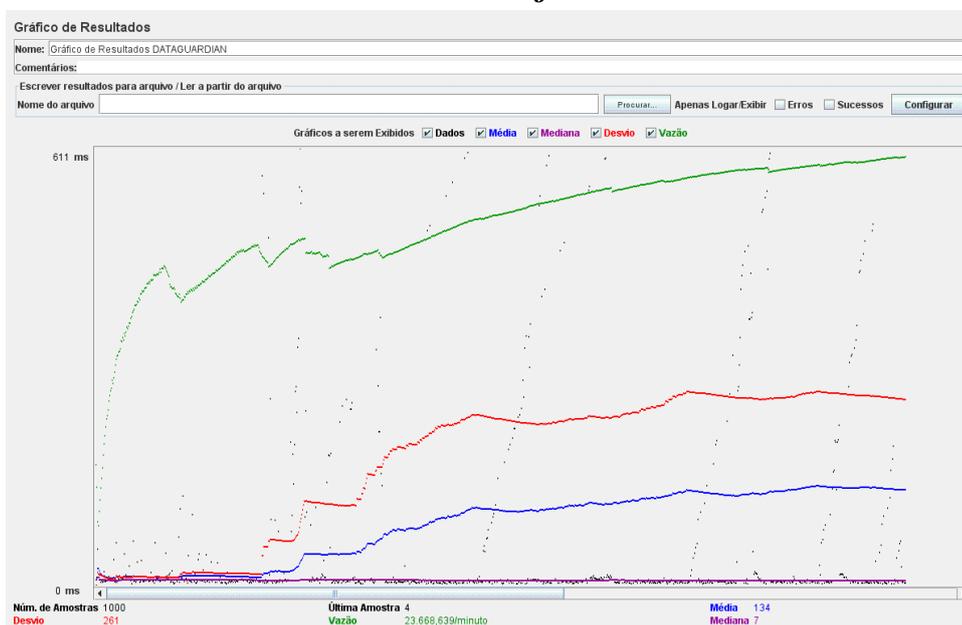
Para a realização deste teste, o JMeter foi configurado para simular o acesso simultâneo de 1000 (mil) usuários efetuando requisições HTTP ao sistema.

Na figura 5.6 pode-se conferir o gráfico produzido a partir de uma simulação realizada. Segundo o gráfico, percebe-se que para a execução desta tarefa, foram gastos o tempo de 611 milissegundos, considerado altamente rápido julgando o número de acessos. Vale ressaltar que o “DATAGUARDIN” é restrito a poucos usuários, vista a realidade de sua aplicação.

Ainda de acordo com o gráfico, pode-se notar a vasão de 23.668,639 por minuto, o que significa, a quantidade de operações por minuto que o sistema pode realizar dado este período de tempo, além do desvio padrão 261 em relação ao tempo decorrido das requisições.

<sup>1</sup>JMeter - <http://jmeter.apache.org/>

**Figura 5.6 – Gráfico gerado a partir do teste de performance usando a ferramenta jMeter**



### 5.3 Teste de Segurança

O teste de segurança teve como foco os pontos onde a segurança tem uma maior probabilidade de ser comprometida. Optou-se por realizar em conjunto os testes de caixa preta e caixa branca para um resultado mais preciso. São testes mais simples de utilizar, porém muito eficientes quanto à verificação da existência de vulnerabilidades, o que motivou sua utilização.

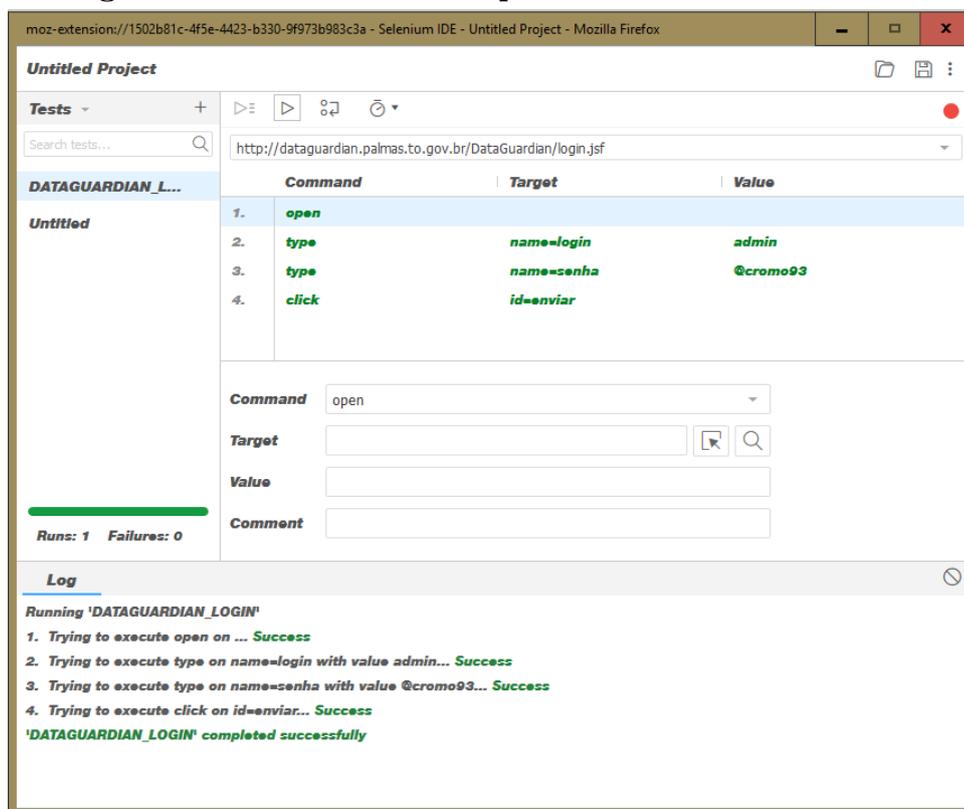
O teste de caixa preta possui uma característica que o identifica: o analista desconhece o código fonte e não tem acesso a estrutura interna do *software*. Este teste consiste em verificar a existência de falhas através da simulação da inserção de entradas inválidas no sistema. Em contrapartida, no teste de caixa branca, o analista tem acesso ao código fonte e conhece a estrutura interna do sistema, possibilitando que sejam escolhidas partes específicas de um componente para serem avaliadas, assim sendo, os dois testes complementares.

Para a realização do teste de caixa preta, utilizou-se a ferramenta de automação de *browsers* Selenium<sup>2</sup>. Esta ferramenta possibilita criar simulações de casos de uso repetidas vezes em uma página *web*. Pode-se observar na figura 5.7 como foi aplicado o teste de segurança no formulário de autenticação do sistema “DATAGUARDIAN”. É simulado a entrada dos dados de login e senha para que sejam validados as informações analisando o comportamento do sistema ao receber entradas com dados inválidos.

Foram simuladas entradas de usuários válidos e inválidos, com e sem permissões de acesso. Foi testado também situações onde o login estava correto e a senha incorreta,

<sup>2</sup>Selenium: <http://www.seleniumhq.org/>

Figura 5.7 – Teste de caixa preta utilizando o Selenium



como o contrário, com o objetivo de assegurar a confiabilidade do sistema. Concluiu-se durante este teste que não houve situação de falha.

O teste de caixa branca também ocorreu na verificação de falhas no método de autenticação do sistema ao inserir valores de *login* e senha válidos e valores inválidos.

A aplicação não permite usuários com o mesmo login, e todas as senhas são resumidas através do algoritmo MD5 (Message-Digest algorithm 5), um algoritmo de *hash* de 128 bits unidirecional desenvolvido pela RSA Data Security, para verificação de integridade e *logins*. As senhas dos usuários não são salvas da mesma forma que digitadas pelo usuário, mas sim o *Hash*, que é uma chave gerada que representa a senha (PISA, 2012). Por se tratar de um algoritmo unidirecional, um *hash* MD5 não pode ser transformado de volta na sua *password* que lhe deu origem. Portanto, o procedimento de verificação é feito pela comparação de duas *hash* (uma da base de dados, e a outra da tentativa de login) (DEV MEDIA, 2012).

Em todas as tentativas de login com dados inválidos, o sistema respondeu de forma esperada, não realizando a autenticação e disparando a mensagem com o respectivo erro ao usuário. Constatou-se que durante a execução do teste não houve nenhuma falha que tenha exposto alguma vulnerabilidade do sistema de autenticação. A combinação do teste de caixa preta com o teste de caixa branca se mostraram bastante úteis na análise do comportamento do sistema durante a entrada de dados inválidos.

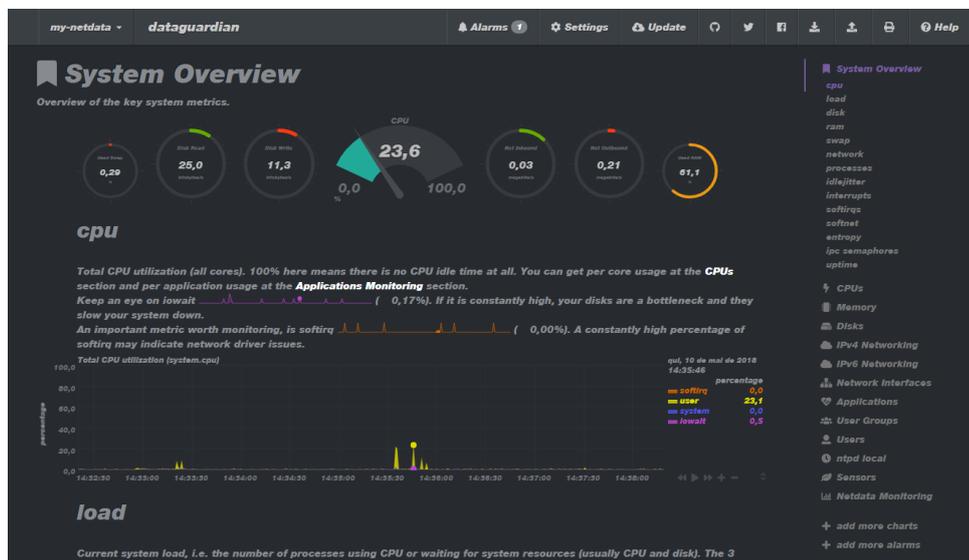
Também foi realizado simulações de *SQL Injection* no formulário de *login* do sistema, o qual obteve ótimos resultados, pelo o motivo de não serem utilizados código puro de MySQL no sistema, e sim, a especificação *Hibernate* para modelagem objeto-relacional da base de dados, e código *Criteria* para realização de consultas.

#### 5.4 Monitoramento do Sistema

Um *software* deve ser implantado em um ambiente que atenda seus requisitos mínimos para que assim funcione de forma adequada. O monitoramento de TI é essencial para que haja controle sobre a infraestrutura e dar sustentação aos sistemas. Através deste gerenciamento, é possível medir disponibilidade, performance e outros indicadores indispensáveis. Desse modo, optou-se pelo uso de sistema de monitoramento do *software* “DATAGUARDIAN”, para supervisionar a relação entre uso do *hardware* e de rede, após sua implantação em um ambiente em produção.

Para a realização do monitoramento em tempo real dos recursos do servidor, utilizou-se a ferramenta de código aberto Netdata<sup>3</sup>. O Netdata é uma ferramenta de monitoramento desenvolvida para sistemas Linux, muitas informações são disponibilizadas como uso da CPU, total de memória RAM, largura de banda e processos em execução dentre muitas outras.

Figura 5.8 – DATAGUARDIAN sendo monitorado pelo Netdata



A figura 5.8 demonstra um exemplo da utilização do Netdata monitorando o sistema “DATAGUARDIAN”, onde são apresentadas informações muito relevantes como uso de CPU, memória RAM e informações sobre uso do disco.

A partir dos relatórios obtidos pelo Netdata, é possível efetuar uma análise mais profunda em busca da necessidade de adicionar mais memória, mais espaço ou largura

<sup>3</sup>Netdata: <https://github.com/firehol/netdata>

de banda, como também verificar a imposição de novos servidores. Em suma, através do monitoramento, pode-se realizar um diagnóstico das principais características da aplicação com finalidade de efetuar configurações adicionais que possam aumentar a performance do sistema em produção.

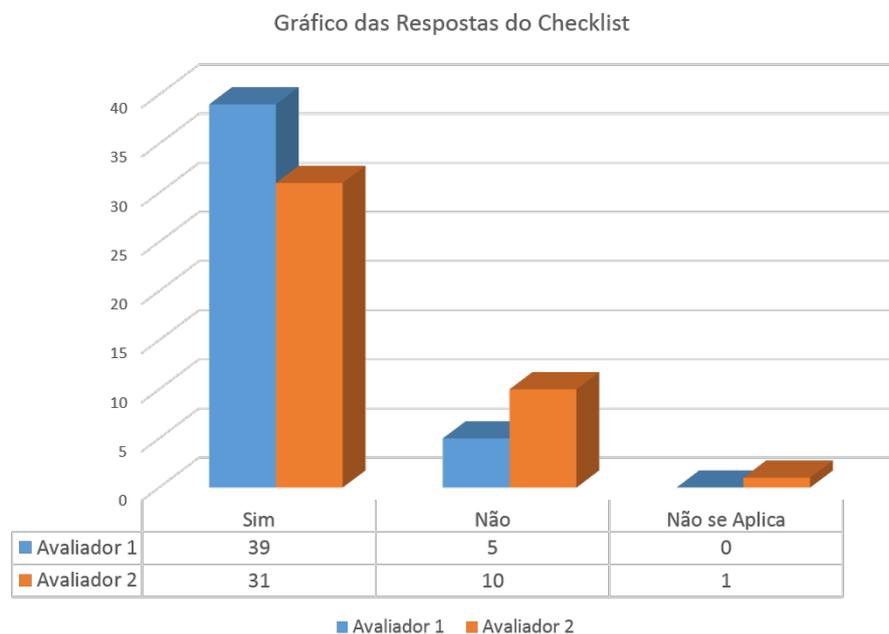
## 5.5 Resultados da Avaliação Baseada no Especialista

A análise heurística foi realizada considerando a resolução do *checklist* elaborado (Anexo C) e os 10 princípios de Nielsen referentes à usabilidade. Esta avaliação teve como objetivo levantar resultados qualitativos de usabilidade do sistema “DATAGUARDIAN”.

### Resultados do *Checklist*

Os resultados do *checklist* dos dois avaliadores são apresentados no gráfico 5.9, estes resultados permitiram identificar em que nível de grau de usabilidade o sistema avaliado se encontra e em quais aspectos atendem ou não questões de usabilidade.

**Figura 5.9 – Gráfico gerado a partir das respostas do checklist de usabilidade.**



Segundo o gráfico 5.9, para ambos avaliadores, o sistema alcançou ótimos resultados, recebendo “Sim” para a grande maioria das questões. Dentre as questões que obtiveram resposta negativa, a maioria está ligada à documentação e ajuda, onde não foi encontrado no sistema tais opções. A questão 28 também foi considerada, pois de acordo com o avaliador, algumas funcionalidades requerem vários passos para serem concluídas. Outra questão que obteve resposta negativa foi a questão 6, que embora algumas telas

apresentem informações em formato de listas e sejam disponibilizadas de forma organizada e de fácil entendimento, em alguns casos, não seguiram critério algum de ordenação.

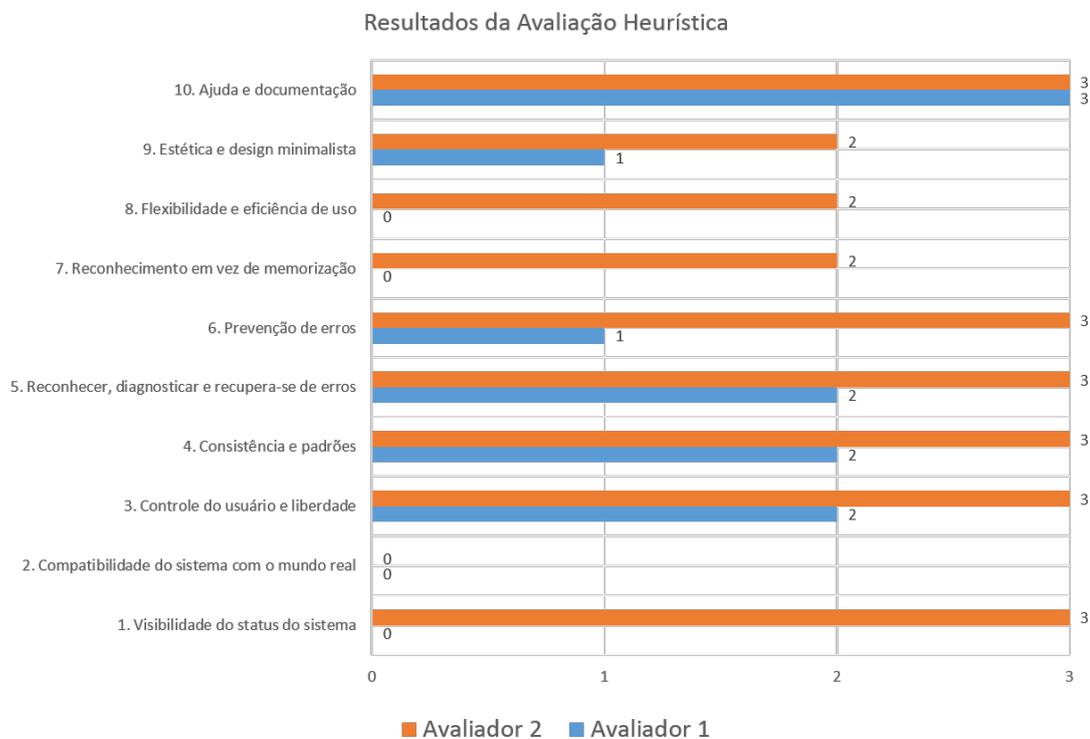
Algumas ressalvas foram destacadas, entretanto o resultado alcançado com as respostas do *checklist* demonstraram-se muito satisfatórios visto a quantidade de perguntas avaliadas de forma positiva.

## Resultados da Avaliação Heurística

Ao inspecionar os diversos elementos da interface do sistema, o avaliador atribuiu a cada problema encontrado um valor identificando sua gravidade, conforme a tabela 2.1. Quanto à localidade, as atribuições ficaram da seguinte forma: 1 – em um único local na interface; 2 – em dois ou mais locais na interface; 3 – na estrutura geral da interface; 4 – inexistente, deve ser incluído. A compilação dos resultados alcançados desta análise são apresentados no Apêndice K.

Conforme o gráfico 5.10, o sistema “DATAGUARDIAN” obteve resultados satisfatórios quanto a avaliação, porém alguns elementos devem ser ajustados, em razão de terem recebido nota três no que diz respeito à gravidade do problema, mostrando que possui alta prioridade.

**Figura 5.10 – Gráfico gerado a partir das respostas da análise heurística.**



Os princípios que apresentaram erros de maior gravidade foram **1** – Visibilidade do status do sistema, **3** – Controle do usuário e liberdade, **4** – Consistência e padrões, **5** – Ajuda os usuários a reconhecer, diagnosticar e recupera-se de erros e **10** – Ajuda e

documentação, onde este último conquistou a pior avaliação — por não possuir documentação ou manual de ajuda — adotando nota 3 em gravidade para ambos avaliadores, informando que possui alta prioridade em sua solução.

Os itens melhor avaliados foram os princípios **6** – Prevenção de erros, **7** – Reconhecimento em vez de memorização, **8** – Flexibilidade e eficiência de uso e **9** – Estética e design minimalista, que apesar de algumas ressalvas, as características positivas ficaram em destaque.

Para um teste de usabilidade mais criterioso, seria necessário uma análise mais profunda e uma equipe especializada em IHC e usabilidade composta de mais integrantes. Outro ponto importante, seria a utilização de laboratórios com ambientes controlados, para que todos os usuários pudessem efetuar os testes em condições semelhantes e sem a interferência do ambiente externo.

## 5.6 Aplicação do Questionário SUS

Aplicar o questionário SUS foi fundamental para levantar métricas quantitativas em relação ao nível de usabilidade do sistema desenvolvido. Através do questionário foi possível analisar o grau de satisfação dos usuários, e como eles se sentiram ao realizar as tarefas antes feitas sem ajuda de um sistema interativo, e agora de maneira informatizada.

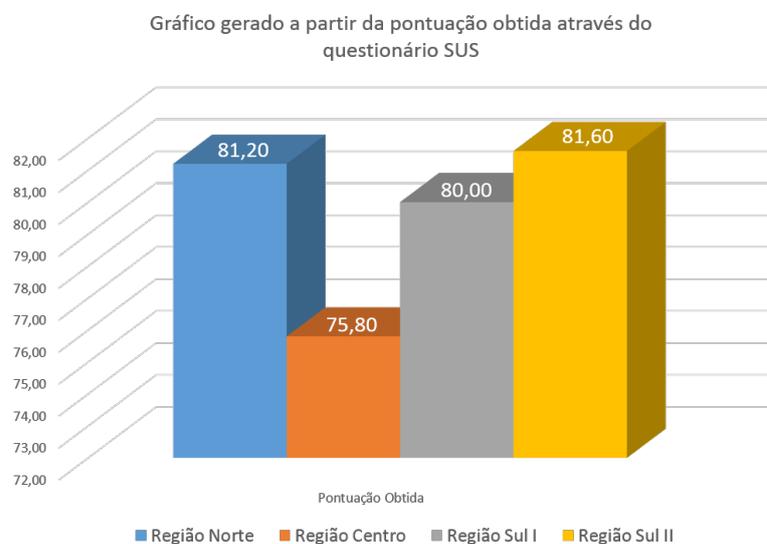
Após todos os questionários terem sido respondidos, realizou-se o cálculo das pontuações. O gráfico 5.11 apresenta as pontuações obtidas separados por Unidade Conselheira para melhor interpretação.

De acordo com este teste, pontuações acima de 68 significam que a usabilidade do sistema está acima da média. Como pode ser observado no gráfico, o sistema desenvolvido pode ser classificado como ótimo no quesito usabilidade. Após efetuar um cálculo de média aritmética entre as pontuações, obteve-se o valor 79,65. Segundo as respostas obtidas, a afirmativa de número 4 (Achei que seria necessário o apoio de um técnico para poder usar este sistema) foi a que trouxe mais avaliações negativas, sendo em muitos casos respondido como “Concordo” ou “Concordo Totalmente”. Entretanto, as outras questões foram em sua grande maioria avaliadas de forma muito positiva.

Antes que o questionário fosse respondido pelos participantes, os mesmos passaram por sessões de treinamento do sistema (Apêndice D), e tiveram que desempenhar algumas atividades pré-definidas para a devida adaptação com a ferramenta. De fato, uns tiveram mais facilidade que outros, pois possuem mais afinidade com sistemas de computador. As atividades definidas para execução dos teste foram as seguintes:

- **Tarefa 1:** Registrar denúncia;
- **Tarefa 2:** Cadastrar criança ou adolescente;
- **Tarefa 3:** Registrar atendimento; e

**Figura 5.11 – Gráfico gerado a partir da pontuação obtida através do questionário SUS.**



- **Tarefa 4:** Cadastrar casos antigos.

O processo executado para a realização de cada tarefa é demonstrado a partir dos *prints* das telas. Para realizar a **Tarefa 1** é necessário que o usuário clique no submenu intitulado “Registro Denúncia”. Ao abrir a página, o sistema abre uma caixa de diálogo perguntando ao usuário — através de botões — se ele deseja cadastrar uma nova denúncia ou apenas consultar uma existente. Dependendo da escolha, novas opções são fornecidas para o usuário. Por exemplo, caso clique em consultar denúncia existente, o sistema oferece opções para que se possa buscar denúncias no banco de dados, caso contrário, o sistema abre uma nova caixa de diálogo perguntando se essa denúncia refere-se a um caso novo ou se é reincidente. Se for um novo caso, o sistema oferece um formulário para cadastro da criança, caso contrário, o sistema proporciona opções para que possa realizar a busca dessa criança/adolescente no banco de dados. Este fluxo pode ser acompanhado visualizando as imagens no Apêndice E.

Por fim, o usuário preenche os dados do formulário com as informações referente à denúncia a ser cadastrada com possibilidade de incluir anexos.

Para concluir a **Tarefa 2** (Apêndice F), o sistema “DATAGUARDIAN” oferece várias possibilidades. O cadastro de crianças e adolescentes pode ser realizado a partir do cadastro de denúncia, no cadastro de pastas antigas ou no módulo de cadastro, na opção “Cadastro Criança/Adolescente”, esta última foi a escolhida para o teste.

Ao abrir a tela de cadastro de crianças e adolescentes, um formulário é exibido ao usuário. Este formulário é dividido em abas para melhor organização dos campos. Informações como “nome”, “data de nascimento”, “pais ou responsável” e dados de endereço são obrigatórios. Para efetivar este cadastro, basta esses dados serem preenchidos e clicar

no botão "Salvar". O cadastro de Pais/Responsável pode ser realizado na mesma tela.

A **Tarefa 3** acontece sempre que uma denúncia é registrada no sistema e esta denúncia após apurado os fatos gera um atendimento. No Apêndice G pode-se verificar o caminho percorrido para registrar este atendimento. A tela é dividida em abas, sugerindo ao usuário que para efetivação deste registro o mesmo deve ser feito em etapas. O primeiro passo é escolher qual denúncia gerou aquele atendimento. A escolha da denúncia se dá por meio de filtros de pesquisa, que ao ser selecionada, os campos do formulário do atendimento são automaticamente preenchidos. Posteriormente o usuário seleciona os direitos que foram violados. Por fim, o usuário descreve as narrativas do Agente Violador, da Vítima e do próprio conselheiro, para então selecionar as medidas protetivas que serão aplicadas àquela denúncia. Ao fim da inserção de todos os dados, a finalização do atendimento ocorre ao se clicar no botão "Salvar Atendimento".

Ao implantar o *software* no Conselho Tutelar de Palmas, os conselheiros sugeriram um módulo específico para cadastro dos casos antigos. Entendendo a necessidade, foi-se então implementado este módulo.

A **Tarefa 4** se deu por registrar as pastas antigas, e o procedimento para efetuar esta tarefa se encontra no Apêndice H. O Usuário abre a funcionalidade e preenche os dados principais do caso, como número do caso, a unidade conselheira e o conselheiro responsável por atender aquela família. Posteriormente, o usuário entra com os dados dos Pais ou Responsável pela(s) criança(s) para então, adicionar as crianças e adolescentes que compõem a família. Toda a documentação gerada para o caso cadastrado pode ser digitalizado e anexado através de uma funcionalidade que o sistema fornece. Ao fim do fluxo, o usuário clica em "Salvar Pasta".

As tarefas foram executadas garantindo um nível mais elevado de interatividade entre usuário-sistema. Concluiu-se que o sistema foi satisfatório perante este teste, e que a prática e uso periódico trará benefícios ao Conselho Tutelar de Palmas, como também, através de sugestões dos usuários, o aprimoramento e desenvolvimento de novas funcionalidades.

## Considerações finais sobre o capítulo

Neste capítulo foram apresentados os resultados obtidos através da implantação do sistema, onde apresentou-se o *software* de forma breve, os testes realizados, e a avaliação heurística de usabilidade. A combinação do *checklist* com a avaliação heurística mostrou-se bastante vantajosa, sendo as respostas do *checklist* determinantes na qualidade do *software*, e a avaliação heurística, as análises dos avaliadores.

Estes resultados se mostraram bastante satisfatórios, pois possibilitaram prever o grande potencial que o sistema terá a longo prazo, reconhecendo o "DATAGUARDIAN" como uma ferramenta de fácil manuseio e compreensão.

## 6 CONCLUSÃO E TRABALHOS FUTUROS

O presente trabalho apresentou o estudo e desenvolvimento do sistema computacional “DATAGUARDIN”. Tal sistema teve como finalidade o auxílio à gestão em direitos humanos de crianças e adolescentes, mais especificamente, o auxílio ao Conselho Tutelar da cidade de Palmas-TO, possibilitando o registro e tratamento de informações a respeito da promoção e defesa dos direitos fundamentais previstos no ECA.

Para tanto, o primeiro passo tomado foi o estudo a cerca do trabalho desenvolvido pelos conselheiros, compreendendo a fase de análise de requisitos. Foram levantados os requisitos funcionais e não funcionais, além disso, elaborou-se os diagramas de caso de uso, de classe e de atividades a fim de garantir o correto desenvolvimento do sistema.

Após essa fase, o próximo passo foi a implementação da aplicação *web* para o acompanhamento, atendimento, registro de denúncias e geração de relatórios analíticos referentes aos processos dos casos atendidos pelo Conselho Tutelar. Nessa fase aplicou-se a metodologia *Scrum*, a qual segue os princípios do manifesto ágil, otimizando o processo de desenvolvimento do *software*.

Foi observado no Capítulo de Resultados e Discussão os resultados obtidos durante a elaboração do sistema e após sua implantação em um ambiente em produção. Tais resultados compreenderam a apresentação das telas de determinadas funcionalidades, além de testes de desempenho, segurança e disponibilidade. Também foram apresentados os resultados obtidos através das avaliações da usabilidade e das resoluções do questionário SUS, demonstrando o nível satisfatório obtido na interação dos usuários com o sistema.

A partir da implantação do sistema, foi constatado que o mesmo provou-se ser de grande utilidade aos Conselheiros Tutelares, e tornaram a realização de suas atividades muito mais produtiva. Os requisitos funcionais levantados a partir de reuniões com os usuários foram atendidos de forma satisfatória, com algumas ressalvas, onde certas modificações necessárias foram realizadas, de forma a aprimorar e facilitar o manuseio do *software*.

Depois de implantado e realizado os treinamentos do sistema no Conselho Tutelar, deu-se início ao registro de informações. Em um período de 30 dias, cerca de 35 transferências de pastas para base de dados do sistema foram realizadas, mais de 50 crianças foram cadastradas e 7 novas denúncias foram registradas com suas devidas medidas protetivas aplicadas.

Diante disso, o objetivo deste trabalho foi alcançado, uma vez que, o sistema “DATAGUARDIAN” possibilitou o registro e tratamento das informações obtidas nas unidades conselheiras e a integração entre essas unidades, como também, uma maior garantia da segurança e integridade desses dados que possuem alto nível de importância, tanto para o município como para as famílias atendidas, documentando o desfecho do *software* de-

envolvido, atingindo assim, os objetivos propostos para esse trabalho de conclusão de curso.

## 6.1 Sugestões para Trabalhos Futuros

Como sugestão de trabalhos futuros, seria interessante o desenvolvimento de um aplicativo *mobile* que oferecesse as funcionalidades do sistema *web*, tendo uma versão para o usuário comum e outra para os membros do Conselho Tutelar. Uma ideia seria utilizar o GPS do dispositivo para demarcar o local das ocorrências que configuram violação de direitos violados de crianças e adolescente, possibilitando também o uso da câmera para envio de fotos como prova da violação.

Uma outra sugestão de aprimoramento do sistema, seria a implementação de um módulo de processos jurídicos, onde estes processos poderiam ser sincronizados com o sistema e-Proc<sup>1</sup>, dispensando a tarefa de usar outro sistema apenas para este tipo de serviço.

Seria interessante também como projeto futuro, a realização do processo de extração de conhecimento a partir dos dados registrados no sistema “DATAGUARDIAN”, este processo é chamado de *Knowledge Discovery in Databases* (KDD), com o objetivo de obter informações relevantes implícitas e de grande utilidade registrados na base de dados dos sistema.

Outra abordagem para expansão do sistema consistiria na implementação de algoritmos de recomendação para automatizar o processo de sugestões de Conselheiros Tutelares que irão ser responsáveis por determinado caso.

Por fim, uma outra sugestão de trabalho futuro seria a implementação de um sistema de recuperação de informações, o qual compreende na preservação de dados em uma base de dados local durante momentos de instabilidade de conexão com a internet, e quando esta restabelecida, os dados são sincronizados para o banco de dados online.

---

<sup>1</sup>e-Proc: <https://eproc.tjto.jus.br/>

## REFERÊNCIAS

- AFFAIRS, A. S. for P. **System Usability Scale (SUS)**. Department of Health and Human Services, 2013. Último acesso em 28/03/2018. Disponível em: <<https://www.usability.gov/how-to-and-tools/methods/system-usability-scale.html>>.
- BARTIÉ, A. **Garantia da qualidade de software: adquirindo maturidade organizacional**. [S.l.]: Elsevier Academic, 2002.
- BASTOS, A. et al. **Base de conhecimento de teste de software**. [S.l.]: Martins, 2007.
- BENYON, D. **Designing interactive systems: a comprehensive guide to HCI and interaction design**. [S.l.]: Pearson Education Limited, Reino Unido, 2011. 49-55 p.
- BENYON, D. **Designing interactive systems: a comprehensive guide to HCI and interaction design**. [S.l.]: Pearson Education Limited, Reino Unido, 2011. 6 p.
- BENYON, D. **Designing interactive systems: a comprehensive guide to HCI and interaction design**. [S.l.]: Pearson Education Limited, Reino Unido, 2011. 150 p.
- BRASIL, P. **Cresce número de municípios no Brasil**. 2013. <<http://www.brasil.gov.br/economia-e-emprego/2013/06/cresce-numero-de-municipios-no-brasil-em-2013>>. Último acesso em 15/5/2018.
- BRASILEIRINHO, O. **Conselho tutelar – quando surgiu e para que serve**. 2016. <<https://www.obrasileirinho.com.br/denunciar-crime-contracrianca-e-facil-veja-como-conselho-tutelar-quando-surgiu-e-para-que-serve/>>. Último acesso em 15/5/2018.
- BROOKE, J. **Sus - a quick and dirty usability scale**. 1986. Disponível em: <<http://www.usabilitynet.org/trump/documents/Suschapt.doc>>.
- BROOKE, J. **SUS - A quick and dirty usability scale**. 1986. <<http://www.usabilitynet.org/trump/documents/Suschapt.doc>>. Último acesso em 28/03/2018.
- CALZA, T. Z. **TRADUÇÃO E ADAPTAÇÃO DE SOFTWARE PARA O AUXÍLIO NA IDENTIFICAÇÃO DE MAUS TRATOS EM CRIANÇAS E ADOLESCENTES**. Universidade Federal do Rio Grande do Sul - Instituto de Psicologia, 2014. Disponível em: <<https://www.lume.ufrgs.br/bitstream/handle/10183/102332/000933068.pdf?sequence=1>>.
- DESLANDES, S. et al. **Indicadores das ações municipais para a notificação e o registro de casos de violência intrafamiliar e exploração sexual de crianças e adolescentes**. Cadernos de Saúde Pública, 2011. Disponível em: <<http://www.scielo.br/pdf/csp/v27n8/18.pdf>>.
- DEV MEDIA. **Criptografia MD5**. DevMedia, 2012. Disponível em: <<https://www.devmedia.com.br/criptografia-md5/2944>>.

DIGIÁCOMO, M. J.; DIGIÁCOMO, I. de A. **Estatuto da Criança e Adolescente Anotado e Interpretado**. Ministério Público do Estado do Paraná, 2010. Disponível em: <[http://www.mpdft.mp.br/portal/pdf/unidades/promotorias/pdij/Legislacao%20e%20Jurisprudencia/ECA\\_comentado.pdf](http://www.mpdft.mp.br/portal/pdf/unidades/promotorias/pdij/Legislacao%20e%20Jurisprudencia/ECA_comentado.pdf)>.

DILILLO, D. et al. Development and initial psychometric properties of the computer assisted maltreatment inventory (cami): A comprehensive self-report measure of child maltreatment history. v. 34, p. 305–17, 03 2010.

DOBKE, V. M.; SANTOS, S. S. dos; DELL'AGLIO, D. D. Abuso sexual intrafamiliar: da notificação ao depoimento no contexto processual-penal. *Temas em Psicologia*, v. 18, n. 1, 2010. Disponível em: <<http://pepsic.bvsalud.org/pdf/tp/v18n1/v18n1a14.pdf>>.

ELIZA, R.; LAGARES, V. **Teste de Segurança - agregando confiança ao software**. 2013. <https://www.devmedia.com.br/teste-de-seguranca-agregando-confianca-ao-software/27792>.

FRANCETO, S. **ESPECIFICAÇÃO E IMPLEMENTAÇÃO DE UMA FERRAMENTA PARA ELICITAÇÃO DE REQUISITOS DE SOFTWARE BASEADA NA TEORIA DA ATIVIDADE**. Universidade Metodista de Piracicaba – UNIMEP, 2005. 17 p. Último acesso em 14/05/2017. Disponível em: <<https://www.unimep.br/phpg/bibdig/pdfs/2006/VMEPWGALFNMD.pdf>>.

GONÇALVES, H. S.; FERREIRA, A. L. A notificação da violência intrafamiliar contra crianças e adolescentes por profissionais de saúde. *Cadernos de Saúde Pública*, v. 18, n. 1, 2002. Disponível em: <<http://www.scielo.br/pdf/csp/v18n1/8168.pdf>>.

JÚNIOR, M. G. C. **Estudo de mecanismos e fatores que impactam no desempenho de aplicações Web**. 2017. Disponível em: <[http://www.monografias.ufop.br/bitstream/35400000/622/1/MONOGRAFIA\\_EstudoMecanismosFatores.pdf](http://www.monografias.ufop.br/bitstream/35400000/622/1/MONOGRAFIA_EstudoMecanismosFatores.pdf)>. Acesso em: 02/04/2018.

KARAT, C.-m. Cost-benefit analysis of usability engineering techniques. v. 34, 10 1990.

KRONE, C. Validação de heurísticas de usabilidade para celulares touchscreen. v. 10, 01 2013. Disponível em: <[http://www.gqs.ufsc.br/wp-content/uploads/2013/07/WorkingPaper\\_WP\\_GQS\\_01-2013\\_v10.pdf](http://www.gqs.ufsc.br/wp-content/uploads/2013/07/WorkingPaper_WP_GQS_01-2013_v10.pdf)>.

KWAADSTENIET, L. de et al. Improved decision making about suspected child maltreatment: Results of structuring the decision process. *Children and Youth Services Review*, 2013. Disponível em: <<http://www.socsci.ru.nl/ciliaw/Site%20Cilia/Children.pdf>>.

LEFFINGWELL, D. **Agile Software Requirements**. [S.l.]: Addison-Wesley, 2011. ISBN 978-0-321-63584-6.

MARTINS, E. **Apostila sobre Testes de Software**. [S.l.]: Instituto de Computação, UNICAMP, 2005.

MOBILE, I. **Proteja Brasil**. Último acesso em 17/05/2017. Disponível em: <[https://play.google.com/store/apps/details?id=br.com.ilhasoft.protejaBrasil&hl=pt\\_BR](https://play.google.com/store/apps/details?id=br.com.ilhasoft.protejaBrasil&hl=pt_BR)>.

MOBILE, I. **Proteja Brasil**. Último acesso em 17/05/2017. Disponível em: <<http://www.protejabrasil.com.br/br/>>.

NIELSEN, J. **Usability Engineering**. [S.l.]: Morgan Kaufmann, 1993. ISBN 0-12-518406-9.

NIELSEN, J. **Usability Inspection Methods**. [S.l.]: John Wiley & Sons, 1994. ISBN 0-471-01877-5.

NIELSEN, J. **Why You Only Need to Test with 5 Users**. 2000. Último acesso em 30/03/2018. Disponível em: <<https://www.nngroup.com/articles/why-you-only-need-to-test-with-5-users/>>.

NORMAN, D. A. **O Design do Dia-a-Dia**. [S.l.]: Rocco, 2006.

ORACLE. **O banco de dados de código aberto mais conhecido no mundo**. <<https://www.oracle.com/br/mysql/index.html>>. Último acesso em 15/05/2017.

PÁDUA, C. I. P. da Silva e. Engenharia de usabilidade material de referência. 08 2012.

PEREZ, J. R. R.; PASSONE, E. F. Políticas sociais de atendimento às crianças e aos adolescentes no brasil. **Cadernos de Pesquisa**, v. 40, n. 140, 2010. Disponível em: <[http://www.scielo.br/scielo.php?script=sci\\_arttext&pid=S0100-15742010000200017&lng=en&nrm=iso&tlng=pt](http://www.scielo.br/scielo.php?script=sci_arttext&pid=S0100-15742010000200017&lng=en&nrm=iso&tlng=pt)>.

PFLEEGER, S. L. **Engenharia de Software: Teoria e Prática. 2a ed.** [S.l.]: Prentice Hall, 2004.

PISA, P. **O que é Hash?** Techtudo, 2012. Disponível em: <<http://www.techtudo.com.br/artigos/noticia/2012/07/o-que-e-hash.html>>.

PRATES, R. O.; BARBOSA, S. D. J. **Avaliação de Interfaces de Usuário - Conceitos e Métodos**. 2003. Último acesso em 31/03/2018. Disponível em: <[http://www-di.inf.puc-rio.br/~simone/files/JAI2003\\_avaliacao\\_s.pdf](http://www-di.inf.puc-rio.br/~simone/files/JAI2003_avaliacao_s.pdf)>.

PRESSMAN, R. S. **Engenharia de Software. 6a ed.** [S.l.]: McGraw-Hill, Sao Paul, 2006.

PRIKLADNICKI, R.; WILLI, R.; MILANI, F. **Métodos Ágeis para Desenvolvimento de Software**. [S.l.]: Bookman Editora, 2013.

REPÚBLICA, P. da. **LEI 8.069/1990 (LEI ORDINÁRIA) 13/07/1990 01:00:00**. 1990. <[http://www.planalto.gov.br/ccivil\\_03/leis/l8069.htm](http://www.planalto.gov.br/ccivil_03/leis/l8069.htm)>. Último acesso em 15/5/2018.

REPÚBLICA, S. de Direitos Humanos da Presidência da. **Cadastro Nacional dos Conselhos Tutelares**. 2012. <<http://www.mdh.gov.br/assuntos/criancas-e-adolescentes/programas/fortalecimento-de-conselhos/cadastro-nacional-dos-conselhos-tutelares-1>>. Último acesso em 15/05/2018.

REZENDE, D. A. **Engenharia de Software Empresarial**. [S.l.]: Rio de Janeiro: Brasport, 1997.

RIOS, E.; MOREIRA, T. **Teste de Software**. Rio de Janeiro, RJ: Alta Books, 2013. ISBN 978-85-7608-775-5.

RUBIN, K. S. **Essential Scrum: a practical guide to the most popular agile process**. [S.l.]: Pearson Education, 2012.

SAURO, J. **SUS Guide & Calculator Package**. 2011. <<https://measuringu.com/product/suspack/>>.

SAURO, J. **SUS Score**. 2011. <<https://measuringu.com/sus/>>. Último acesso em 02/4/2018.

SCHWABER, K. **Agile Project Management With Scrum**. [S.l.]: Microsoft, 2004.

SIPIA. **SIPIA - Sistema de Informações para Infância e Adolescência**. Último acesso em 11/05/2017. Disponível em: <<http://www.sipia.gov.br/CT/?x=z8CMXhVtEY8BPmqMYE886g>>.

SIPIA. **SIPIA - Sistema de Informações para Infância e Adolescência**. 1997. Último acesso em 11/05/2017. Disponível em: <[http://www.sipia.gov.br/CT/?x=ER5jhmbbJ\\*Q10cvlq0NQDw](http://www.sipia.gov.br/CT/?x=ER5jhmbbJ*Q10cvlq0NQDw)>.

SIPIA. **Sobre o Sopia**. 1998. <<http://www.sipia.gov.br/CT/?x=X4IJjIS4zj8PJxADK2TNw>>. Último acesso em 17/02/2017.

SISTEMAS, S. E. de Softwares e. **Desenvolvimento completo de software**. 2014. <[http://www.synergia.dcc.ufmg.br/?page\\_id=115](http://www.synergia.dcc.ufmg.br/?page_id=115)>. Último acesso em 15/05/2018.

SOARES, M. dos S. **Comparação entre Metodologias Ágeis e Tradicionais para o Desenvolvimento de Software**. [S.l.]: Unipac - Universidade Presidente Antônio Carlos, 2014.

SOMMERVILLE, I. **Engenharia de Software 9ª Edição**. [S.l.]: Pearson Education, Inc., 2011. 38-77 p. ISBN 978-85-7936-108-1.

STANDARDIZATION, I. O. for. **Ergonomics of human-system interaction – Part 210: Human-centred design for interactive systems**. 2010. Último acesso em 28/03/2018. Disponível em: <<https://www.iso.org/standard/52075.html>>.

SUASNET. **SUASnet - Módulo Conselho Tutelar**. Último acesso em 11/05/2017. Disponível em: <<https://www.suasnet.com.br/site/?p=ctutelar>>.

SUASNET. **SUASnet - Sistema de Assistência Social de Municípios**. 2006. Último acesso em 11/05/2017. Disponível em: <<https://www.suasnet.com.br/site/?p=home>>.

TEIXEIRA, F. **O que é o SUS (System Usability Scale) e como usá-lo em seu site**. 2015. <<https://brasil.uxdesign.cc/o-que-%C3%A9-o-sus-system-usability-scale-e-como-us%C3%A1-lo-em-seu-site-6d63224481c8>>.

TELEFÔNICA, F. **O dia-a-dia do Conselheiro Tutelar**. 2016. Último acesso em 17/05/2017. Disponível em: <<http://fundacaotelefonica.org.br/promenino/trabalho infantil/conselhotutelar/o-dia-a-dia-do-conselheiro-tutelar/>>.

TESTARME. **TESTE DE PERFORMANCE ou TESTE DE DESEMPENHO**. 2015. <<https://www.testar.me/single-post/2015/11/23/TESTE-DE-PERFORMANCE-ou-TESTE-DE-DESEMPENHO>>. Último acesso em 02/04/2018.

WIKIPÉDIA. **Escala Likert**. Último acesso em 03/01/2018. Disponível em: <[https://pt.wikipedia.org/wiki/Escala\\_Likert](https://pt.wikipedia.org/wiki/Escala_Likert)>.

## APÊNDICE I - REQUISITOS FUNCIONAIS E NÃO FUNCIONAIS

Figura 1 – RF001 - Cadastro de Usuários

Identificador	<b>RF 001</b>		
Nome	Cadastrar Usuários		
Módulo	Controle de Acesso		
Data de Criação	13/05/2017	<b>Autor</b>	Rodrigo Magalhães
Versão	1	<b>Prioridade</b>	Essencial
Descrição	<p>O sistema deve permitir o cadastro de usuários, sendo que estes usuários podem ser conselheiros, administradores, coordenadores como também assistentes que complementam o quadro de funcionários do Conselho Tutelar.</p> <p>Cada usuário terá seu perfil, e as devidas funcionalidades ligadas a este perfil, formando assim o controle de acesso do usuário.</p>		

Figura 2 – RF002 - Aplicar Perfis aos Usuários

Identificador	<b>RF 002</b>		
Nome	Aplicar perfis aos usuários		
Módulo	Controle de Acesso		
Data de Criação	13/05/2017	<b>Autor</b>	Rodrigo Magalhães
Versão	1	<b>Prioridade</b>	Essencial
Descrição	<p>O controle de acesso de cada usuário é definido por seus perfis. Um tipo de perfil para cada módulo do sistema. Sendo que este perfil é formado pelas funcionalidades (telas) que estarão disponíveis para o usuário. Essas funcionalidades já devem estar cadastradas no sistema.</p> <p>Um exemplo, seria um perfil "NIVEL_1" ligado ao Módulo Cadastro, onde apenas a funcionalidade de Cadastro de Crianças e Adolescentes estaria vinculada a este perfil, sendo assim, ao aplicar o perfil "NIVEL_1" ao usuário, este usuário teria acesso a apenas a funcionalidade vinculada.</p> <p>O sistema deve ter uma funcionalidade apenas para vincular perfis aos usuários cadastrados. Módulos e Funcionalidades, deverão já estar presentes no banco de dados do sistema.</p>		

Figura 3 – RF003 - Cadastro de Responsáveis

Identificador	<b>RF 003</b>		
Nome	Cadastro de Responsáveis		
Módulo	Cadastro		
Data de Criação	13/05/2017	<b>Autor</b>	Rodrigo Magalhães
Versão	1	<b>Prioridade</b>	Essencial
Descrição	Os responsáveis são os tutores legais das crianças e adolescentes, sejam eles os pais, outro integrante da família ou até mesmo o poder público. Para cadastrar uma criança/adolescente, é fundamental que seu responsável esteja cadastrado no sistema também, pois é este que irá ser responsável por assinar termos de responsabilidade como se responsabilizar em manter a integridade e a dignidade de seus protegidos.		

Figura 4 – RF004 - Cadastro de Crianças e Adolescentes

Identificador	<b>RF 004</b>		
Nome	Cadastro de Crianças e Adolescentes		
Módulo	Cadastro		
Data de Criação	13/05/2017	<b>Autor</b>	Rodrigo Magalhães
Versão	1	<b>Prioridade</b>	Essencial
Descrição	<p>Um Conselho Tutelar trabalha em prol dos direitos das crianças e adolescentes. E é de suma importância, que ao ser reconhecido que alguns dos direitos da mesma foram violados, o seu registro no sistema, para que os conselheiros possam tomar as devidas providências.</p> <p>Essa funcionalidade, deve permitir o cadastro de todas as informações relevantes, como dados pessoais, situação de saúde, situação escolar e familiar, assim como, o anexo de arquivos.</p> <p>É obrigatório que seja vinculado o Responsável para efetuar o cadastramento da Criança/Adolescente.</p>		

Identificador	<b>RF 005</b>		
Nome	Registrar Denúncia		
Módulo	Denúncia		
Data de Criação	13/05/2017	<b>Autor</b>	Rodrigo Magalhães
Versão	1	<b>Prioridade</b>	Essencial
Descrição	<p>Conselhos Tutelares recebem denúncias de violação contra crianças/adolescentes diariamente. O registro dessas denúncias é fundamental para a possível erradicação dessas práticas.</p> <p>O sistema deve permitir o cadastro de denúncias com todas as informações que podem ser passadas pelo denunciante, também liberar que esta funcionalidade esteja habilitada para usuários comuns através de um link externo por exemplo.</p>		

Figura 5 – RF005 - Registrar Denúncia

Identificador	<b>RF 006</b>		
Nome	Acompanhamento de Denúncias		
Módulo	Denúncia		
Data de Criação	13/05/2017	<b>Autor</b>	Rodrigo Magalhães
Versão	1	<b>Prioridade</b>	Importante
Descrição	Ao registrar uma denúncia no sistema, uma chave, ou um número de protocolo deve ser gerado com o objetivo de poder consultar o estado dessa denúncia e quais foram as providências tomadas.		

Figura 6 – RF005 - Acompanhamento de Denúncias

Figura 7 – RF007 - Registro de Casos e Reincidências

Identificador	<b>RF 007</b>		
Nome	Construção de Casos e Reincidências		
Módulo	Denúncia		
Data de Criação	13/05/2017	<b>Autor</b>	Rodrigo Magalhães
Versão	1	<b>Prioridade</b>	Essencial
Descrição	<p>Sempre que uma denúncia é registrada, dois a três conselheiros se responsabilizam para visitar o possível local da violação retratada a fim de apurar os fatos ocorridos. Sendo verdade os acontecimentos, os conselheiros abrem um Caso, se esta criança/adolescente for reincidente, então é registrado uma reincidência.</p> <p>No cadastro de caso/reincidência, vincula-se a denúncia geradora do mesmo, as crianças/adolescentes que sofreram seus direitos violados, como também os próprios direitos violados.</p> <p>O usuário poderá aplicar as medidas protetivas no ato do registro deste caso/reincidência.</p>		

Identificador	<b>RF 008</b>		
Nome	Aplicar medidas protetivas aos casos/reincidências registrados		
Módulo	Denúncia		
Data de Criação	13/05/2017	<b>Autor</b>	Rodrigo Magalhães
Versão	1	<b>Prioridade</b>	Essencial
Descrição	<p>As medidas protetivas são providências tomadas pelo conselho tutelar com o objetivo de resguardar a criança/adolescente.</p> <p>Cada caso/reincidência possuirá um histórico de medidas aplicadas, sendo estas podendo ser alteradas ou adicionadas sempre que houver necessidade.</p> <p>Ao aplicar essas medidas protetivas, o sistema deve gerar o relatório informando as medidas que devem ser aplicadas com campos para assinatura dos conselheiros e do responsável pela(s) criança(s)/adolescente(s).</p>		

Figura 8 – RF008 - Aplicar Medidas Protetivas

**Figura 9 – RF009 - Registro de Solicitações**

<b>Identificador</b>	<b>RF 009</b>		
<b>Nome</b>	Registro de Solicitações		
<b>Módulo</b>	Solicitação		
<b>Data de Criação</b>	13/05/2017	<b>Autor</b>	Rodrigo Magalhães
<b>Versão</b>	1	<b>Prioridade</b>	Essencial
<b>Descrição</b>	<p>O sistema deve permitir o registro de solicitações por parte dos conselheiros.</p> <p>Estas solicitações possuem categorias para que possam ser gerados futuros relatórios e separar de forma mais organizada as solicitações efetivadas.</p> <p>É necessário possibilitar a inserção de anexos à solicitação quando assim for conveniente, como também, o registro de respostas a mesma.</p> <p>Permitir consulta de histórico de solicitações cadastradas através de filtros, como período entre datas, categorias ou criança/adolescente.</p>		

**Figura 10 – RF010 - Registro de Atividades Extras**

<b>Identificador</b>	<b>RF 010</b>		
<b>Nome</b>	Registro de Atividades Extras		
<b>Módulo</b>	Atividade		
<b>Data de Criação</b>	13/05/2017	<b>Autor</b>	Rodrigo Magalhães
<b>Versão</b>	1	<b>Prioridade</b>	Essencial
<b>Descrição</b>	<p>O sistema deve permitir o registro de atividades extras desempenhadas pelos conselhos tutelares.</p> <p>Estas atividades possuem categorias para que possam ser gerados futuros relatórios e separar de forma mais organizada as atividades efetivadas.</p> <p>Permitir consulta de histórico de atividades cadastradas através de filtros, como período entre datas e categorias.</p>		

**Figura 13 – RNF003 - Manutenção de Usuários**

<b>Identificador</b>	<b>RNF 003</b>		
<b>Nome</b>	Manutenção de Usuários e Senhas		
<b>Módulo</b>	Controle de Acesso		
<b>Data de Criação</b>	13/05/2017	<b>Autor</b>	Rodrigo Magalhães
<b>Versão</b>	1	<b>Prioridade</b>	Importante
<b>Descrição</b>	<p>O sistema deve possibilitar que o usuário altere algumas informações de seu cadastro, como dados de endereço e alteração de senha.</p> <p>Quando o usuário esquecer de sua senha, o administrador deve resetar esta senha para a senha padrão, onde será solicitado uma nova senha no próximo login.</p>		

Figura 11 – RNF001 - Extensão de Relatórios

Identificador	<b>RNF 001</b>		
Nome	Extensões de relatórios		
Módulo	Relatórios		
Data de Criação	13/05/2017	<b>Autor</b>	Rodrigo Magalhães
Versão	1	<b>Prioridade</b>	Importante
Descrição	Todos os relatórios do sistema devem ser no formato PDF por seu tipo ser universal e não precisar de aplicativos externos para serem executados. Alguns destes, gerar também em <u>xlsx</u> (Excel).		

Figura 12 – RNF002 - Disponibilidade do Sistema

Identificador	<b>RNF 002</b>		
Nome	Disponibilidade do Sistema		
Módulo			
Data de Criação	13/05/2017	<b>Autor</b>	Rodrigo Magalhães
Versão	1	<b>Prioridade</b>	Importante
Descrição	O sistema deve estar disponível 24h por dia podendo ser acessado de qualquer lugar bastando apenas conexão com a internet.		

Figura 14 – RNF004 - Utilização de captcha para consulta de denúncias

Identificador	<b>RNF 004</b>		
Nome	Utilização de captcha para consultar denúncias através de acesso externo		
Módulo	Segurança		
Data de Criação	13/05/2017	<b>Autor</b>	Rodrigo Magalhães
Versão	1	<b>Prioridade</b>	Importante
Descrição	Importante verificação de captcha ao solicitar status de denúncia através de acesso externo.		

Figura 15 – RNF005 - Responsividade

Identificador	<b>RNF 005</b>		
Nome	Responsividade		
Módulo	Usabilidade		
Data de Criação	13/05/2017	<b>Autor</b>	Rodrigo Magalhães
Versão	1	<b>Prioridade</b>	Importante
Descrição	Todas as interfaces de usuário devem possibilitar a responsividade web (técnica de estruturação que se adapta a todos os tipos de navegadores), possibilitando o acesso de qualquer dispositivo.		

Figura 16 – RNF006 - Padrões de Projeto

Identificador	<b>RNF 006</b>		
Nome	Padrões de Projeto		
Módulo	Padrões		
Data de Criação	13/05/2017	<b>Autor</b>	Rodrigo Magalhães
Versão	1	<b>Prioridade</b>	Importante
Descrição	Desenvolver o sistema utilizando as técnicas de orientação a objetos fornecidos pela linguagem proposta.		

## APÊNDICE B - DIAGRAMAS DE CASO DE USO

Figura 17 – Caso de Uso - Cadastro Responsável

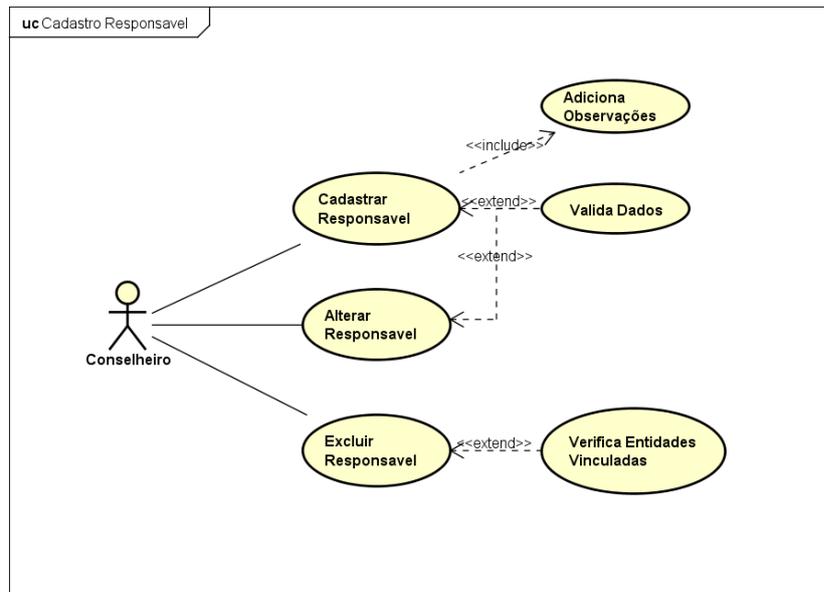


Figura 18 – Especificação Caso de Uso - Cadastro de Responsáveis

UseCase : UC – Cadastro/Alteração/Exclusão de Responsável	
Fase	Fluxo Principal
1	Conselheiro preenche os campos referentes ao Responsável a ser cadastrado.
2	Clica no botão cadastrar. Exibe mensagem de cadastro realizado.
3	Caso queira alterar ou excluir um Responsável já cadastrado, buscar o mesmo através do nome ou CPF.
4	Selecionar Responsável
5	Alterar os dados que deseja e clicar no botão de alterar. Ou clicar no botão de excluir, caso deseja efetuar a exclusão do Responsável selecionado.

Figura 19 – Caso de Uso - Cadastro Criança e Adolescente

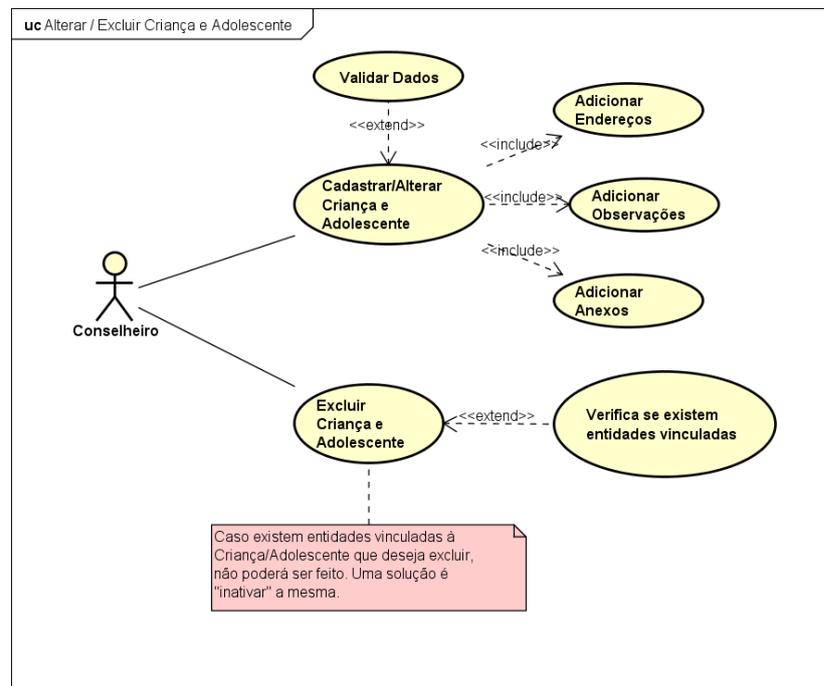


Figura 20 – Especificação Caso de Uso - Cadastro Criança e Adolescente

UseCase : UC – Cadastro/Alteração/Exclusão de Criança/Adolescente	
<b>Fase</b>	<b>Fluxo Principal</b>
1	Conselheiro preenche os campos referentes a criança/adolescente.
2	Clica no botão cadastrar. Exibe mensagem de cadastro realizado.
3	Caso queira alterar ou excluir uma criança/adolescente já cadastrado, buscar a mesma através do nome ou CPF.
4	Selecionar criança/adolescente
5	Alterar os dados que deseja e clicar no botão de alterar. Ou clicar no botão de excluir, caso deseja efetuar a exclusão da criança/adolescente.
<b>Fase</b>	<b>Fluxo Alternativo</b>
1	Vincular Responsável a criança/adolescente.
2	Cadastrar Responsável caso não existir. Aciona caso de uso UC – Cadastro Responsável.
3	Adicionar Endereços.
4	Adicionar Observações.
5	Adicionar Anexos.
6	Criança não pode ser excluída se tiver casos vinculados a mesma. Exibe mensagem de erro.

Figura 21 – Caso de Uso - Registro de Denúncia

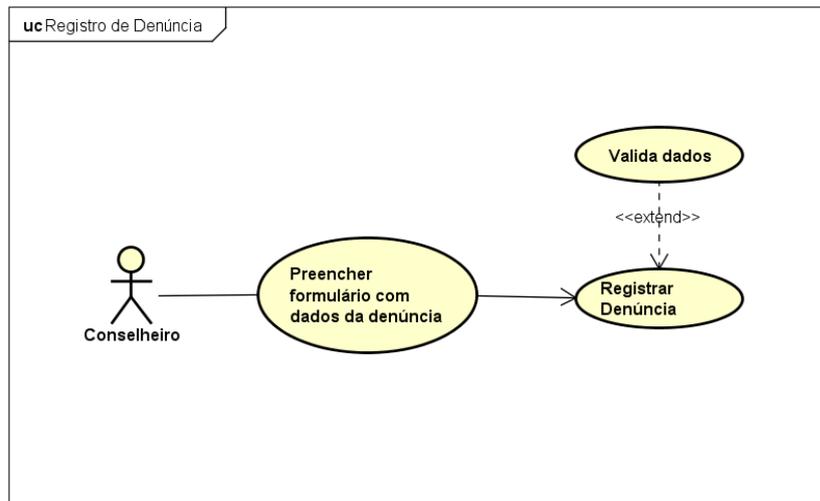


Figura 22 – Especificação Caso de Uso - Registro de Denúncia

UseCase : UC – Registro Denúncia	
<b>Fase</b>	<b>Fluxo Principal</b>
1	O formulário é preenchido com as informações obtidas pelo denunciante.
2	Clica no botão cadastrar. Exibe mensagem de cadastro realizado.
3	Caso queira alterar ou excluir uma Criança/Adolescente já cadastrado, buscar a mesma através do nome ou CPF.
4	Selecionar criança/adolescente
5	Alterar os dados que deseja e clicar no botão de alterar. Ou clicar no botão de excluir, caso deseje efetuar a exclusão da Criança/Adolescente.
<b>Fase</b>	<b>Fluxo Alternativo</b>
1	Ao cadastrar nova Denúncia, é verificado se todos os campos foram preenchidos. Caso contrário Exibe Mensagem de erro informando o que está faltando para efetivação do registro.
2	Há possibilidade de se consultar uma Denúncia pela chave que a representa.

Figura 23 – Caso de Uso - Caso/Reincidência

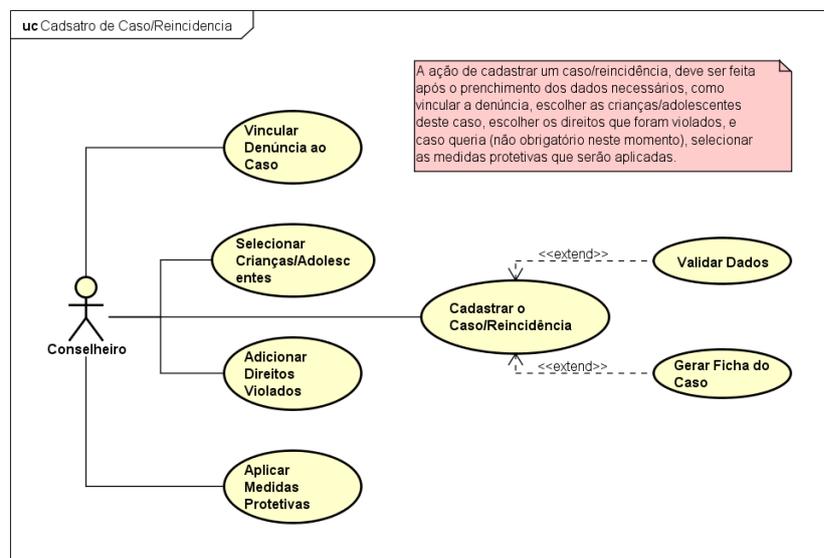


Figura 24 – Especificação Caso de Uso - Caso/Reincidência

UseCase : UC – Registro de Caso/Reincidência	
<b>Fase</b>	<b>Fluxo Principal</b>
1	O formulário é preenchido com as informações relevantes ao Caso/Reincidência a ser cadastrado.
2	É necessário a vinculação da denúncia ao caso, assim como a(s) Criança(s)/Adolescente(s) que sofreram os direitos violados, e os próprios Direitos Violados.
3	Ao clicar no botão de cadastrar, mensagem de cadastro com sucesso é exibida.
<b>Fase</b>	<b>Fluxo Alternativo</b>
1	Pode ser vinculado as medidas protetivas no ato do cadastro de Caso/Reincidência.
2	A ficha no formato PDF é gerada ao ser efetivado com sucesso o registro.
3	Todos os dados que devem ser vinculados ao Caso/Reincidência, devem ter sido previamente cadastrados no sistema.

Figura 25 – Caso de Uso - Solicitações Diversas

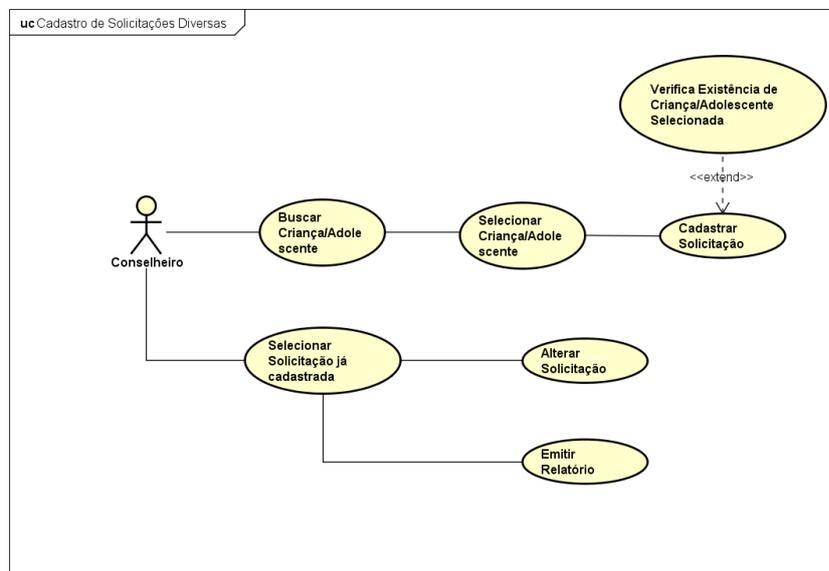


Figura 26 – Especificação Caso de Uso - Registro de Solicitações

UseCase : UC – Solicitações Diversas	
<b>Fase</b>	<b>Fluxo Principal</b>
1	O formulário é preenchido com as informações referentes a Solicitação.
2	Pesquisa criança/adolescente referente a esta Solicitação.
3	Seleciona Criança/Adolescente.
4	Clica em cadastrar Solicitação. Mensagem de sucesso é exibida.
<b>Fase</b>	<b>Fluxo Alternativo</b>
1	O usuário pode adicionar anexos a esta Solicitação.
2	Não será permitido o registro da Solicitação sem ter selecionado a Criança/Adolescente.

Figura 27 – Caso de Uso - Atividade Extra

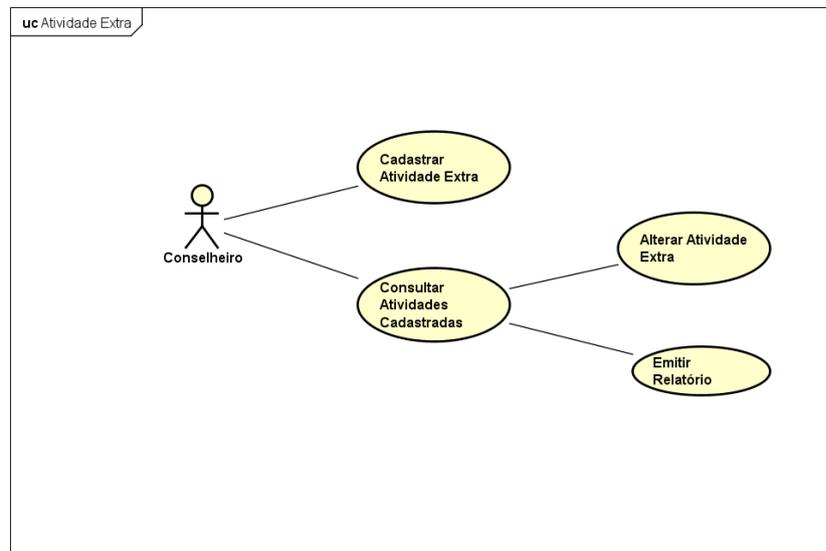


Figura 28 – Especificação Caso de Uso - Atividades Extras

UseCase : UC – Cadastro de Atividades Extras	
Fase	Fluxo Principal
1	O formulário é preenchido com as informações referentes a Atividade Extra efetivada ou que será ainda realizada.
2	Selecionam-se os Conselheiros que irão participar da Atividade Extra.
3	Ao clicar no botão de cadastrar, mensagem de cadastro com sucesso é exibida.
Fase	Fluxo Alternativo
1	Para alterar ou excluir determinada Atividade Extra, basta selecionar a mesma na tabela que está presente na mesma tela de cadastro de Atividades. Os campos serão preenchidos automaticamente, possibilitando assim a alteração ou exclusão.

## APÊNDICE C - DIAGRAMAS DE ATIVIDADE

Figura 29 – Diagrama de Atividade - Cadastro de Responsável

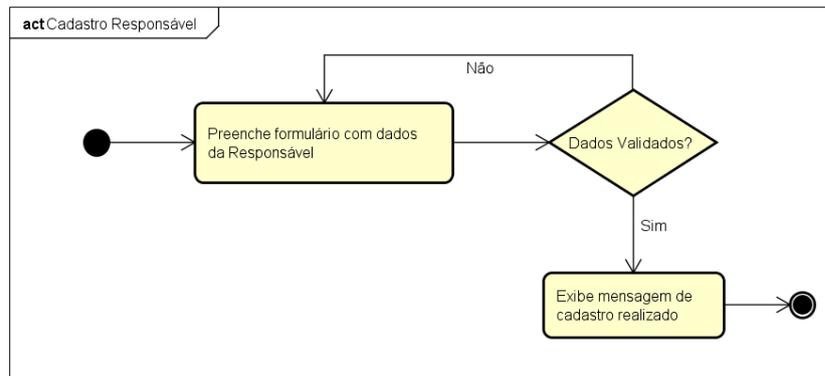


Figura 30 – Diagrama de Atividade - Cadastro de Criança e Adolescente

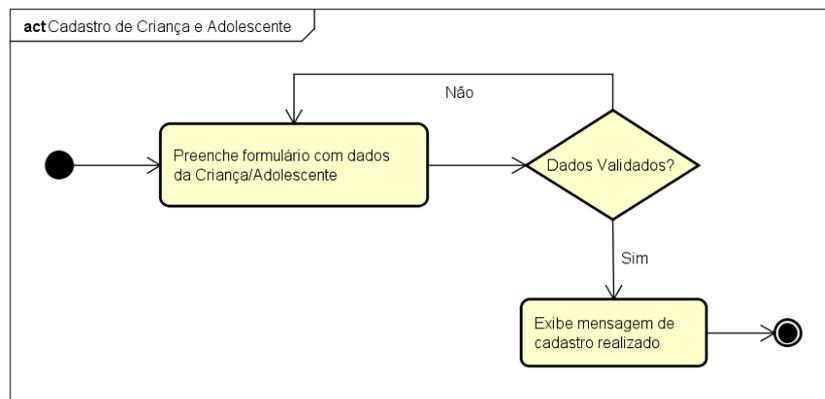
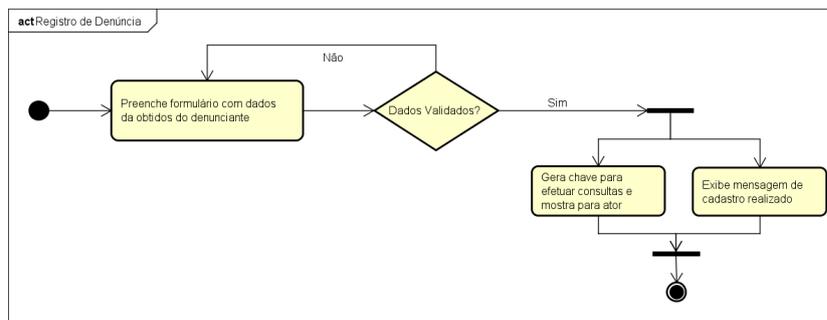


Figura 31 – Diagrama de Atividade - Registro de Denúncias

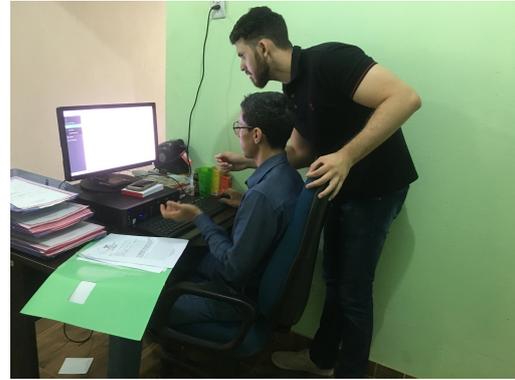




## APÊNDICE D - FOTOS DOS TREINAMENTOS REALIZADOS



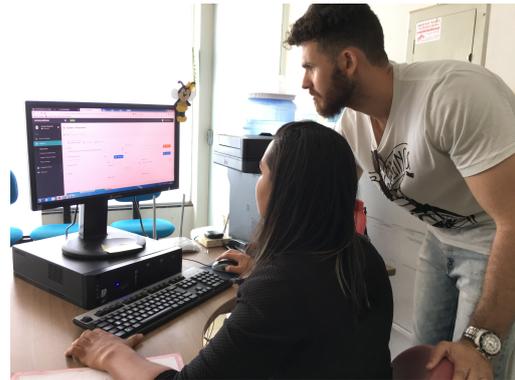
(a) a



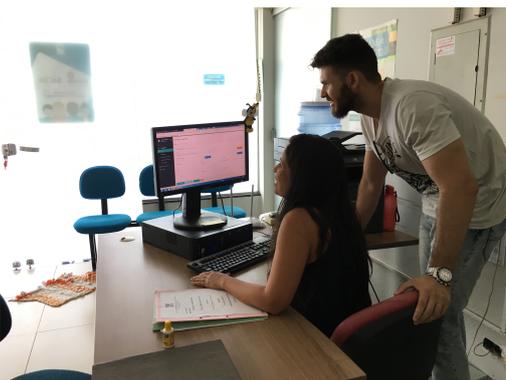
(b) a



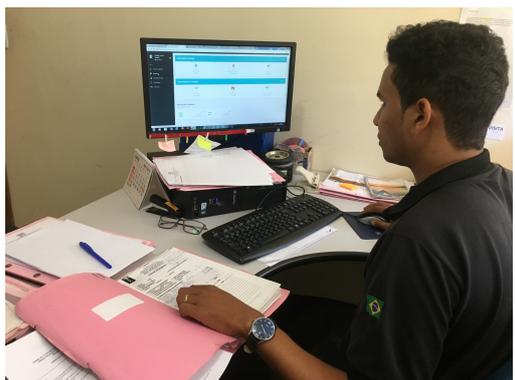
(c) a



(d) a



(e) a



(f) a

Figura 33 – Fotos capturadas em treinamentos realizados no Conselho Tutelar de Palmas. Fonte: (Produção do Próprio Autor)

## APÊNDICE E - TAREFA 1: CADASTRO DENÚNCIA

Figura 34 – Selecionar Funcionalidade

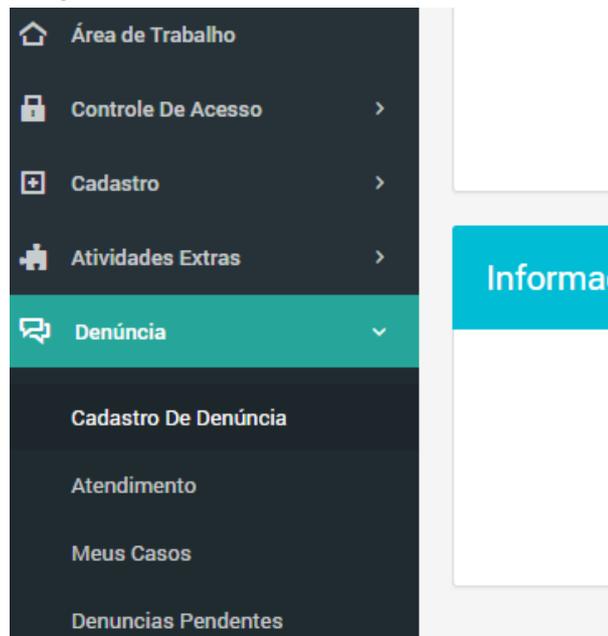


Figura 35 – O sistema oferece ao usuário duas opções para prosseguir no cadastro

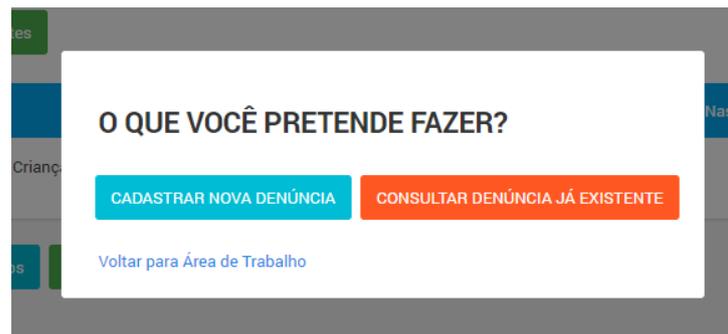


Figura 36 – O usuário pode pesquisar denúncias já cadastradas a partir dos filtros



Figura 37 – O sistema pergunta se a denúncia é referente a uma criança reincidente ou não

Clique no X para voltar

A CRIANÇA/ADOLESCENTE É REINCIDENTE?

SIM NÃO

Figura 38 – Se a criança/adolescente for reincidente, o sistema oferece opções para pesquisar tal pessoa

Buscar Crianças ou Adolescentes

BUSCAR CRIANÇA/ADOLESCENTE POR:

NOME DA CRIANÇA

NOME DO RESPONSÁVEL

Figura 39 – Tela onde pode-se pesquisar crianças e adolescentes pelo nome

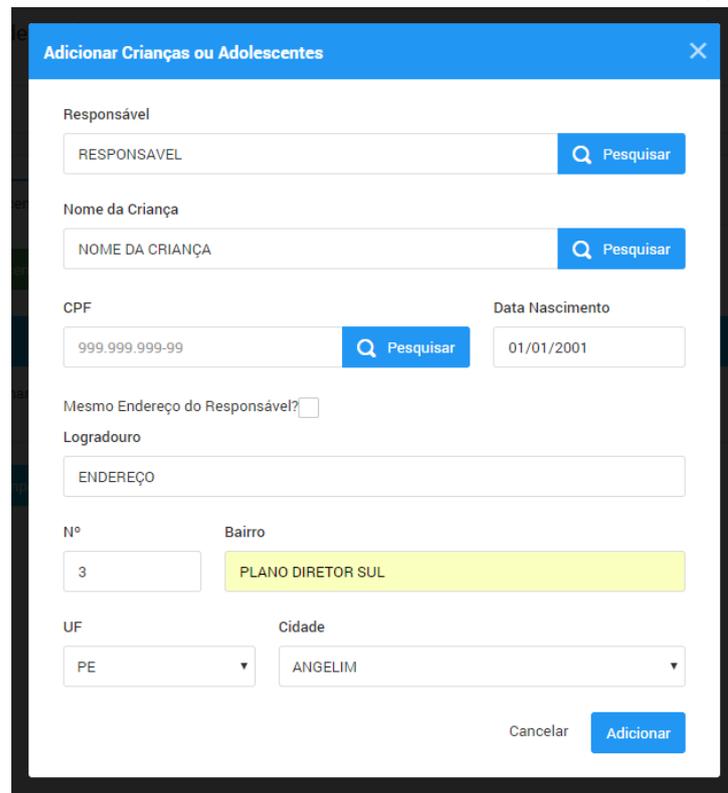
Buscar Criança/Adolescente por Nome

Nome: MAGA

Q Pesqui

ID	CPF	Nome	Dt. Nasc.	Responsável
1	022.642.731-51	RODRIGO MAGALHÃES DIAS DA SILVA	19/07/1993	BEATRIZ MAGALHÃES PEREIRA
2		miguel magalhaes	08/12/2015	BEATRIZ MAGALHÃES PEREIRA
3		MIGUEL MAGALHAES	08/12/2015	BEATRIZ MAGALHÃES PEREIRA

Figura 40 – Tela onde é possível cadastrar uma nova criança ou adolescente



**Adicionar Crianças ou Adolescentes**

Responsável  
 RESPONSÁVEL

Nome da Criança  
 NOME DA CRIANÇA

CPF   Data Nascimento

Mesmo Endereço do Responsável?

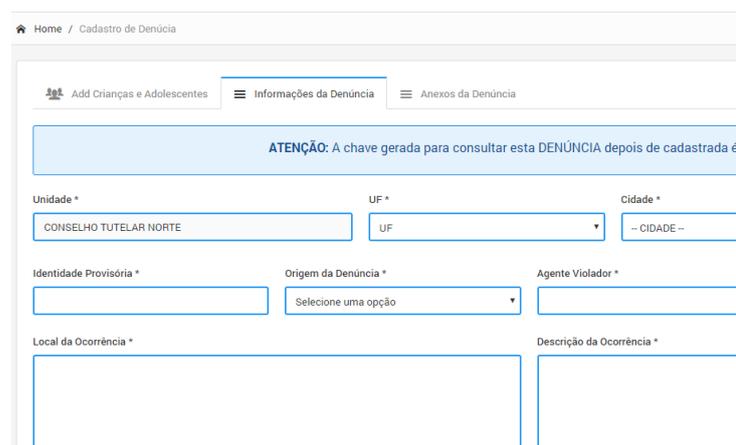
Logradouro  
 ENDEREÇO

Nº  Bairro

UF  Cidade

Figura 41 – Formulário a ser preenchido com os dados da denúncia

Denúncia - Cadastro de Denúcia



Home / Cadastro de Denúcia

**ATENÇÃO:** A chave gerada para consultar esta DENÚNCIA depois de cadastrada é

Unidade \*  UF \*  Cidade \*

Identidade Provisória \*  Origem da Denúncia \*  Agente Violador \*

Local da Ocorrência \*  Descrição da Ocorrência \*

## APÊNDICE F - TAREFA 2: CADASTRO DE CRIANÇAS E ADOLESCENTES

**Figura 42 – Formulário a ser preenchido com os dados da Criança / Adolescente**

🏠 Cadastro - Criança e Adolescente Nº Pz

---

🏠 Home / Cadastro de Criança e Adolescente

☰ Dados Criança e Adolescente   📍 Endereços   📄 Outras Informações

Unidade: CONSELHO TUTELAR NORTE   Responsável: RESPONSÁVEL POR JOÃO SILVA 🔍 Pesquisar

NIS/PIS/PASEP:  🔍 Pesquisar   Nome: CRIANÇA JOAO SILVA 🔍 Pesquisar   CPF: 999.999.999-99 🔍 Pesquisar   D. Cadastrar Responsável: 05/06/2007

Filiação Pai: PAI DO JOAO   Filiação Mãe: MAE DO JOAO

Sexo: Masculino   Certidão Nascimento:    Cor ou Raça: Selecione uma opção

RG:    Orgão Expeditor:    Data Expedição:

**Figura 43 – Formulário a ser preenchido com os dados de endereço da Criança / Adolescente. É possível cadastrar um histórico de endereços**

☰ Dados Criança e Adolescente   📍 Endereços   📄 Outras Informações

**INFO:** Preencha os campos com os dados corretamente e clique no botão de Adicionar Endereço.

Logradouro:    Nº:    Bairro:

UF: UF   Cidade: -- CIDADE --   CEP:

Complemento:    Referência:

✓ Adicionar
✎ Alterar
🗑 Excluir
🔄 Limpar

---

Endereço

⚪ Nenhum Endereço adicionado

## APÊNDICE G - TAREFA 3: REGISTRO DE ATENDIMENTO

Figura 44 – Formulário a ser preenchido com os dados do Atendimento.

DataGuardian - Atendimento

Home / Cadastro de Atendimento

Dados da Denúncia
  Crianças e Adolescentes
  Direitos Violados
  Relatos e Aplicação de Medidas Protetivas
  Anexos

Unidade  
CONSELHO TUTELAR NORTE

Denúncia que Gerou este ATENDIMENTO

Origem Denúncia 
 Agente Violador 
 Data Ocorrência

Narrativa da Ocorrência

CONSELHEIRO 1 NORTE
  CONSELHEIRO 2 NORTE
  CONSELHEIRO 3
  CONSELHEIRO 4 NORTE

Figura 45 – Caixa de diálogo onde o usuário seleciona os direitos que foram ameaçados ou violados.

Selecionar os Direitos Violados

<input type="checkbox"/>	D.V	CATEGORIA	DIREITO FUNDAMENTAL
<input checked="" type="checkbox"/>	FALTA DE ATENDIMENTO PERI E PRÉ-NATAL	NÃO ATENDIMENTO MÉDICO	DIREITO À VIDA E À SAÚDE
<input type="checkbox"/>	FALTA DE ATENDIMENTO EMERGENCIAL	NÃO ATENDIMENTO MÉDICO	DIREITO À VIDA E À SAÚDE
<input checked="" type="checkbox"/>	FALTA DE ATENDIMENTO ESPECIALIZADO	NÃO ATENDIMENTO MÉDICO	DIREITO À VIDA E À SAÚDE
<input type="checkbox"/>	FALTA DE ACOMPANHAMENTO MÉDICO DE ROTINA	NÃO ATENDIMENTO MÉDICO	DIREITO À VIDA E À SAÚDE
<input type="checkbox"/>	FALTA DE ACOMP. ODONTOLÓGICO DE ROTINA	NÃO ATENDIMENTO MÉDICO	DIREITO À VIDA E À SAÚDE
<input type="checkbox"/>	FALTA DE EQUIPAMENTOS	NÃO ATENDIMENTO MÉDICO	DIREITO À VIDA E À SAÚDE
<input type="checkbox"/>	FALTA DE VACINAÇÃO	NÃO ATENDIMENTO MÉDICO	DIREITO À VIDA E À SAÚDE
<input type="checkbox"/>	RECUSA DE ATENDIMENTO	NÃO ATENDIMENTO MÉDICO	DIREITO À VIDA E À SAÚDE
<input type="checkbox"/>	FALTA DE LEITOS PARA INTERNAÇÃO HOSPITALAR	NÃO ATENDIMENTO MÉDICO	DIREITO À VIDA E À SAÚDE

Figura 46 – Formulário para entrada das narrativas dos envolvidos no caso.

☰ Dados da Denúncia   Crianças e Adolescentes   Direitos Violados   Relatos e Aplicação de Medidas Protetivas   Anexos

INFO! Aqui é descrito os relatos tanto do Acusado como da Vítima.

Data Notificação \*   Data Comparecimento \*   Data Atendimento \*

10/05/2018 00:00   14/05/2018 08:37   01/05/2018 15:53

Narrativa do Agente Violador

Lorem ipsum hendrerit vivamus tincidunt blandit, dapibus bibendum elit varius, aenean hendrerit aliquam sociosqu. pulvinar sociosqu est tellus ligula iaculis vivam lobortis molestie ut nam sodales felis, cursus semper consectetur venenatis mauris at

Narrativa da Vítima

ttis vivamus sodales, elementum purus vulputate hac morbi. sapien augue torquent primis torquent aliquet iaculis, facilisis mauris pharetra donec vestibulum, fusc

Decisão Preliminar do Conselho

Vel nam inceptos curabitur adipiscing tortor duis, condimentum vestibulum potenti etiam cursus porttitor auctor, cubilia tempus faucibus luctus eleifend. suscipit f sed ultrices aliquet mollis primis vel, at accumsan class himenaeos vehicula vestibulum, porttitor curae odio rutrum massa primis varius aenean eros eget sodales, curabitur adipiscing tortor duis, condimentum vestibulum potenti etiam cursus porttitor auctor, cubilia tempus faucibus luctus eleifend. suscipit felis netus cursus aliquet mollis primis vel, at accumsan class himenaeos vehicula vestibulum, porttitor curae odio rutrum massa primis varius aenean eros eget sodales, sagittis vel

Figura 47 – Caixa de diálogo para seleção das medidas protetivas a serem aplicadas.

**Selecionar Medidas Protetivas** ✕

<input type="checkbox"/>	Descrição	Artigo
<input checked="" type="checkbox"/>	ENCAMINHAMENTO AOS PAIS MEDIANTE TERMO DE RESPONSABILIDADE	ARTIGO 101
<input type="checkbox"/>	ORIENTAÇÃO, APOIO E A ACOMPANHAMENTO TEMPORÁRIO	ARTIGO 101
<input checked="" type="checkbox"/>	MATRÍCULA EM ESTABELECIMENTO OFICIAL DE ENSINO	ARTIGO 101
<input checked="" type="checkbox"/>	FREQUÊNCIA OBRIGATÓRIA EM ESTABELECIMENTO DE ENSINO	ARTIGO 101
<input type="checkbox"/>	INCLUSÃO EM PROGRAMAS OFICIAL DE AUXÍLIO À CRIANÇA E AO ADOLESCENTE	ARTIGO 101
<input type="checkbox"/>	REQUISIÇÃO DE TRATAMENTO MÉDICO HOSPITALAR OU AMBULATORIAL	ARTIGO 101
<input type="checkbox"/>	REQUISIÇÃO DE TRATAMENTO PSICOLÓGICO EM REGIME HOSPITALAR OU AMBULATORIAL	ARTIGO 101
<input type="checkbox"/>	REQUISIÇÃO DE TRATAMENTO PSIQUIÁTRICO EM REGIME HOSPITALAR OU AMBULATORIAL	ARTIGO 101
<input type="checkbox"/>	INCLUSÃO EM PROGRAMA DE AUXÍLIO, ORIENTAÇÃO E TRATAMENTO A ALCOÓLATRAS E TOXICÔMANOS	ARTIGO 101
<input type="checkbox"/>	ACOLHIMENTO INSTITUCIONAL (ABRIGO)	ARTIGO 101

1 < 2 > >>

🔍 Pronto

## APÊNDICE H - TAREFA 4: CADASTRO DE PASTAS ANTIGAS

Figura 48 – Formulário a ser preenchido com os dados da pasta

**1º Informações do CASO**

Unidade Conselheira \* CONSELHO TUTELAR NORTE  
 Conselheiro Responsável \* CONSELHEIRO 3  
 Nº Pasta \* 222/2017

**2º Adicionar Responsável ao CASO**

**INFORMAÇÕES DO RESPONSÁVEL OU GUARDIÃO**

Pais/Responsável/Guardião   CPF

Data Nascimento  Sexo Masculino Estado Civil Selezione uma opção Nº Filhos 0

Cônjuge  Telefone  Telefone

Figura 49 – Adicionar crianças e adolescentes à pasta

**3º Adicionar Crianças e Adolescentes ao CASO**

**CRIANÇAS OU ADOLESCENTES DA PASTA**

Tabelas de Crianças Adicionadas à Pasta

#	Nome	Dt. Nasc.	Responsável	Nº Caso
Nenhuma Criança / Adolescente Adicionada a PASTA				

Figura 50 – Formulário para o cadastro das crianças

Telefone

### Adicionar Crianças ou Adolescentes

Nome da Criança

REBECA FABIANA ESTHER Pesquisar

CPF

999.999.999-99 Pesquisar

Data Nascimento

09/09/2006

Logradouro

RUA PRÍMULAS

Nº

106

Bairro

REDINHA

Complemento

NATAL

UF

TO

Cidade

PALMAS

Cancelar Adicionar

Adolescente Adicionada a PASTA

Figura 51 – Tabela de Anexos inseridos na pasta.

4º Adicionar Adicionar Anexos ao CASO

ADICIONAR ANEXOS À PASTA

+ Anexar Arquivos Enviar Arquivo Cancelar

Lista de Anexos

Descrição	URL	BAIXAR
Guia-de-impressao.pdf	arq1525317270142.pdf	<span>Download</span>

Atualizar Pasta Limpar Formulário

**APÊNDICE I - PERFIL DOS AVALIADORES PARTICIPANTES DA  
AVALIAÇÃO HEURÍSTICA**

Perfil dos Avaliadores					
<b>X</b>	<b>Formação</b>	<b>Função</b>	<b>Local de Trabalho</b>	<b>Sexo</b>	<b>Idade</b>
Avaliador 1	Bacharel em Ciência da Computação	Desenvolvedor Web	Tecnotins Sistemas	M	35
Avaliador 2	Doutor em Ciência da Computação	Professor e Pesquisador	Universidade Federal do Tocantins	M	37

## APÊNDICE J - PERFIL DOS USUÁRIOS PARTICIPANTES DOS TESTES DE USABILIDADE

PARTICIPANTE	IDADE	SEXO	FUNÇÃO	UC
USUÁRIO 1	29	F	CONSELHEIRO(A)	UCRSI
USUÁRIO 2	36	F	CONSELHEIRO(A)	UCRSI
USUÁRIO 3	42	M	CONSELHEIRO(A)	UCRSI
USUÁRIO 4	38	F	CONSELHEIRO(A)	UCRSI
USUÁRIO 5	52	F	CONSELHEIRO(A)	UCRSI
USUÁRIO 6	46	F	CONSELHEIRO(A)	UCRSII
USUÁRIO 7	40	F	CONSELHEIRO(A)	UCRSII
USUÁRIO 8	47	F	CONSELHEIRO(A)	UCRSII
USUÁRIO 9	40	M	CONSELHEIRO(A)	UCRSII
USUÁRIO 10	28	M	CONSELHEIRO(A)	UCRSII
USUÁRIO 11	42	F	ADMINISTRATIVO	UCRSII
USUÁRIO 12	23	F	ADMINISTRATIVO	UCRSII
USUÁRIO 13	31	M	CONSELHEIRO(A)	UCRC
USUÁRIO 14	39	F	CONSELHEIRO(A)	UCRC
USUÁRIO 15	34	F	CONSELHEIRO(A)	UCRC
USUÁRIO 16	30	F	CONSELHEIRO(A)	UCRC
USUÁRIO 17	47	F	CONSELHEIRO(A)	UCRC
USUÁRIO 18	29	F	ADMINISTRATIVO	UCRC
USUARIO 19	40	F	ADMINISTRATIVO	UCRC
USUÁRIO 20	29	M	CONSELHEIRO(A)	UCRN
USUÁRIO 21	40	M	CONSELHEIRO(A)	UCRN
USUÁRIO 22	57	F	CONSELHEIRO(A)	UCRN
USUÁRIO 23	53	F	CONSELHEIRO(A)	UCRN
USUÁRIO 24	37	F	CONSELHEIRO(A)	UCRN
USUÁRIO 25	40	F	ADMINISTRATIVO	UCRN

**Legenda:**

UC: Unidade Conselheira

UCRSI: Unidade Conselheira Região Sul 1

UCRSII: Unidade Conselheira Região Sul 2

UCRC: Unidade Conselheira Região Central

UCRN: Unidade Conselheira Região Norte

**APÊNDICE K - COMPILAÇÃO ANÁLISE HEURÍSTICA**

## Compilação da Análise Heurística, segundo os Princípios de Nielsen

### Glossário

**Localidade:** **1** - em um único local na interface; **2** - em dois ou mais locais na interface; **3** - na estrutura geral da interface; **4** - inexistente, deve ser incluído.

**Gravidade:** **0** - Não é encarado necessariamente como um problema de usabilidade; **1** - Não necessita ser corrigido, a menos que haja tempo disponível; **2** - Baixa prioridade; **3** - Alta prioridade; **4** - Catastrófico; deve ser corrigido imediatamente.

Heurística	Localidade	Gravidade	Problemas
<b>1. Visibilidade do status do sistema</b>	2	3	Ficou um pouco confuso o acesso ao sistema. Foi-me informada uma senha temporária, porém fiquei confuso sobre qual usuário utilizar, utilizei meu próprio nome. Ao fazer login, fui direcionado para uma tela de alteração de senha, já que a senha era provisória, porém, quando tentei alterar a senha, sempre emitiu uma mensagem de erro informando que a senha atual estava incorreta, e não estava. Na Área de Trabalho, como o próprio nome sugere, eu imaginava que existiria uma série de ações disponíveis para que eu pudesse executá-las, o que não foi verdade. Lá tem mais uma espécie de Dashboard com dados de ações previamente executadas.
<b>2. Compatibilidade do sistema com o mundo real</b>			Não tenho parâmetros para avaliar.
<b>3. Controle do usuário e liberdade</b>	1	3	Permite o usuário navegar nas abas de uma determinada funcionalidade sem perder os dados no formulário, acho esta característica interessante. Se existe um menu Cadastro, por que existe uma opção de cadastro dentro do menu Atividades Extras? Muito confuso.
<b>4. Consistência e padrões</b>	2	3	São utilizados padrões que facilitam o entendimento do sistema. Fiquei um pouco confuso na parte de cadastro porque aparecem opções para cadastro de Criança ou Adolescente, depois a opção de Responsáveis, então surgiu a pergunta, Responsáveis de que? Responsável pela criança? Pelo órgão? Se for pela criança, teria que colocar esta opção acessível a partir da opção de Cadastro de criança, caso seja do órgão, teria que explicitar na opção correspondente. Nos formulários, talvez fosse interessante ter opção para Cancelar, que fechava a tela e voltaria para a tela principal. Além disso, em relação aos padrões, é recomendável utilizar cores diferentes, por exemplo, distinguindo as opções que concluem uma ação (Salva) e uma opção que limpa o formulário.
<b>5. Ajuda os usuários a reconhecer, diagnosticar e recupera-se de erros</b>	2	3	Não localizei a opção de ajuda. As mensagens de erro são exibidas no canto superior direito, não é uma prática recomendada de HCI, se colocar no topo central ou no meio da tela dará maior visibilidade ao usuário.
<b>6. Prevenção de erros</b>	2	3	O sistema possui um tratamento de erro e informa ao usuário, porém quando existe mais de um erro em um formulário de cadastro, as mensagens são exibidas todas de uma vez e logo em seguida desaparecem. É impossível verificar todos os erros que ocorreram.
<b>7. Reconhecimento em vez de memorização</b>	2	2	As ações e as disposições destas são bem coerentes à proposta do projeto.
<b>8. Flexibilidade e eficiência de uso</b>	2	2	O sistema é simples de ser utilizado, embora apresente confusão na disposição de algumas funcionalidades, por exemplo, repetindo opções de cadastro em diferentes localidades no menu. Talvez a estrutura organizacional do menu devesse ser repensada.
<b>9. Estética e design minimalista</b>	2	2	O design é bem feito, utiliza componentes visuais oportunos, compacta formulários grandes em pequenas partes, porém poderia melhorar modificando as cores de algumas ações, por exemplo, opções para Cancelar ou limpar o Formulário.
<b>10. Ajuda e documentação</b>	2	3	Não consegui localizar nenhuma opção correspondente.

Heurística	Localidade	Gravidade	Problemas
<b>1. Visibilidade do status do sistema</b>	2	0	A visibilidade do sistema está muito bem implementada. As funcionalidades são fáceis de interpretar através dos ícones. O feedback é bem posicionado. Em todas as telas, o usuário sabe em qual funcionalidade se encontra. Em alguns casos, alguns botões poderiam possuir ícone ao lado dos rótulos para maior compreensão.
<b>2. Compatibilidade do sistema com o mundo real</b>			O sistema possui uma linguagem semelhante à usada nos Conselhos Tutelares, como "caso", "denúncia", "pasta" e "direitos violados".
<b>3. Controle do usuário e liberdade</b>	2	2	O sistema oferece condições ao usuário de salvar e alterar informações. O sistema não apresentou mensagem de erro ou trava na tentativa de voltar a tela anterior ou qualquer outra tela disponível, porém as informações são perdidas quando realizado, tendo o usuário ter que refazer o que tinha feito. Na tela de registro de denúncia, o usuário é guiado pelo próprio software para realizar a tarefa, tendo como opções apenas o caminho que o sistema oferece. Entretanto, foi analisado como uma forma de evitar que o usuário mais leigo faça algo indesejável. O sistema não permite alteração de nome de arquivos em anexo.
<b>4. Consistência e padrões</b>	3	2	O sistema apresentou ótima consistência. Todas as telas possuem o mesmo menu de navegação na vertical. O sistema possui de apenas uma paleta de cores, sendo muito bem explorada. Todas as telas possuem títulos na mesma posição, tamanho e cor de fonte. Os botões são iguais mudando apenas rótulos e ícones, com exceção de botões do tipo "Excluir" que possui cor vermelha com intenção de alertar ao usuário. Uma ressalva, em alguns campos de formulário a pesquisa se houve por campos de auto completar, e em outros (a maioria) possui botões de pesquisa ao lado do campo para realizar esta tarefa. Talvez fosse mais viável, escolher entre um ou outro.
<b>5. Ajuda os usuários a reconhecer, diagnosticar e recupera-se de erros</b>	3	2	O sistema exibe mensagem de erro como caixa de diálogo fluando no canto superior direito da tela e também em cima de formulários quando mal preenchidos, permitindo ao usuário reconhecer que realizou algo que não deveria ou de forma indevida. Campos obrigatórios são informados ao usuário quando não preenchidos. Uma ressalva constatada, observou-se em mensagens de erro que não foram claras, deixando o usuário sem saber o que realmente ocorreu.
<b>6. Prevenção de erros</b>	3	2	Para tentar prevenir o usuário de errar, o sistema desabilitou botões ou deixou de redimensionar alguns componentes. Por exemplo, o sistema não permite cadastrar a mesma pessoa mais de uma vez. Ao buscar uma pessoa no banco de dados ele desabilita a opção de salvar, e permite apenas que ela seja alterada. Dependendo do controle de acesso do usuário, alguns componentes não são visíveis para ele, impedindo-o assim de realizar tarefas que não deveria.
<b>7. Reconhecimento em vez de memorização</b>			As telas apresentam o caminho percorrido para chegar até ali, não obrigando o usuário a decorá-lo. É fácil identificar o que está sendo feito. A ordem dos itens de menu segue uma lógica semelhante às etapas seguidas no ambiente real dos Conselhos Tutelares. Por exemplo, para cadastrar uma criança ou adolescente, é necessário o cadastro do responsável, portanto, a opção de cadastrar o responsável vem antes do da criança ou adolescente.
<b>8. Flexibilidade e eficiência de uso</b>			O sistema é responsivo, se adaptando aos diferentes tamanhos de tela. O usuário pode navegar entre campos de formulário utilizando a tecla "tab" do teclado, e fechar caixas de diálogo com a tecla "esc". Registro de responsáveis e crianças podem ser realizadas de várias formas e em diversos lugares. Ao cadastrar uma criança ou adolescente, o sistema permite com um simples "clique" a cópia dos mesmo dados de endereço do seu tutor, tornando a tarefa mais simples e rápida.
<b>9. Estética e design minimalista</b>			Os formulários apresentaram apenas informações realmente importantes. Formulários que contém muitos campos, foram divididos em abas, onde os rótulos são bem descritivos e com ícones para representá-los. O design das telas são bem limpos e organizados. O espaçamento entrelinhas, entre os campos de formulário e entre os itens de menu tornava a leitura mais agradável. Foi avaliado como um sistema bonito e elegante.
<b>10. Ajuda e documentação</b>	4	3	O sistema não possui manual de ajuda ou documentação. Apesar do sistema ser muito claro em suas funções, não há um link por exemplo para recorrer a um manual caso necessário.

## ANEXO A - QUESTIONÁRIO DE TESTE DE USABILIDADE - SUS

Figura 52 – Questionário SUS

Participante: \_\_\_\_\_

Data: \_\_\_/\_\_\_/\_\_\_

## Questionário de Avaliação de Usabilidade - System Usability Scale

**Instruções:** Para cada uma das seguintes declarações, marque a opção que melhor descreve suas reações ao sistema hoje.

		Discordo Totalmente	Discordo	Neutro	Concordo	Concordo Totalmente
1.	Acho que gostaria de usar este sistema com frequência.	<input type="checkbox"/>				
2.	Achei o sistema desnecessariamente complexo.	<input type="checkbox"/>				
3.	Achei o sistema fácil de usar.	<input type="checkbox"/>				
4.	Achei que seria necessário o apoio de um técnico para poder usar este sistema.	<input type="checkbox"/>				
5.	As funções deste sistema estavam bem integradas.	<input type="checkbox"/>				
6.	Achei este sistema muito inconsistente.	<input type="checkbox"/>				
7.	Eu imagino que as pessoas aprenderão como usar esse sistema rapidamente.	<input type="checkbox"/>				
8.	Eu achei o sistema muito confuso e me atrapalhei ao usá-lo.	<input type="checkbox"/>				
9.	Eu me senti confiante ao usar o sistema.	<input type="checkbox"/>				
10.	Eu precisei aprender várias coisas novas antes de conseguir usar o sistema.	<input type="checkbox"/>				

Por favor, forneça quaisquer comentários sobre este sistema:

## ANEXO B - TEMPLATE DO QUESTIONÁRIO PARA AVALIAÇÃO HEURÍSTICA DE USABILIDADE

### Glossário

**Localidade:** 1- em um único local na interface; 2 – em dois ou mais locais na interface; 3 – na estrutura geral da interface; 4 – inexistente, deve ser incluído.

**Gravidade:** 0 – Não é encarado necessariamente como um problema de usabilidade; 1 – Não necessita ser corrigido, a menos que haja tempo disponível; 2 – Baixa prioridade; 3 – Alta prioridade; 4 – Catastrófico; deve ser corrigido imediatamente.

Heurística	Localidade	Gravidade	Problemas
1. Visibilidade do status do sistema			
2. Compatibilidade do sistema com o mundo real			
3. Controle do usuário e liberdade			
4. Consistência e padrões			
5. Ajuda os usuários a reconhecer, diagnosticar e recupera-se de erros			
6. Prevenção de erros			
7. Reconhecimento em vez de memorização			
8. Flexibilidade e eficiência de uso			
9. Estética e design minimalista			
10. Ajuda e documentação			

**ANEXO C - CHECKLIST PARA AVALIAÇÃO HERUÍSTICA DE  
USABILIDADE DO SISTEMA DATAGUARDIAN**

## CHECKLIST PARA AVALIAÇÃO HEURÍSTICA DE USABILIDADE DO SISTEMA DATAGUARDIAN

Data:	Avaliador:			
Heurística	Questões	Não	Sim	NA <sup>1</sup>
Visibilidade do status do sistema	1. Para cada ação do usuário o aplicativo oferece feedback imediato e adequado sobre seu status? <i>Por exemplo, após tarefas como registro de denúncia, exibir uma mensagem de confirmação do tipo "Denúncia Cadastrada".</i>			
	2. Os componentes interativos selecionados são claramente distintos dos demais? <i>Por exemplo, o estado de botões muda quando são pressionados e destaca a aba do menu que está sendo visualizada.</i>			
	3. As mensagens sobre o status do aplicativo possuem uma linguagem clara e concisa? <i>Por exemplo, os títulos das telas e das mensagens de erro são de fácil compreensão.</i>			
	4. Fornece um <i>update</i> do status para operações mais lentas? <i>Por exemplo, uma indicação seja na forma de ícone ou texto sobre o progresso do carregamento do sistema ou de um arquivo.</i>			
Compatibilidade do sistema com o mundo real	5. O significado de símbolos e ícones são compreensíveis e intuitivos? <i>Utilizar ícones e símbolos fáceis de reconhecer e relacionar com a tarefa à qual estão associados.</i>			
	6. As informações são dispostas em uma ordem lógica e natural? <i>Por exemplo, itens em listas de seleção (nomes, produtos, etc.) são ordenados por um critério adequado (p.ex. alfabeticamente).</i>			
Controle do usuário e liberdade	7. É o usuário quem inicia e encerra tarefas e não o sistema? <i>Por exemplo, aguardar o usuário teclar enter após preencher o campo de busca para iniciar a tarefa.</i>			
	8. É possível retornar a tela anterior a qualquer momento? <i>Seja a partir da navegação por abas, de um botão voltar ou outra opção.</i>			
	9. É possível sair da tela atual a qualquer momento?			
	10. O usuário pode cancelar uma ação em progresso?			
Consistência e padrões	11. As telas com o mesmo tipo de conteúdo possuem o mesmo título? <i>Por exemplo, todas as telas de busca possuem o mesmo título.</i>			
	12. Controles e botões se distinguem do restante do layout, deixando claro que são clicáveis?			
	13. Todas as informações textuais do sistema utilizam o mesmo idioma?			
	14. Funções diferentes são apresentadas de maneira distinta ao usuário? <i>Por exemplo, funções como salvar e cancelar não são representadas pelo mesmo ícone ou nome.</i>			
	15. Funções semelhantes são apresentadas de forma semelhante? <i>Por exemplo, uso de mesmo ícone ou rótulo para botões cuja funcionalidade é a mesma, ou propõe a mesma forma de entrada de dados para uma mesma funcionalidade em telas diferentes.</i>			
	16. A forma de navegação é consistente entre as telas no aplicativo? <i>Mantém o mesmo tipo de navegação (rolagem vertical, rolagem horizontal, menus ou abas) em todas as telas.</i>			

	17. Os links são tratados de forma consistente entre as telas? <i>Mantém o mesmo tratamento visual em termos de cor, tipo e estilo (p.ex. negrito, sublinhado, cor azul) de fonte.</i>			
	18. As informações textuais são apresentadas de forma padronizada? <i>Apresenta informações textuais semelhantes na mesma disposição e com o mesmo tratamento visual (tamanho, tipo, cor).</i>			
	19. Em campos onde há necessidade de inserção de dados, isso é evidente? <i>Por exemplo, ter uma caixa de texto com um cursos.</i>			
Prevenção de erros	20. O sistema impede o usuário de realizar tarefas que não deveriam fazer? <i>Por exemplo, desabilitar um botão para que não seja clicado, ou não renderizar uma funcionalidade que não deveria ser acessada.</i>			
	21. Se o usuário errar ao realizar uma tarefa, a mesma pode ser desfeita ou o sistema acusa o erro antes mesmo da tarefa ser realizada?			
Ajuda os usuários a reconhecer, diagnosticar e recupera-se de erros	22. As mensagens de erro são fácil de identificar? <i>Por exemplo, mensagens de erro em tons de cor vermelho para demonstrar que algo de errado ocorreu.</i>			
	23. As mensagens de erro estão bem localizadas na tela?			
Reconhecimento em vez de memorização	24. O sistema utiliza em seus textos e rótulos, uma linguagem habitual e conhecida pelo usuário do aplicativo?			
	25. Os títulos das telas descrevem adequadamente seu conteúdo?			
	26. O usuário sabe identificar facilmente a funcionalidade que está sendo acessada?			
Flexibilidade e eficiência de uso	27. O sistema funciona corretamente, sem apresentar problemas durante a interação? <i>Por exemplo, travar ou ter botões que não funcionam no primeiro clique.</i>			
	28. As tarefas são relativamente simples de serem executadas? <i>Por exemplo, uma tarefa pode ser concluída em poucos passos.</i>			
	29. As funções mais utilizadas são facilmente acessadas? <i>Oferecer aos usuários caminhos alternativos para acessarem a mesma tarefa.</i>			
	30. O sistema utiliza-se de ícones ao invés do seu significado? <i>Por exemplo, utilizar um ícone de impressora para imprimir ao invés de utilizar a palavra impressora.</i>			
	31. Todas as telas mantêm acessíveis menus e funções comuns do sistema?			
Estética e design minimalista	32. São exibidas apenas informações relacionadas a tarefa que está sendo realizada?			
	33. São usados textos somente quando estes são realmente indispensáveis?			
	34. O menu é esteticamente simples e claro? <i>Com opções fáceis de encontrar, dispostas em uma ordem lógica e com títulos curtos.</i>			
	35. O sistema exibe quantidades pequenas/adequadas de informações em cada tela?			
	36. Os títulos de telas/janelas e rótulos de botões/links são curtos e claros?			
	37. O sistema possui uma opção de ajuda?			

Ajuda e documentação	38. É fácil localizar a ajuda?			
	39. Ajuda é de fácil compreensão?			
Legibilidade layout	40. O espaçamento entrelinhas favorece a leitura?			
	41. As fontes utilizadas favorecem a leitura?			
	42. É fácil identificar as funções dos ícones?			
	43. O sistema realça conteúdos que possuem maior importância?			
	44. O alinhamento utilizado favorece a leitura?			