

**FAI – CENTRO DE ENSINO SUPERIOR EM GESTÃO,
TECNOLOGIA E EDUCAÇÃO
CURSO DE SISTEMAS DE INFORMAÇÃO**

**ANDRÉ SILVA LOPES
DANIEL FERNANDO PEREIRA
TAÍSES DA SILVA MENDES**

**HELP A PET – SISTEMA DE SOFTWARE PARA AUXÍLIO À CAUSA
DA PROTEÇÃO ANIMAL**

SANTA RITA DO SAPUCAÍ – MG

2019

**FAI – CENTRO DE ENSINO SUPERIOR EM GESTÃO,
TECNOLOGIA E EDUCAÇÃO
CURSO DE SISTEMAS DE INFORMAÇÃO**

**ANDRÉ SILVA LOPES
DANIEL FERNANDO PEREIRA
TAÍSES DA SILVA MENDES**

**HELP A PET – SISTEMA DE SOFTWARE PARA AUXÍLIO À CAUSA
DA PROTEÇÃO ANIMAL**

Projeto de final de curso apresentado na FAI – Centro de Ensino Superior em Gestão, Tecnologia e Educação, como requisito parcial para obtenção do Título de Bacharel em Sistemas de Informação, sob a orientação da profa. Eunice Gomes de Siqueira.

SANTA RITA DO SAPUCAÍ – MG

2019

LISTA DE REVISÕES

Continua.

Data	Versão	Autor(es)	Descrição
11/12/2019	4.7	André da Silva Lopes Daniel Fernando Pereira Taíses da Silva Mendes	Ajustes finais.
02/12/2019	4.6	Eunice G. Siqueira	Ajustes na formatação.
02/12/2019	4.5	André da Silva Lopes Daniel Fernando Pereira Taíses da Silva Mendes	Correção da Fase 4.
24/11/2019	4.4	André da Silva Lopes Daniel Fernando Pereira Taíses da Silva Mendes	Correção da Fase 4.
20/11/2019	4.3	Eunice G. Siqueira	Apontamentos na versão.
16/11/2019	4.2	André da Silva Lopes Daniel Fernando Pereira Taíses da Silva Mendes	Correção da Fase 4.
11/11/2019	4.1	Eunice G. Siqueira	Correção parcial da Fase 4.
03/11/2019	4.0	André da Silva Lopes Daniel Fernando Pereira Taíses da Silva Mendes	Correção parcial da Fase 3 e Fase 4.
09/10/2019	3.2	Eunice G. Siqueira	Correção da Fase 3.
07/09/2019	3.1	André da Silva Lopes	Inclusão das seções da Fase 3.
29/06/2019	2.2	André da Silva Lopes	Correção parcial da Fase 2.
10/06/2019	2.1	Eunice G. Siqueira	Correção parcial da Fase 2.

Data	Versão	Autor(es)	Descrição
31/05/2019	2.0	André da Silva Lopes Lucas Soares de Oliveira	Formatação e revisão do documento.
22/05/2019	1.8	André da Silva Lopes Lucas Soares de Oliveira Taíses da Silva Mendes	Seção Interação Humano-Computador.
19/05/2019	1.7	Daniel Fernando Pereira Taíses da Silva Mendes	Construção dos modelos da visão de dados.
18/05/2019	1.6	Daniel Fernando Pereira Taíses da Silva Mendes	Construção dos diagramas da visão comportamental e implementação dos casos de uso.
15/05/2019	1.5	Eunice G. Siqueira	Correção parcial da Fase 1.
12/05/2019	1.4	Daniel Fernando Pereira Taíses da Silva Mendes	Construção dos diagramas da visão estrutural e implementação dos casos de uso.
12/05/2019	1.3	André Silva Lopes Lucas Soares de Oliveira	Atualização do Plano de Projeto e seus artefatos para Fase 2.
10/05/2019	1.2	André Silva Lopes Daniel Fernando Pereira Lucas Soares de Oliveira Taíses da Silva Mendes	Aplicação das Métricas para Estimativa do Esforço.
14/04/2019	1.1	André Silva Lopes Daniel Fernando Pereira Lucas Soares de Oliveira Taíses da Silva Mendes	Revisão e correção da Fase 1.
01/04/2019	0.9	André Silva Lopes Daniel Fernando Pereira Taíses da Silva Mendes	Criação do Modelo de Casos de Uso.
28/03/2019	0.8	Daniel Fernando Pereira Lucas Soares de Oliveira	Criação do Modelo Conceitual de Dados.
23/03/2019	0.7	Daniel Fernando Pereira Taíses da Silva Mendes	Descrição dos Requisitos Funcionais.

Data	Versão	Autor(es)	Descrição
16/03/2019	0.6	Daniel Fernando Pereira Taíses da Silva Mendes	Descrição dos Requisitos não Funcionais.
15/03/2019	0.5	André Silva Lopes Lucas Soares de Oliveira	Definições das Áreas de Conhecimento.
13/03/2019	0.4	Lucas Soares de Oliveira	Criação do Plano de Projeto.
08/03/2019	0.3	Daniel Fernando Pereira André Silva Lopes Taíses da Silva Mendes	Objetivos do Projeto.
04/03/2019	0.2	Taíses da Silva Mendes André Silva Lopes	Descrição da Revisão Bibliográfica.
26/02/2019	0.1	Lucas Soares de Oliveira	Criação do documento.

RESUMO

A educação da população quanto aos cuidados dos animais e de sua posse e guarda responsável são elementos essenciais para o convívio em sociedade. O combate aos maus-tratos infligidos aos animais e a melhoria das condições de vida daqueles que estão abandonados nas ruas devem ser medidas protetoras adotadas por todos os seres humanos. Neste documento, encontra-se o detalhamento do Projeto *Help a Pet* que teve como objetivo principal desenvolver um sistema de software que possibilite às entidades protetoras de animais divulgarem seus serviços. Por meio desse sistema, as associações e organizações não governamentais poderão divulgar suas ações, apresentar os animais resgatados e receber pedidos de adoção feitos pelas pessoas interessadas. Apresenta-se o processo de desenvolvimento do projeto, contemplando os estudos e entrevistas realizadas, o gerenciamento, a modelagem do sistema de software, a codificação e registros de testes realizados.

Palavras-chave: Proteção animal. Organizações não governamentais. Sociedade. Sistemas de Informação.

LISTA DE FIGURAS

FIGURA 1: Foto do animal submetido aos maus-tratos em Osasco-SP	19
FIGURA 2: Campanha da Delegacia Eletrônica de Proteção Animal	20
FIGURA 3: Saldo financeiro referentes ao mês de março 2019	23
FIGURA 4: Lagarto ferido resgatado pela ONG Resgacti	23
FIGURA 5: Campanha do combate ao abandono de animais	24
FIGURA 6: Grupo de voluntários em prol da causa da proteção animal	26
FIGURA 7: Telas do aplicativo Adota Pet GO	27
FIGURA 8: Telas do aplicativo Adota Fácil	28
FIGURA 9: Telas do <i>website</i> Amigo não se Compra	29
FIGURA 10: Telas do <i>website</i> Amor Vira-lata	29
FIGURA 11: Objetivos do Desenvolvimento Sustentável	34
FIGURA 12: Objetivo global #15	35
FIGURA 13: Grupo de processos de gerenciamento de projetos	36
FIGURA 14: Matriz poder x interesse	37
FIGURA 15: Modelo Incremental	38
FIGURA 16: Armazenamento do projeto <i>Help a Pet</i> usando <i>Google Drive</i>	46
FIGURA 17: Controle de versão do projeto <i>Help a Pet</i> usando <i>Github</i>	47
FIGURA 18: Método para inclusão de um novo usuário da instituição	76
FIGURA 19: Método para gerar um <i>token</i> de segurança	76
FIGURA 20: Método para gerar uma nova senha para o usuário	77
FIGURA 21: Diagrama de sequência usando JWT	80
FIGURA 22: <i>Spring Data</i>	82
FIGURA 23: Diagrama de distribuição	84
FIGURA 24: Consistência e padrões visuais	87
FIGURA 25: Correspondência entre o sistema e o mundo real	87
FIGURA 26: Mensagens de ajuda nas entradas de dados	88
FIGURA 27: Mensagens que incentivam a adoção, o trabalho voluntário e as doações	97
FIGURA 28: Animais em destaque para adoção	98
FIGURA 29: Página sobre a lei de crimes ambientais	98
FIGURA 30: Página sobre o que pode ser considerado como maus-tratos	99
FIGURA 31: Página para cadastro da pessoa interessada na adoção	99
FIGURA 32: Página que mostra os dados de um animal que está no abrigo	100
FIGURA 33: Termo de adoção e responsabilidade	100

LISTA DE QUADROS

QUADRO 1: Entrevista parcial feita com administradores das instituições	32
QUADRO 2: Esforço estimado x realizado	44
QUADRO 3: Propostas de mudanças para o projeto <i>Help a Pet</i>	49
QUADRO 4: Métricas de estimativa do projeto	68
QUADRO 5: Estrutura física do banco de dados	72
QUADRO 6: <i>Design Patterns</i> adotados	74
QUADRO 7: Referências documentadas	88
QUADRO 8: Equipamento para realização de testes	89
QUADRO 9: Software para realização de testes	89
QUADRO 10: Itens a testar	90
QUADRO 11: Rastreabilidade entre requisitos e casos de teste	90
QUADRO 12: Papéis e responsabilidades na implantação	93
QUADRO 13: Treinamento previsto	94
QUADRO 14: Cronograma de atividades da implantação	94
QUADRO 15: Documentos de apoio à implantação	95

LISTAS DE ABREVIATURAS E SIGLAS

ABNT	Associação Brasileira de Normas Técnicas
API	<i>Application Programming Interface</i>
CASE	<i>Computer-Aided Software Engineering</i>
CEP	Código de Endereçamento Postal
CPF	Cadastro de Pessoa Física
CSS	<i>Cascading Style Sheets</i> (Folhas de Estilo Encadeadas)
DDL	<i>Data Definition Language</i>
DEPA	Delegacia Eletrônica de Proteção Animal
DER	Diagrama Entidade e Relacionamento
DML	<i>Data Manipulation Language</i>
EAP	Estrutura Analítica de Projeto
FAI	FAI - Centro de Ensino Superior em Gestão, Tecnologia e Educação
HD	<i>Hard Disk</i> (Disco Rígido)
HMAC	<i>Hash-based Message Authentication Code</i>
HTML	<i>HyperText Markup Language</i> (Linguagem de Marcação de Hipertexto)
HTTPS	<i>Hypertext Transfer Protocol Secure</i>

IBGE	Instituto Brasileiro de Geografia e Estatística
IDE	<i>Integrated Development Environment</i>
JDBC	<i>Java Database Connectivity</i>
JPA	<i>Java Persistence API</i>
JPEG	<i>Joint Photographics Experts Group</i>
JWT	<i>JSON Web Token</i>
MER	Modelo Entidade Relacionamento
MVC	<i>Model-view-controller</i> (Modelo-visão-controlador)
MVP	<i>Minimum Viable Product</i>
ODS	Objetivos do desenvolvimento Sustentável
ONG	Organização Não Governamental
ORM	<i>Object-relational Mapping</i>
PCU	Pontos de Casos de Uso
PDF	<i>Portable Document Format</i>
PFC	Projeto Final de Curso
PMBok	<i>Project Management Body of Knowledge</i>
PMI	<i>Project Management Institute</i>
RAM	<i>Random Access Memory</i> (Memória de Acesso Aleatório)
RESGACTI	Rede Solidária Entre Grupos Ambientais, Culturais e Trabalhos Intelectuais

REST	<i>Representational State Transfer</i>
RF	Requisito Funcional
RNF	Requisito Não Funcional
RSA	<i>Rivest – Shamir – Adleman Cryptography</i>
SGBD	Sistema Gerenciador de Banco de Dados
SMTP	<i>Simple Mail Transfer Protocol</i>
SUS	Sistema Único de Saúde
UML	<i>Unified Modeling Language</i>
UVZ	Unidade de Vigilância de Zoonoses
UNRIC	<i>United Nations Regional Information Center</i>

SUMÁRIO

1 INTRODUÇÃO	15
2 REVISÃO BIBLIOGRÁFICA	17
2.1 HISTÓRICO DE MAUS-TRATOS AOS ANIMAIS	17
2.2 IMPORTÂNCIA DE UMA EDUCAÇÃO PARA A PROTEÇÃO DOS ANIMAIS	19
2.3 VIGILÂNCIA DE ZOONOSES	20
2.4 PAPEL DAS ONGS NA PROTEÇÃO DOS ANIMAIS	22
2.5 FORMAS DE CONTRIBUIÇÃO PARA A PROTEÇÃO DOS ANIMAIS	23
2.6 TRABALHOS RELACIONADOS	26
3 OBJETIVO DO PROJETO	30
3.1 FORMULAÇÃO DO PROBLEMA	30
3.2 OBJETIVOS	32
3.3 JUSTIFICATIVA	32
3.4 PÚBLICO ALVO	33
3.5 NÍVEIS DE DECISÃO E GRUPOS FUNCIONAIS ATENDIDOS	33
3.6 CONSIDERAÇÕES SOBRE OS OBJETIVOS DO DESENVOLVIMENTO SUSTENTÁVEL (ODS)	33
4 GERÊNCIA DO PROJETO	36
4.1 PLANO DE PROJETO	36
4.2 ÁREAS DE CONHECIMENTO	41
5 ESPECIFICAÇÃO E ANÁLISE DOS REQUISITOS	52
5.1 REQUISITOS DO SISTEMA DE SOFTWARE	52
5.2 ANÁLISE DOS REQUISITOS	65
5.3 MÉTRICAS PARA ESTIMATIVAS DE ESFORÇO	66
6 ARQUITETURA E PROJETO DO SISTEMA DE SOFTWARE	67
6.1 VISÃO ESTRUTURAL	67
6.2 VISÃO COMPORTAMENTAL	68
6.3 VISÃO DE DADOS	69
6.4 VISÃO FÍSICA	70
6.5 PROJETO DO SISTEMA DISTRIBUÍDO	75
6.6 PROJETO DA INTERAÇÃO HUMANO-COMPUTADOR	83
7 PLANO DE TESTES	86
7.1 FINALIDADE	86
7.2 ESCOPO	86

7.3 ESPECIFICAÇÃO DOS CASOS DE TESTES	88
7.4 RESULTADOS	89
8 PLANO DA IMPLANTAÇÃO	90
8.1 METODOLOGIA	90
8.2 TREINAMENTO PREVISTO	91
8.3 CRONOGRAMA DE IMPLANTAÇÃO	92
8.4 DOCUMENTOS DE APOIO À IMPLANTAÇÃO	93
8.5 VISÃO DA IMPLANTAÇÃO	93
9 CONCLUSÃO	94
REFERÊNCIAS	99
OBRAS CONSULTADAS	104
APÊNDICE A - ENTREVISTAS	105
APÊNDICE B – ESTRUTURA ANALÍTICA DO PROJETO	106
APÊNDICE C – CRONOGRAMA E DICIONÁRIO EAP	107
APÊNDICE D – RELATÓRIO DE DESEMPENHO	108
APÊNDICE E – LISTA DE VERIFICAÇÃO DA QUALIDADE	109
APÊNDICE F – ACOMPANHAMENTO DOS RISCOS	110
APÊNDICE G – DIAGRAMAS DE CASOS DE USO	111
APÊNDICE H – DESCRIÇÃO DOS CASOS DE USO	112
APÊNDICE I – DIAGRAMA ENTIDADE-RELACIONAMENTO	113
APÊNDICE J – MODELO INICIAL DE INTERFACE DE USUÁRIO	114
APÊNDICE K – ESTIMATIVA DE ESFORÇO	115
APÊNDICE L – DIAGRAMA DE PACOTES	116
APÊNDICE M – DIAGRAMA DE CLASSES	117
APÊNDICE N – DIAGRAMA DE OBJETOS	118
APÊNDICE O – DIAGRAMA DE SEQUÊNCIA	119
APÊNDICE P – DIAGRAMA DE VISÃO GERAL DE INTERAÇÃO	120
APÊNDICE Q – MODELO LÓGICO	121
APÊNDICE R – DICIONÁRIO DE DADOS	122
APÊNDICE S – ENTREVISTAS (PERFIL DO USUÁRIO)	123
APÊNDICE T – DIAGRAMA DE ATIVIDADES	124
APÊNDICE U – SCRIPT DDL	125
APÊNDICE V – SCRIPT DML	126
APÊNDICE X – DESCRIÇÃO E RESULTADO DOS CASOS DE TESTE	127
APÊNDICE W – MANUAL DO USUÁRIO	127

APÊNDICE Y – DIAGRAMA DE IMPLANTAÇÃO	129
APÊNDICE Z – DIAGRAMA DE COMPONENTES	130
ANEXO A – TERMO DE ADOÇÃO	131
ANEXO B – MANUAIS DE INSTALAÇÃO E CONFIGURAÇÃO	132

1 INTRODUÇÃO

A população de animais que vive nas ruas é expressiva, sendo a maior parte constituída por cães e gatos não domiciliados, abandonados, desamparados ou perdidos. Trata-se de uma situação que requer a atenção de toda a sociedade, porque dela decorrem inúmeros problemas, como maus-tratos, doenças e desequilíbrio ambiental.

Muitos animais para sobreviverem dependem da ajuda de Organizações não Governamentais (ONGs), abrigos e associações protetoras dos animais. Pelo elevado número dessa população, essas entidades têm a tendência de superlotarem seus abrigos e de necessitarem de apoio e doações para prestarem seus serviços de resgate, guarda, alimentação, tratamento veterinário, vacinação, esterilização, dentre outros.

Diante disso, este projeto, chamado *Help a Pet*, tem a intenção de apoiar a causa da proteção animal, por meio do desenvolvimento de um sistema de software a ser disponibilizado na Web. Esse sistema deve ser capaz de facilitar a divulgação dos serviços das entidades protetoras dos animais e adoção dos animais mantidos nos seus abrigos.

No presente documento, apresenta-se o processo de desenvolvimento do projeto *Help a Pet*, contemplando os estudos, o planejamento e gerenciamento, a modelagem do sistema de software, a codificação e os registros dos testes realizados.

Este documento se divide inicialmente em 5 capítulos, sendo que o primeiro capítulo apresentado é a introdução.

O Capítulo 2 apresenta a revisão bibliográfica e todo embasamento necessário para a elaboração do projeto.

O Capítulo 3 apresenta o problema, a justificativa para a realização do projeto, o público alvo, os níveis de decisão e grupos funcionais atendidos.

O Capítulo 4 apresenta o plano e a gerência do projeto e aborda as gerências do escopo, do tempo, da integração, da qualidade e dos riscos.

O Capítulo 5 apresenta os requisitos funcionais e não funcionais da aplicação *Help a Pet*, também exhibe o modelo de casos de uso, o modelo conceitual dos dados, o modelo da interface de usuário e as estimativas de esforços.

O Capítulo 6 apresenta a arquitetura do sistema de software, a visão estrutural, comportamental e física. Também mostra os padrões de projeto escolhidos, as convenções e os guias para codificação, bem como o projeto da distribuição e da interação humano-computador.

O Capítulo 7 apresenta o plano de testes e contém uma descrição dos testes, os itens testados, apresentando também os resultados obtidos destes testes.

O Capítulo 8 apresenta o plano da implantação e são descritas a metodologia de implantação, treinamento previsto para que os usuários utilizem corretamente o sistema, cronograma de implantação, toda documentação de apoio à implantação do sistema e uma visão da implantação.

O Capítulo 9 apresenta a conclusão do projeto, sendo seguido das referências, apêndices e anexos.

2 REVISÃO BIBLIOGRÁFICA

O objetivo deste capítulo é apresentar o referencial teórico consultado para realização deste projeto. Apresentam-se o problema histórico de maus-tratos a que são submetidos os animais, a importância de uma educação para a conscientização da proteção aos animais e o trabalho da vigilância de zoonoses. O papel das ONGs e formas de contribuição para a proteção dos animais também são discutidos. Por fim, são apresentados alguns trabalhos relacionados com o presente projeto.

2.1 HISTÓRICO DE MAUS-TRATOS AOS ANIMAIS

Entende-se por maus-tratos o ato de submeter alguém ao tratamento cruel, trabalho forçado e/ou privação de alimentos ou cuidados, prejudicando-lhe a saúde física ou mental. Esse crime é praticado pelos mais variados tipos de pessoas e os motivos envolvem aspectos culturais, sociais e psicológicos, sendo muitas vezes praticado sem a consciência de que tal ato é prejudicial (DELABARY, 2012).

A Lei Federal nº 9.605, de 12 de fevereiro de 1998, dispõe sobre as sanções penais e administrativas derivadas de condutas e atividades lesivas ao meio ambiente e nelas se enquadram práticas de maus-tratos em animais silvestres, domésticos ou domesticados (BRASIL, 1998).

Contudo, segundo a Delegacia Eletrônica de Proteção Animal (DEPA) da Secretaria da Segurança Pública (SSP) de São Paulo, em dois anos de funcionamento, ela recebeu mais de 16 mil denúncias de agressões e violência contra animais domésticos. Muitas foram feitas por telefone celular, com o envio de fotos e vídeos. De janeiro ao início de novembro de 2017, foram 8.162 denúncias - 25 por dia no estado. Em todo o ano de 2017, 8.193 denúncias aconteceram. Praticamente, todas foram acatadas e verificadas, e em muitos casos, os animais foram salvos (TOMAZELA, 2018).

Em um trabalho sobre a relação entre humanos e animais na cidade de São Paulo, no período entre o final do século 19 e o início do século 20, Aprobato Filho (2006) explica que o processo de modernização dessa cidade se iniciou, gradualmente, pelo extermínio e isolamento de certas espécies animais. Aprobato Filho (2006) descreve ainda que, em 1907, a carta de um leitor ao jornal O Estado de São Paulo relatava o fato de que cães soltos pela

cidade ofereciam perigo à população e reclamava da passagem da “carrocinha” que recolhiam esses animais das ruas, mas que depois os soltavam. Essa carta demonstrou que a população não estava preparada para essa “guerra urbana” que a prefeitura travava, desde há muito, contra os cães, submetendo-os ao sacrifício. Não houve nenhuma ação pedagógica da parte da prefeitura que não fosse punitiva, revelando-se essa falha. Na época, a prefeitura de São Paulo condenou milhares de cães à morte ao invés de empreender uma reeducação da população local. Os cães pagaram pelos hábitos de seus donos. Com isso, pode-se compreender que os problemas relacionados aos maus-tratos e abandonos de animais acontecem há muitos anos e, até o presente, esses maus-tratos são evidenciados pelo grande número de denúncias de violência.

Segundo Moutinho, Nascimento e Paixão (2015):

Durante muitos anos, houve um predomínio de ações sistemáticas de captura e eliminação destes animais por meio da eutanásia em massa efetuada pelo Poder Público. Como essas ações não resultaram em redução considerável na densidade populacional e, considerando-se ainda aspectos éticos e legais, essa estratégia vem sendo paulatinamente substituída por ações focadas na esterilização em massas dos cães, que é uma das medidas preconizadas [...]. Acredita-se que, ao atuar sobre a reprodução destes animais, conseguir-se-á, com apoio de ações de educação para guarda responsável e legislação apropriada, o almejado controle da densidade populacional. Nesse contexto, o papel das Organizações não Governamentais de Proteção Animal é de grande importância.

Conforme a Lei Nº 9.605/1998, maltratar animais é crime e segundo Calhau (2005) há muito tempo foi superado o entendimento que os animais são coisas sem nenhuma proteção jurídica. Em 2018, o Senado Federal aprovou um projeto de lei de autoria do senador Randolfe Rodrigues, que amplia as sanções para crimes de maus-tratos a animais, alterando a pena de 3 meses a 1 ano de detenção para 1 a 4 anos de detenção, com a possibilidade de multa mantida. Também estabelece punição financeira para estabelecimentos comerciais que concorrerem para o crime. O projeto de lei foi encaminhado para análise da Câmara dos Deputados (AGENCIA DO SENADO, 2018).

Vale ressaltar que os maus-tratos estão relacionados a todo tipo de animal e de diversas formas, como: caça exploratória de animais; confinamento de animais em cativeiros; brigas de galo; transporte inadequado de animais; falta de ajuda a um animal ferido; submissão a trabalho e carga exaustiva (cavalos e jumentos); contrabando; abandono; exposição em

shows que possam lhes causar lesão, pânico ou estresse; prisão de animal em correntes ou cordas muito curtas; falta de abrigo do sol, chuva e frio; guarda do animal em locais pequenos e anti-higiênicos; envenenamento; dentre outros.

De acordo com Portinari (2018), um caso de repercussão pública notória ocorreu em Osasco (SP), em 2018, quando um funcionário da rede Carrefour foi acusado de ter envenenado e espancado um cachorro. O animal morreu e fotos dele ferido foram publicadas nas redes sociais, conforme mostra a Figura 1. O crime tem sido investigado pela Polícia Civil e foi citado como justificativa para projeto de lei do senador Randolfe Rodrigues.



FIGURA 1: Foto do animal submetido aos maus-tratos em Osasco-SP

FONTE: Portinari (2018)

2.2 IMPORTÂNCIA DE UMA EDUCAÇÃO PARA A PROTEÇÃO DOS ANIMAIS

Na maioria das vezes, os maus-tratos contra animais sequer são denunciados, porque para algumas pessoas são considerados naturais, devido ao seu alto índice de ocorrência. Muitos desses atos estão vinculados a um comportamento humano permissivo quanto à ignorância ou crueldade praticada pelas pessoas. Ademais, devido à necessidade ou ganância, os direitos dos animais são deixados de lado para atender aos anseios dos seres humanos.

A omissão aos maus-tratos é um dos aspectos mais preocupantes, porque favorece que atos cruéis continuem acontecendo e sejam passados adiante para as gerações futuras. É preciso realizar um trabalho abrangente e eficaz dentro das comunidades para que os animais não sejam mais vistos como objetos.

Sendo assim, a educação sobre a proteção dos animais deve ser a principal ferramenta para diminuir os casos de maus-tratos, visto que esses ensinamentos podem conscientizar e encorajar a sociedade a denunciar esse crime.

[...] Todos os seres vivos devem ser considerados por terem valores por si mesmos e não por sua utilidade ou capacidade de servir ao homem. O ser humano precisa compreender que suas ações geram consequências não somente sobre outros seres humanos, mas sobre todas as formas de vida. E que todas são igualmente importantes e merecedoras do nosso respeito (SCHEFFER, 2016, p. 2).

Várias campanhas estão sendo feitas pela Internet, televisão e rádio, a fim de tentar conter esses maus-tratos, conforme mostra a Figura 2.



FIGURA 2: Campanha da Delegacia Eletrônica de Proteção Animal

Fonte: DEPA (2019)

De acordo com Toledo e Girardi (2016), o aumento das denúncias pode ser explicado pela facilidade em se obter provas. A maior parte da população possui um aparelho celular com câmera fotográfica, o que facilita a denúncia juntamente com provas da violência infligida ao animal.

A educação da população quanto aos cuidados para com os animais, sintetizados na noção de posse ou guarda responsável é um elemento fundamental. Retira-se, portanto, dos animais qualquer responsabilidade e a coloca sobre os humanos.

2.3 VIGILÂNCIA DE ZOONOSES

O abandono de animais reflete na superpopulação de cães e gatos vivendo nas ruas das cidades. De acordo com Santana e Marques (2001), esse quadro se agrava a cada dia, pois são muitos animais parindo, e o período de gestação é curto, geralmente, a cada três meses, o que dificulta o controle. Essa superpopulação abre as portas para um grande número de

maus-tratos em que os animais são surrados, envenenados ou passam fome e frio. Devido a essas condições precárias, cães e gatos de rua se tornam vetores de zoonoses.

A vigilância de zoonoses é uma área integrante da vigilância epidemiológica do Sistema Único de Saúde (SUS) que desenvolve atividades, ações e estratégias para o controle de zoonoses, das doenças transmitidas por vetores e dos agravos causados por animais peçonhentos e que têm como enfoque a vigilância e o controle de vetores, hospedeiros, reservatórios, amplificadores, portadores, suspeitos ou suscetíveis às zoonoses e de animais peçonhentos (MINISTÉRIO DA SAÚDE, 2018).

As estruturas físicas e técnicas, denominadas Unidade de Vigilância de Zoonoses (UVZ), podem estar organizadas de forma municipal, regional e/ou estadual. A UVZ é responsável por ações e serviços de vigilância das populações de animais de relevância para a saúde pública, com o objetivo de identificar oportuna e precocemente o risco e, assim, prevenir e monitorar as zoonoses e os acidentes causados por animais peçonhentos e venenosos.

De acordo com o Ministério da Saúde (2018, p. 2), os objetivos específicos das Unidades de Vigilância de Zoonoses são:

- a) realizar ações, atividades e estratégias de vigilância, de forma contínua e sistemática, de populações de animais potencialmente ou sabidamente de relevância para a saúde pública;
- b) realizar ações, atividades e estratégias de prevenção, de forma sistemática, de zoonoses e de acidentes causados por animais peçonhentos e venenosos, de relevância para a saúde pública;
- c) realizar ações, atividades e estratégias de controle, quando pertinente e necessário, de animais peçonhentos, venenosos, vetores, hospedeiros, reservatórios, amplificadores, portadores, suspeitos ou suscetíveis às zoonoses, quando estes forem de relevância para a saúde pública.

Em uma pesquisa realizada no município de Santa Rita do Sapucaí (MG), pôde-se verificar que não há Unidade de Vigilância de Zoonoses. Suspeitas de doenças e problemas relativos aos animais, geralmente, são relatados à vigilância sanitária do município (FERRARI, 2019).

2.4 PAPEL DAS ONGS NA PROTEÇÃO DOS ANIMAIS

Mesmo nas cidades em que o poder público tem atuado para proteger os animais, o papel das organizações não governamentais é fundamental para auxiliar nesse trabalho. Quando o poder público não atua para o bem-estar animal, limitando-se apenas ao controle de zoonoses, as ONGs e associações protetoras de animais passam a ser a única esperança de cuidados para milhares de animais abandonados pelas ruas.

De acordo com Silva Neta *et al.* (2014, p. 199):

As Organizações Não Governamentais (ONGs) voltadas para a proteção animal desempenham um papel importante como divulgadoras de informações que visem o bem-estar animal, atuando também na saúde pública, quando realizam resgate de cães e gatos de rua.

Essas ONGs fazem o recolhimento dos animais nas ruas e encaminham os que estão doentes ou feridos para tratamento médico-veterinário. Com o restabelecimento da saúde, os animais permanecem em abrigos, onde são expostos para a adoção no local, em feiras e por fotos divulgadas pela Internet. Para realizarem essas atividades, as ONGs precisam de ajuda para custear suas despesas com rações, medicamentos, aluguel de abrigo e combustíveis. As doações e o serviço voluntário são muito importantes, sendo necessários produtos de limpeza para a higienização do local onde os animais são mantidos, cobertores, apoio para resgate de animais feridos, abandonados ou que estão em circunstâncias irregulares, lares temporários, eventos para a adoção, bazares, apadrinhamento de algum animal, dentre outros.

A ONG Rede Solidária Entre Grupos Ambientais, Culturais e Trabalhos Intelectuais (Resgacti), que atua em Santa Rita do Sapucaí e Itajubá (MG), publica em suas redes todos os gastos e arrecadações que obtiveram em cada mês. A Figura 3 mostra os saldos financeiros no mês de março de 2019.

SALDO PREVISTO ATÉ 08/03/2019:	
Saldo Inicial Março Itajubá:	R\$1.023,29
Saldo Inicial Março Sta. Rita do Sapucaí:	R\$3.325,95
Saldo Total:	R\$4.349,24
Saldo Arrecadação Março Itajubá:	R\$1.040,00
Saldo Arrecadação Março Sta. Rita do Sapucaí:	R\$342,00
Saldo Total:	R\$1.382,00
Saldo Despesa Itajubá:	R\$2.080,58
Saldo Despesa Sta. Rita do Sapucaí:	R\$2.035,98
Saldo Total:	R\$4.116,56
Saldo Final Itajubá	-R\$17,29
Saldo Final Sta. Rita do Sapucaí	R\$1.631,97

FIGURA 3: Saldo financeiro referentes ao mês de março 2019

FONTE: Resgacti (2019)

A ONG Resgacti ajuda a todos os tipos de animais e não se restringe somente aos domésticos. Em entrevistas realizadas com outras ONGs das cidades de Santa Rita do Sapucaí, Pouso Alegre e Itajubá, pôde-se constatar que elas ajudam animais de várias espécies, como: tatu, lobo-guará, tucano, porco-espinho, arara, coelho, ramister, lagarto e cobra.



FIGURA 4: Lagarto ferido resgatado pela ONG Resgacti

FONTE: Resgacti (2019)

2.5 FORMAS DE CONTRIBUIÇÃO PARA A PROTEÇÃO DOS ANIMAIS

A ajuda em prol da proteção aos animais pode ser realizada de diversas maneiras e são muitas ações que podem ser realizadas para ajudá-los. Algumas pessoas limitam-se ao

pensamento de que a ajuda a algum animal compete ao poder público, às ONGs, associações ou protetores que já atuam nesse cenário, ou que a ajuda necessariamente tenha que vir seguida de recursos financeiros.

A proteção animal pode ser realizada por diversas formas, como:

- a) denúncia de ocorrência de maus-tratos;
- b) compartilhamento de campanhas contra maus-tratos;
- c) divulgação de feiras de adoção e de eventos para ajuda à causa animal;
- d) serviço voluntário nas ONGs para realizar atividades propostas pelas mesmas;
- e) doações (cobertores, ração, utensílios, produtos de limpeza e dinheiro);
- f) adoção de animal ou resgate da rua;
- g) apadrinhamento de algum animal que esteja em abrigo;
- h) promoção de eventos para arrecadar recursos para alguma ONG;
- i) compra de produtos de alguma ONG;
- j) participação de mutirões de limpeza e castração.

O compartilhamento de campanhas realizadas contra o abandono é uma das formas simples de se ajudar à causa da proteção animal.



FIGURA 5: Campanha do combate ao abandono de animais
FONTE: Jucá (2018)

A oferta de lar temporário é uma opção para quem quer ajudar os animais, mas não pode adotá-los. Essa prática consiste em um abrigo temporário ao animal em situação de abandono. As pessoas que ajudam dessa forma colocam-se como responsáveis em cuidar do animal até que ele seja adotado por outra pessoa.

De acordo com Grisci, Locatelli e D’Arisbo (2017, p. 25):

A prática de acolhimento temporário de animais vem apresentando uma nova modalidade de intervenção à causa de proteção animal. O acolhimento temporário de animais por protetores independentes – indivíduos sem vínculo com ONGs, associações ou abrigos de animais tem se feito notar, em especial, via postagens nas redes de relacionamento.

Essa forma de ajudar na forma de oferta de lares temporários é uma ação que o grupo de voluntários *Yellow Brick* apoia. O *Yellow Brick* surgiu em Santa Rita do Sapucaí, em maio de 2017, com o intuito de fomentar a cena de música autoral na cidade e, a longo prazo, criar uma rede de auxílio de pessoas que se dispõem a resgatar animais vítimas de maus-tratos.

O *Yellow Brick* organiza o ‘Música a Domicílio’, um evento periódico que une bandas locais e de fora nas residências de pessoas que acreditam na música independente brasileira. Sendo assim, o grupo utiliza outra forma de apoiar a causa da proteção animal, que é a realização de eventos culturais, sendo as quantias arrecadadas destinadas às pessoas que ajudam os animais. O grupo possui uma rede de contatos para obter informações de pessoas que precisam de ajuda com algum animal, sendo ele de qualquer espécie. Dessa forma, conseguem repassar uma quantia para poder ajudar nos cuidados desse animal. Segundo a criadora do grupo: “criei um serviço para que as pessoas vissem valor e usassem o serviço, no caso, os eventos para apoiar os lares temporários” (OPPENHEIMER, 2019).

O grupo de voluntários do *Yellow Brick* ajuda na causa de proteção dos animais com foco na sustentação financeira de lares temporários. Contudo, ele ajuda a qualquer animal em qualquer situação, até mesmo os que estão abrigados em ONGs. A Figura 6 mostra o logotipo do grupo.



FIGURA 6: Grupo de voluntários em prol da causa da proteção animal

FONTE: *Yellow Brick* (2019)

Esse grupo já foi procurado por uma ONG que precisava de ajuda com os custos de um cachorro que se encontrava em situações de enfermidade, mas que a ONG não tinha condição de cuidar. O *Yellow Brick* realizou um evento e conseguiu levantar recursos para outro animal que, posteriormente, veio a óbito. O dinheiro então foi destinado para a ONG cuidar do cachorro, conforme havia sido solicitado.

De acordo com Oppenheimer (2019), os eventos realizados eram cobrados somente na bilheteria, contudo, o grupo passou a oferecer serviços como praça de alimentação, a fim de aumentar a arrecadação. Vale ressaltar que todo o alimento fornecido dentro do evento é de origem vegana.

2.6 TRABALHOS RELACIONADOS

Nesta seção são apresentadas aplicações de software similares às propostas apresentadas pelo projeto *Help a Pet*. São citadas quatro soluções que oferecem serviço de adoção de animais. São elas: Adota Pet GO, Adota Fácil, Amigo não se Compra e Amor Vira-lata.

2.6.1 Adota Pet GO

O Adota Pet GO é um aplicativo (app) *mobile*, de licença gratuita, desenvolvido por Marlon Henrique Ramalho Afonso, formado em Análise e Desenvolvimento de Sistemas pelo Instituto de Educação Superior de Brasília, com a colaboração do seu colega Ruben Santos de Almeida e apoio do seu professor Orion Teles.

O Adota Pet GO funciona da seguinte maneira: um usuário pode publicar os dados de um animal para adoção com nome, foto, descrição, espécie, raça e localização. Os interessados em adotar recebem uma lista dos animais que estão mais perto, usando o sistema de geolocalização do aplicativo e podem conversar com o tutor temporário do animal por meio de um *chat*. O aplicativo pode ser usado em qualquer lugar do mundo e tem versão para a língua inglesa (ADOTA PET GO, 2019).

O aplicativo está disponível na Google Play, uma loja de *apps* para dispositivos móveis com Android. A Figura 7 ilustra algumas das telas projetadas em um dispositivo móvel.

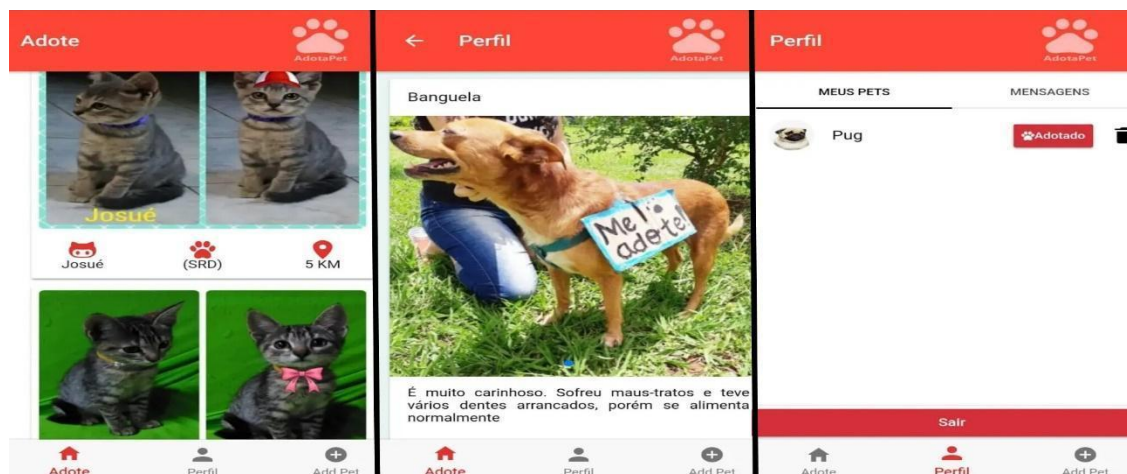


FIGURA 7: Telas do aplicativo Adota Pet GO

FONTE: Adota Pet GO (2019)

2.6.2 Adota Fácil

O Adota Fácil é uma aplicação *mobile*, de licença gratuita, a ser utilizada para intermediar a adoção de cães e gatos entre pessoas e ONGs no Brasil.

Um usuário desse aplicativo, após se cadastrar ou entrar via uma rede social, consegue encontrar uma lista com todos os animais disponíveis para adoção. Essa lista pode ser gerada somente com animais localizados em abrigos a certa distância e por espécie animal. Também é possível que o usuário divulgue seu animal para adoção, caso não consiga mais cuidar dele, para que, assim, o animal possa ser adotado por outra pessoa.

Quando houver um interesse mútuo, ambas as partes poderão se conectar por meio do *chat* do aplicativo. Também é possível conhecer as avaliações feitas dos usuários para que haja uma confiança antes de adotar o animal de alguém.

Esse aplicativo se encontra disponível na Google Play. A Figura 8 ilustra algumas das telas projetadas em um dispositivo móvel.

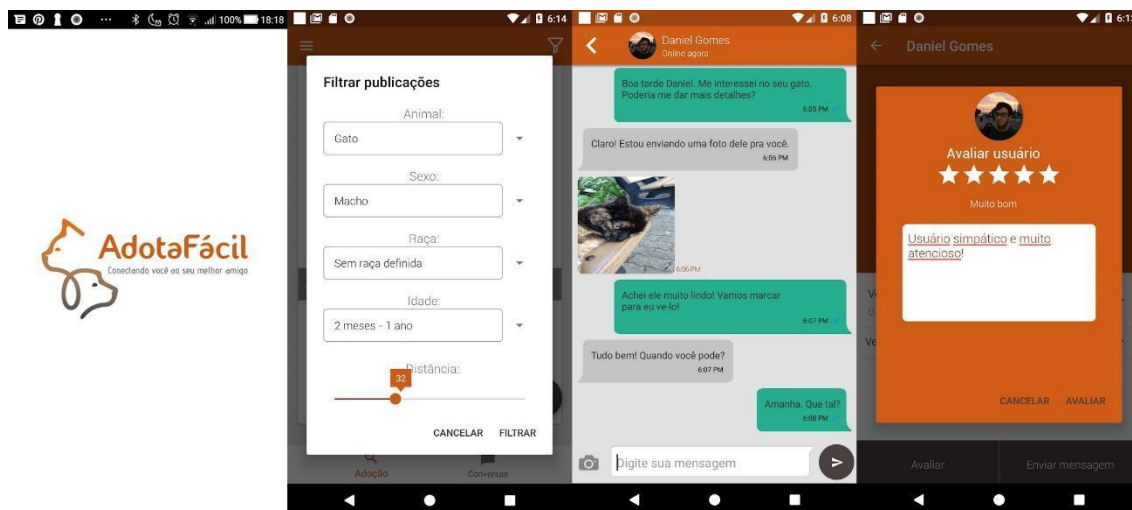


FIGURA 8: Telas do aplicativo Adota Fácil

FONTE: Adota Fácil (2019)

2.6.3 Amigo não se Compra

O Amigo não se Compra é uma associação criada em 2017. Ela disponibiliza uma aplicação na *Web* em que ONGs e protetores de todo o Brasil podem divulgar cães e gatos para adoção.

A aplicação Web funciona da seguinte maneira:

- a) ONGs ou protetores fazem o seu cadastro gratuitamente e, logo após, podem publicar os dados dos animais com detalhes sobre suas características e personalidades;
- b) uma pessoa interessada em adotar um animal pode acessar a aplicação *Web* e procurar pelos disponíveis para adoção;
- c) quando a pessoa encontra o animal desejado, ela seleciona a opção “quero adotar” que reservará o animal para adoção. Em seguida, ela deve combinar com a ONG as providências para retirada do animal.

A Figura 9 ilustra a página *Web* da associação Amigo não se compra.

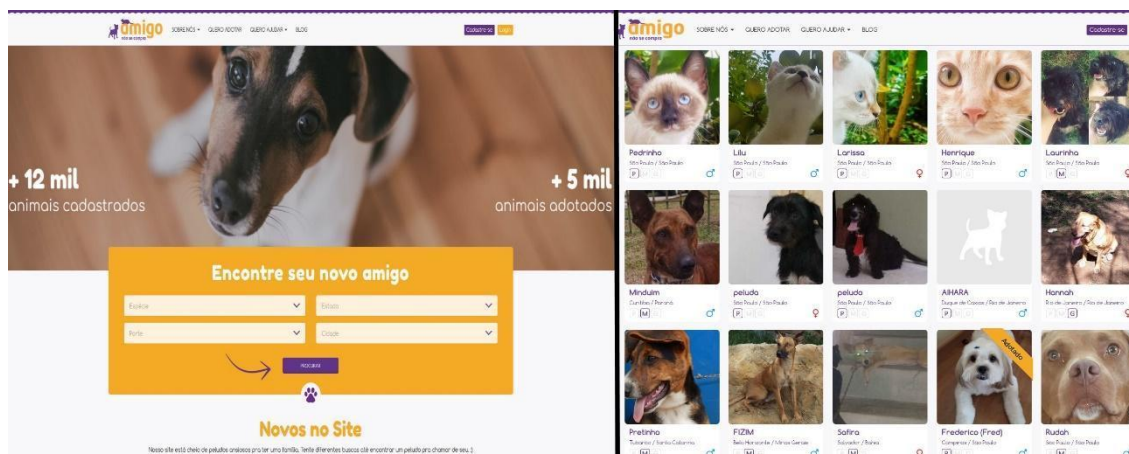


FIGURA 9: Telas do *website* Amigo não se Compra

FONTE: Amigo não se Compra (2019)

2.6.4 Amor Vira-lata

O Amor Vira-lata é uma organização não governamental, sem fins lucrativos, criada oficialmente em 2013, onde todo trabalho é prestado de forma voluntária. Ela presta serviços de utilidade pública que atende Caxias do Sul (RS) e região.

O site da ONG possui um banco de dados de animais para adoção, achados e perdidos, com mecanismo de busca e filtros de anúncios. Mais de cinco mil animais já foram anunciados no site do Amor Vira-lata, desde 2010, quando começou a funcionar, até meados de 2014. A Figura 10 ilustra a página principal do *website*.

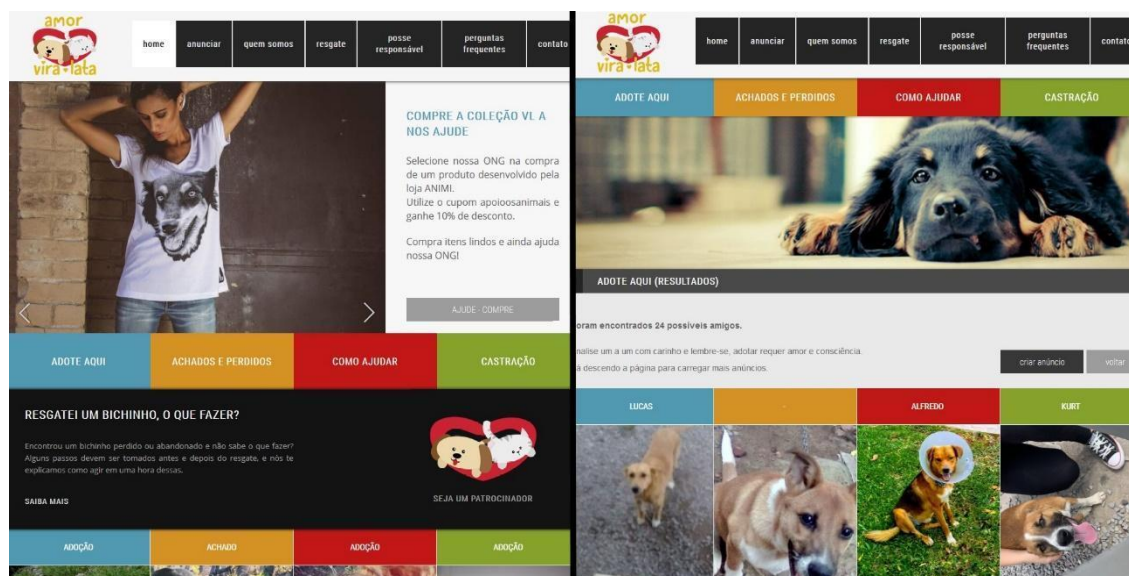


FIGURA 10: Telas do *website* Amor Vira-lata

FONTE: Amor Vira-lata (2019)

3 OBJETIVO DO PROJETO

Neste capítulo é apresentada a formulação do problema, mostrando a necessidade de realização deste projeto, os objetivos traçados, a justificativa, o público alvo e os níveis de decisão e grupos funcionais atendidos.

3.1 FORMULAÇÃO DO PROBLEMA

O problema abordado por este projeto ocorre no Brasil e também no mundo. Maus-tratos e abandono de animais, falta de cuidado quanto ao controle populacional dos animais de rua, poucos serviços governamentais para lidar com esterilização e controle de doenças, dentre outros, são situações corriqueiras.

Por causa delas, as pessoas criaram abrigos, associações e ONGs buscando cuidar dos animais abandonados por seus donos, desaparecidos, acidentados, doentes ou que sofrem maus-tratos, tentando assim amenizar uma parte desses problemas.

Para estudar o problema no âmbito da microregião compreendida por Pouso Alegre, Santa Rita do Sapucaí e Itajubá (MG), as seguintes associações foram entrevistadas: Associação Francisco de Assis de Proteção aos Animais, Rede Solidária Entre Grupos Ambientais, Culturais e Trabalhos Intelectuais (Resgacti) e Associação de Proteção aos Animais – SOS Bichos. Foram feitas entrevistas com as pessoas que trabalham nessas instituições, visando levantar suas dificuldades e formas para lidar com os problemas locais.

No Quadro 1 é mostrado, de forma parcial, o que foi perguntado e as respostas obtidas. As entrevistas completas encontram-se no Apêndice A deste documento.

	ONGs e Associações		
Questionário	Francisco de Assis	Resgacti	SOS Bichos
Bloco 1			
Quais são os animais acolhidos pela instituição?	Cães.	Cães, gatos e outros animais.	Cães e gatos.
Quais são os meios de divulgação?	Facebook, Instagram e Whatsapp.	Facebook, Instagram e Whatsapp.	Facebook, Instagram, Whatsapp e jornais locais.
Possui <i>website</i> ?	Não.	Sim.	Não.
Gostaria de ter um <i>website</i> ?	Sim.	Sim.	Sim.
Bloco 2			
Quais são as principais dificuldades da instituição?	Arrecadar verbas para sustento da instituição.	Arrecadar verbas para sustento da instituição.	Arrecadar verbas para sustento da instituição.
Quais são as formas de lidar com as dificuldades?	Rifas, sorteio de prendas e outros eventos.	Eventos.	Rifas, bazares, cofrinhos no comércio e eventos.
Bloco 3			
Quais são as etapas para se adotar um animal?	Visita ao local e assinatura do termo de adoção.	Visita ao local e assinatura do termo de adoção.	Visita ao local e assinatura do termo de adoção.
Quais os meios para divulgar a adoção de um animal?	Facebook, Instagram, Whatsapp e eventos.	Facebook, Instagram, Whatsapp e eventos.	Facebook, Instagram, Whatsapp e eventos.
Quais são as dificuldades para lidar com esse problema?	Preferência por animais de porte pequeno e divulgação.	Falta de voluntários para divulgar.	Preferência dos adotantes e preconceito com alguns animais.

QUADRO 1: Entrevista parcial feita com administradores das instituições

Analisando o Quadro 1, nas três instituições visitadas, os problemas são resolvidos de formas similares, sempre por meio de divulgação nas redes sociais e pela busca para arrecadar fundos e mostrar animais que necessitam de adoção.

O projeto *Help a Pet* tem como proposta automatizar as etapas para adotar um animal (mostradas no Bloco 3), que é feita de forma presencial, e o preenchimento do termo de adoção que é feito de forma manual. A ideia é que o sistema gere um termo com todos dados preenchidos, tanto do adotante quanto do animal, além de ser um sistema que visa aumentar a divulgação dos animais que necessitam de lares, auxiliando as instituições com a liberação de espaço nos abrigos, a fim de que possam atender outros animais necessitados.

Alguns modelos do termo de adoção estão no Anexo A deste documento.

3.2 OBJETIVOS

O objetivo geral deste projeto final de curso é desenvolver um sistema de software, para plataforma *Web*, que possibilite às ONGs, abrigos e associações protetoras de animais divulgarem os animais resgatados para adoção. Por meio desse aplicativo *Web*, poderá ser intermediado o processo de adoção desses animais mantidos sob a guarda das instituições protetoras.

O sistema de software deverá permitir aos seus usuários:

- a) acesso online das instituições que cuidam de animais abandonados, possibilitando a realização do cadastro dos animais disponíveis para adoção;
- b) acesso online para público que busca por adoção do animal;
- c) gerar um termo de adoção ao final do processo com todos dados preenchidos.

3.3 JUSTIFICATIVA

Segundo as entrevistas realizadas com as ONGs de Santa Rita do Sapucaí e Pouso Alegre, abrigos e associações protetoras dos animais, foi possível identificar que as principais dificuldades são a falta de adoção dos animais e a arrecadação de verbas para o sustento da instituição.

Sendo assim, um sistema para divulgar os animais abrigados e auxiliar no processo de adoção, ajudará essas instituições a abrir novas vagas em suas instalações para que possam resgatar outros animais.

3.4 PÚBLICO ALVO

A aplicação web *Help a Pet* tem como público alvo, as ONGs, abrigos e associações protetoras dos animais. Dessa forma, visa atingir as pessoas que desejam adotar um animal ou contribuir para a manutenção dessas entidades protetoras.

3.5 NÍVEIS DE DECISÃO E GRUPOS FUNCIONAIS ATENDIDOS

O sistema de software atuará em nível operacional. O usuário poderá analisar e visualizar os dados do animal, escolhendo aquele que achar mais adequado para sua convivência, porém deverá passar pela entrevista com a instituição, a fim de finalizar o processo de adoção.

Os grupos funcionais atendidos são as instituições que anunciarão seus animais para adoção e os usuários adotantes que acessarão a aplicação *Web* para escolher algum animal para adoção. A quantidade e diversidade dos animais para adoção serão proporcionadas pelas instituições.

3.6 CONSIDERAÇÕES SOBRE OS OBJETIVOS DO DESENVOLVIMENTO SUSTENTÁVEL (ODS)

Os Objetivos de Desenvolvimento Sustentável (ODS) estão presentes em uma agenda mundial adotada durante a Cúpula das Nações Unidas sobre o Desenvolvimento Sustentável, em setembro de 2015, e composta por 17 objetivos e 169 metas a serem atingidos até 2030.

Nesta agenda estão previstas ações mundiais nas áreas de erradicação da pobreza, segurança alimentar, agricultura, saúde, educação, igualdade de gênero, redução das desigualdades, energia, água e saneamento, padrões sustentáveis de produção e de

consumo, mudança do clima, cidades sustentáveis, proteção e uso sustentável dos oceanos e dos ecossistemas terrestres, crescimento econômico inclusivo, infraestrutura, industrialização, entre outros.

Os temas podem ser divididos em quatro dimensões principais:

a) social: relacionada às necessidades humanas, de saúde, educação, melhoria da qualidade de vida e justiça;

b) ambiental: trata da preservação e conservação do meio ambiente, com ações que vão da reversão do desmatamento, proteção das florestas e da biodiversidade, combate à desertificação, uso sustentável dos oceanos e recursos marinhos até a adoção de medidas efetivas contra mudanças climáticas;

c) econômica: aborda o uso e o esgotamento dos recursos naturais, a produção de resíduos, o consumo de energia, entre outros; e

d) Institucional: diz respeito às capacidades de colocar em prática os ODS.

Os ODS foram construídos em um processo de negociação mundial, que teve início em 2013 e contou com a participação do Brasil em suas discussões e definições a respeito desta agenda. A Figura 11 mostra os 17 objetivos elencados na Agenda 2030.



FIGURA 11: Objetivos do Desenvolvimento Sustentável

FONTE: UNRIC (2019)

O projeto *Help a Pet* pode contribuir com o Objetivo 15 - Proteger a Vida Terrestre (proteger, recuperar e promover o uso sustentável dos ecossistemas terrestres, gerir de forma sustentável as florestas, combater a desertificação, deter e reverter a degradação da terra e deter a perda de biodiversidade), pois aborda um problema que ocorre no Brasil e no mundo, o maus-tratos e abandono de animais, falta de cuidado quanto ao controle populacional dos animais não domiciliados, poucos serviços governamentais para lidar com esterilização e controle de doenças, dentre outros. Sendo assim, propõe um sistema de software que visa divulgar os animais abrigados e auxiliar no processo de adoção deles, ajudando instituições a abrirem novas vagas em suas instalações para que possam resgatar outros animais. Por meio desse sistema, espera-se amenizar os impactos relacionados aos maus-tratos e abandono de animais, contribuindo, assim, com a preservação da biodiversidade e dos ecossistemas terrestres.



FIGURA 12: Objetivo global #15

FONTE: Nações Unidas Brasil (2015)

4 GERÊNCIA DO PROJETO

O *Project Management Institute* (PMI) surgiu nos Estados Unidos, no final da década de 1960, para reunir profissionais de gerenciamento de projetos que desejavam trocar experiências e estabelecer boas práticas. Desse instituto, surgiu o *Project Management Body of Knowledge* (PMBoK), um guia contendo um conjunto de práticas em gerência de projetos, que define o trabalho que deve acontecer durante todo ciclo de vida de um projeto.

Com referência ao PMBoK, neste capítulo são expostos o plano de projeto, subdividido em: partes interessadas, modelo de ciclo de vida e recursos necessários para a realização do projeto. A seção da Gerência de Projeto documenta as gestões do escopo, tempo, integração, qualidade e riscos.

No guia PMBoK é encontrada a seguinte definição para gerenciamento de projetos “é a aplicação do conhecimento, habilidades, ferramentas e técnicas às atividades do projeto para atender aos seus requisitos” (PMI, 2013, p. 5). O gerenciamento de projetos é composto por cinco grupos de processo (Iniciação, Planejamento, Execução, Monitoramento e Encerramento) representados na Figura 13, contendo todos os processos necessários para atender os requisitos de projeto (PMI, 2013).

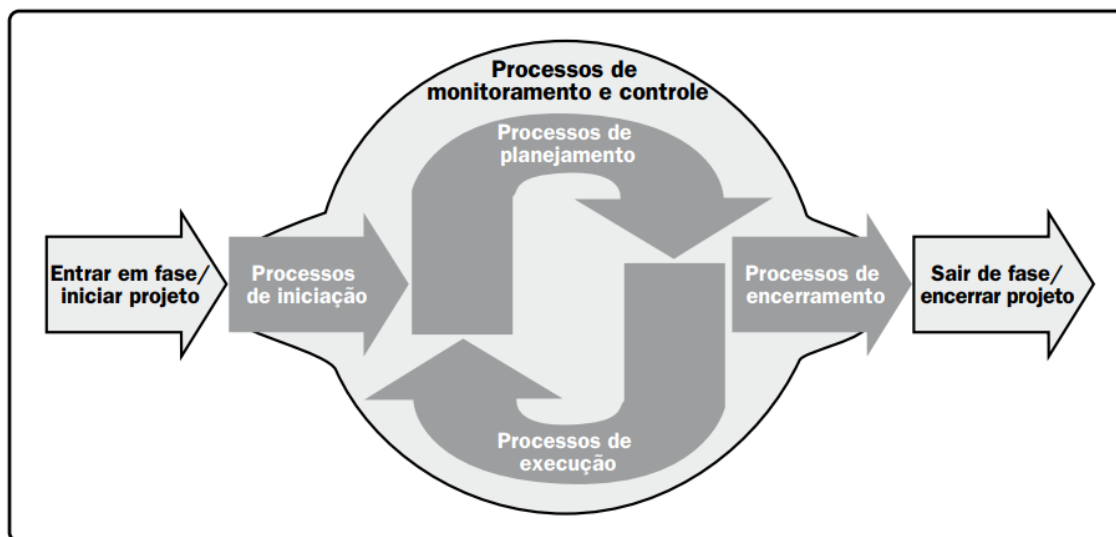


FIGURA 13: Grupo de processos de gerenciamento de projetos

FONTE: PMI (2013)

4.1 PLANO DE PROJETO

Nesta seção estão apresentadas as partes interessadas (*stakeholders*), o modelo de ciclo de vida e os recursos necessários para a realização do projeto.

4.1.1 Partes Interessadas

Segundo o guia PMBoK (PMI, 2013), partes interessadas são todas as pessoas ou organizações que podem vir a ser afetadas pelo projeto e pelas informações documentadas no mesmo.

Para demonstrar as partes afetadas neste projeto se faz uso da matriz de poder x interesse, mostrada na Figura 14.

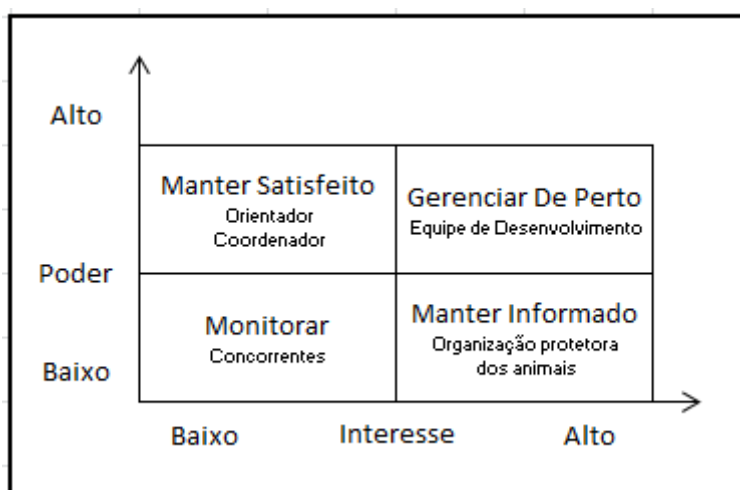


FIGURA 14: Matriz poder x interesse

FONTE: Pressman e Maxin (2016)

- a) Manter satisfeito: Eunice Gomes de Siqueira - professora orientadora do projeto;
- b) Gerencie de perto: a equipe de desenvolvimento constituída pelos integrantes, André Silva Lopes, Daniel Fernando Pereira e Taíses da Silva Mendes;
- c) Monitore: concorrentes para buscar um melhor posicionamento no desenvolvimento do projeto;
- d) Mantenha informado: instituição protetora dos animais, a fim de que o desenvolvimento do projeto seja condizente com as suas necessidades.

4.1.2 Modelo de Ciclo de Vida

Assim como é dito no PMI (2013), um projeto passa por fases, sendo separados em objetivos parciais ou funcionais, resultados e ou entregas, marcos do trabalho ou até disponibilidade financeira e tudo isso ocorre desde seu início até o término, e essas fases representam o seu ciclo de vida.

Levando isso em conta, por este projeto ter como ciclo de vida o Modelo Incremental, cada fase do projeto tem um acréscimo constante de melhorias. Assim como mostra a Figura 15.

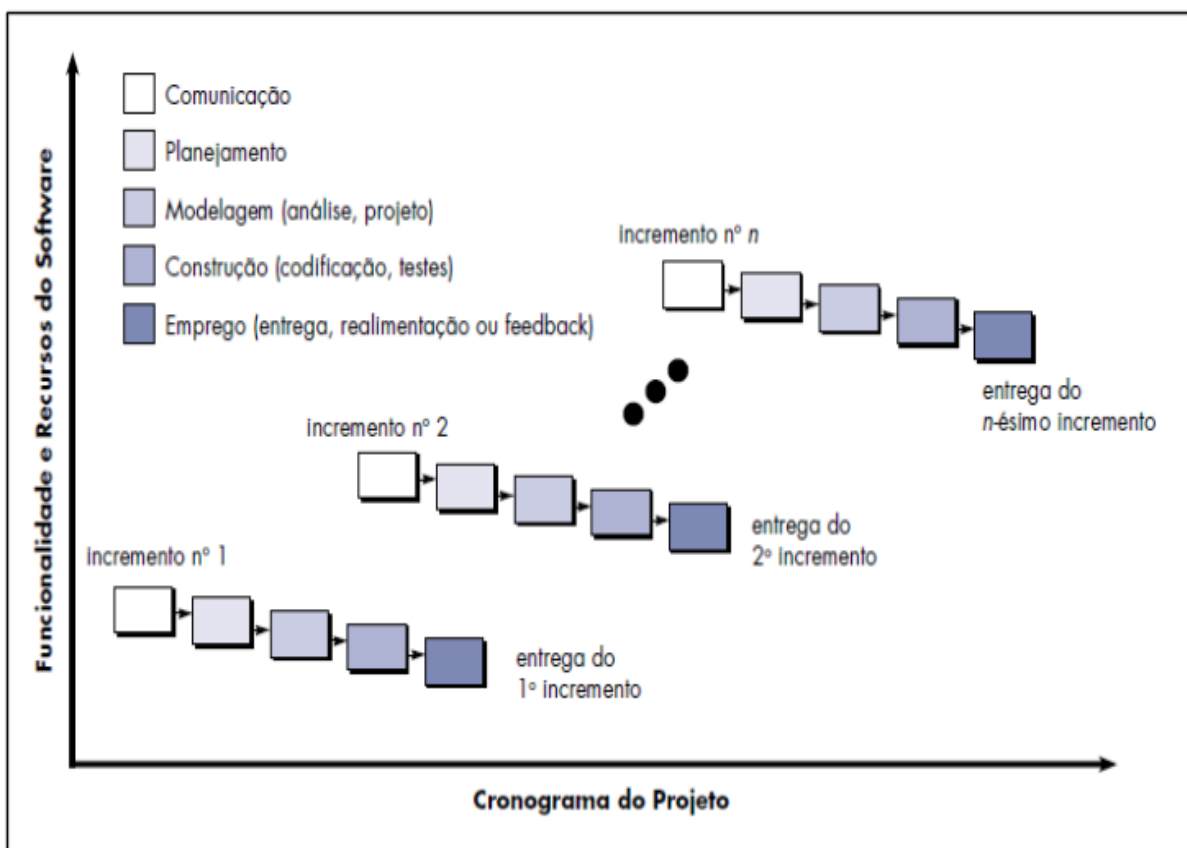


FIGURA 15: Modelo Incremental

FONTE: Pressman e Maxin (2016)

Neste projeto há quatro incrementos, as entregas estão descritas a seguir.

Entrega do primeiro incremento em 06/04/2019: contém a entrega do documento do projeto, com as seções de revisão bibliográfica; objetivos do projeto e suas seções; partes específicas da gerência do projeto, plano de projeto e as subseções: partes interessadas, modelo de ciclo de vida e recursos necessários; áreas de conhecimento e suas subseções de

gestão do escopo e do tempo; especificação e análise de requisitos e as referências bibliográficas.

Entrega do segundo incremento em 1/06/2019: contém a segunda entrega do documento do projeto, ao menos 30% dos casos de uso essenciais codificados e todos os itens da primeira entrega revisados e corrigidos. Além disso, nessa entrega está o complemento do capítulo de gerência de projeto com a gestão dos riscos, da qualidade e da integração; o complemento do Capítulo 5 com as estimativas de esforços; a elaboração do capítulo de arquitetura e projeto, contendo as seções da visão estrutural, visão comportamental, visão de dados e da interação humano-computador.

Entrega do terceiro incremento em 7/09/2019: contém a terceira versão do documento do projeto, 60% dos casos de uso essenciais codificados e todos os itens da primeira e segunda entrega revisados e devidamente corrigidos. Além disso, nesta entrega constam a introdução e o complemento do Capítulo 6.

Entrega do quarto incremento em 2/11/2019: contém a quarta versão do documento do projeto, com 100% dos casos de uso essenciais codificados e todos os itens das entregas anteriores revisados e devidamente corrigidos. Além disso, foi adicionada a seção de “Análise de Complexidade Algorítmica” e feitos os Capítulos 7, 8 e 9, que tratam respectivamente do “Plano de Testes”, “Plano da Implantação” e a “Conclusão”.

4.1.3 Recursos Necessários

Para desenvolver este projeto são necessários recursos humanos, de software e de hardware.

4.1.3.1 Recursos Humanos

Para a realização desse projeto, são necessários quatro graduandos de Sistemas de Informação, com os conhecimentos em *front-end*, *back-end* e PostgreSQL. Encarregam-se das funções de pesquisa, análise, teste e codificação, conforme se segue:

André Silva Lopes: análise, projeto e confecção deste documento;

Daniel Fernando Pereira: análise, projeto, programação e realização de testes;

Taíses da Silva Mendes: análise, projeto, programação e realização de testes.

4.1.3.2 Recursos de Software

Os recursos de software necessários para este projeto são:

- a) Sistema Operacional Windows 10 de 64 bits;
- b) Spring Tool Suite 3.9.7;
- c) IntelliJ IDEA 2019.1.3;
- d) PostgreSQL 11.2;
- e) PgAdmin 4 v4.3;
- f) Dia 0.97.2;
- g) Microsoft Excel 2013;
- h) Microsoft Project 2016;
- i) Microsoft Word 2013;
- j) WBS Tool 0.9 beta;
- k) Google Drive;
- l) Visual Studio Code 1.32.3;
- m) Ideas Modeler 11.96;
- n) Balsamiq 3.5.17;
- o) Postman 7.0.6;
- p) Git 2.21.0;
- q) Java 8
- r) Bootstrap 4.3.1;
- s) Linux Ubuntu 18.04;
- t) Angular 8.3.4.

4.1.3.3 Recursos de Hardware

Os recursos de hardware necessários para este projeto são 3 *notebooks* com as seguintes especificações:

- a) processador i5 2.5 GHz de sétima geração (ou superior);
- b) memória *Random Access Memory* (RAM) *Double-Data-Rate* (DDR4) de 8 *Gbytes*;
- c) *Hard Disk* (HD) de 1 *Terabyte* (ou superior);
- d) mouse óptico;
- e) teclado padrão Associação Brasileira de Normas Técnicas (ABNT2).

4.2 ÁREAS DE CONHECIMENTO

O objetivo da gestão de projetos é lidar com os riscos e incertezas, usando instrumentos para limitar ou contornar tais dificuldades (CARVALHO, 2012).

Segundo PMI (2013), uma área de conhecimento abrange os campos de gerenciamento de projetos, áreas especializadas e profissionais por meio de um conjunto de atividades, conceitos e termos.

Nesta seção são apresentadas as seguintes atividades: gestões do escopo, do tempo, da integração, da qualidade e dos riscos. As demais áreas de conhecimento não foram documentadas neste documento.

4.2.1 Gestão do Escopo

Gerenciamento de escopo segundo PMI (2013) é o controle daquilo que está ou não incluso no projeto, utilizando de processos que garantem a inclusão somente dos trabalhos necessários para garantir o sucesso do projeto.

4.2.1.1 Estrutura Analítica do Projeto (EAP)

Para Oliveira e Chiari (2014), uma EAP é a forma decomposta e hierárquica do escopo do projeto. A decomposição do escopo é interrompida quando o pacote de trabalho apresenta uma complexidade gerenciável.

A EAP deste projeto pode ser visualizada no Apêndice B.

4.2.1.2 Dicionário EAP

Segundo o PMI (2013), o dicionário EAP fornece informações detalhadas sobre atividades, entregas e agendamento de cada componente. Dentre alguns dados contidos nesse artefato estão os marcos do cronograma, descrição do trabalho e as estimativas de esforço.

O dicionário da EAP do projeto encontra-se no Apêndice C deste documento, descrito no campo "Notas" do arquivo elaborado com o *MS-Project*.

4.2.2 Gestão do Tempo

Gestão do tempo, segundo o PMI (2013), envolve o uso de ferramentas e técnicas associadas ao plano de gerenciamento de cronograma e um plano auxiliar e integrado que ajuda no desenvolvimento da gerência do projeto.

Esta seção trata da gerência do projeto em relação ao tempo e mostra a lista de atividades a serem feitas, o diagrama de rede, o cronograma de atividades e o quadro com resumo de esforço.

4.2.2.1 Lista de Atividades, Diagrama de Rede e Cronograma

A lista de atividades tem o objetivo de apresentar todas as atividades que se encontram nos pacotes de trabalho, incluindo, apresentação da atividade, seu início, sua duração e o responsável por cada uma delas. Um pacote é dado como concluído quando todas suas atividades forem concluídas.

No diagrama de rede são mostradas todas as atividades, organizadas cronologicamente por sua ordem de execução, além de mostrar possíveis anomalias no processo. O diagrama de rede é descrito como “uma representação gráfica das relações lógicas, também chamadas de dependências, entre as atividades do cronograma do projeto” (PMI, 2013, p.159).

O cronograma de atividades contém as datas de início e de término, o tempo consumido e o responsável pela execução de cada tarefa. São representadas no gráfico de *Gantt*, que é uma ferramenta que permite a fácil visualização das atividades previstas no projeto.

Neste projeto, a lista de atividades, o diagrama de rede e o cronograma podem ser visualizados no Apêndice C.

4.2.2.2 Quadro com Resumo de Esforços

O quadro com resumo de esforço apresenta um comparativo do esforço estimado pela equipe com o esforço efetivamente realizado. O Quadro 2 mostra a comparação entre o esforço estimado e o efetivamente realizado em cada fase do projeto.

	Fase 1	Fase 2	Fase 3	Fase 4
Estimativa <i>ad-hoc</i>	230/230h	480/710h	270/980h	250/1230h
Dicionário EAP	212/212h	452/664h	260/924h	238/1162h
Efetivo realizado	162/162h	295/457h	158/615h	179/794h

QUADRO 2: Esforço estimado x realizado

Na Fase 1, buscou-se deixar certa margem de folga entre as atividades, uma vez que se considerou a falta de experiência da equipe com relação à gestão do projeto, um risco técnico negativo para sua realização. Ao final da Fase 1 foram previstas dificuldades, mas na prática foram superadas, porém o esforço realizado ficou abaixo do planejado de forma *ad-hoc* e no dicionário EAP.

Na Fase 2, novamente, buscou-se deixar uma margem de folga entre as atividades, assim como na Fase 1, pensando em contornar possíveis imprevistos que poderiam vir a acontecer. Então, mais uma vez o esforço realizado ficou abaixo da estimativa *ad-hoc* e no dicionário EAP.

Na Fase 3, houve um imprevisto, que foi o abandono de um dos integrantes da equipe, mas mesmo assim, continuou-se seguindo o padrão de deixar uma folga nas atividades assim como nas Fases 1 e 2. Portanto, mais uma vez o esforço realizado ficou abaixo da estimativa *ad-hoc* e no dicionário EAP.

Na Fase 4, os esforços foram por conta de fazer com que o projeto fosse apresentado na feira, sendo assim a maior parte do esforço durante o tempo disponível foi voltado à codificação do sistema.

4.2.3 Gestão da Integração

A gestão da integração é uma etapa indispensável, porque nela são feitas escolhas a respeito da alocação de recursos, concessões e resolução de conflitos referentes às demais áreas de conhecimento que estão presentes num projeto.

A gestão da integração orienta e consolida toda execução do projeto, conforme explicado a seguir:

O gerenciamento da integração do projeto inclui os processos e atividades para identificar, definir, combinar, unificar e coordenar os vários processos e atividades dentro dos grupos de processos de gerenciamento do projeto. No contexto de gerenciamento de projetos, integração inclui características de unificação, consolidação, comunicação e ações integradoras que são essenciais para execução controlada do projeto até sua conclusão, a fim de gerenciar com sucesso as expectativas das partes interessadas e atender aos requisitos (PMI, 2013, p. 63).

A gestão da integração, segundo Valeriano (2005), é responsável pela realização das atividades necessárias para harmonia do trabalho, combinando e coordenando todas as partes envolvidas no processo.

Esta seção apresenta o monitoramento, o controle de configuração e os controles de mudança deste presente projeto.

4.2.3.1 Monitoramento

O monitoramento é um processo que visa acompanhar o andamento das atividades do projeto, ajudando a encontrar ações corretivas ou preventivas para prováveis impedimentos. Sendo assim, o monitoramento deve acontecer em todas as fases do projeto, fornecendo uma melhor compreensão e identificando áreas específicas que necessitam de auxílio.

O monitoramento deste projeto é realizado por meio de relatórios de desempenho e podem ser consultados no Apêndice D.

4.2.3.2 Controle de Configuração

O controle de configuração é parte essencial de um projeto, pois ajuda a manter a integridade dos artefatos produzidos revisando todas as solicitações de mudança de forma que não permita mudanças negativas ou que possam atrapalhar a entrega de uma fase, sendo independente de tamanho ou da equipe envolvida. Um controle de configuração bem feito visa proporcionar que a equipe trabalhe com as versões mais atualizadas dos artefatos.

As versões deste documento são controladas conforme a seguinte nomenclatura:

PFC_NomeDoProjeto_vs

Sendo que:

- a) PFC: identifica o projeto como sendo Projeto Final de Curso;
- b) NomeDoProjeto: identifica o nome do projeto, que é *Help a Pet*.
- c) vs: versão identificada por dois números inteiros, o primeiro indica a fase do projeto e o segundo número indica o acréscimo de novas particularidades.

Exemplo: PFC_*Help_a_Pet*_vs1.0

Para auxiliar no controle de versão da documentação deste projeto, optou-se pelo serviço *Google Drive*. Esse serviço oferece 15 *Gigabytes* de armazenamento em nuvem, disponível de forma gratuita; histórico de revisões onde é possível verificar quem alterou o quê e quando; pasta sincronizada com o computador de cada um da equipe; e compartilhamento de arquivos online que permite que simultaneamente vários integrantes da equipe acessem e editem o mesmo arquivo.

Para controlar a integridade do documento e de seus artefatos, somente o gerente de projeto Lucas Soares possui permissão para alterar os arquivos armazenados em nuvem, impedindo assim mudanças não autorizadas.

Na Figura 16 é mostrado o uso do Google Drive para o compartilhamento de arquivos online.

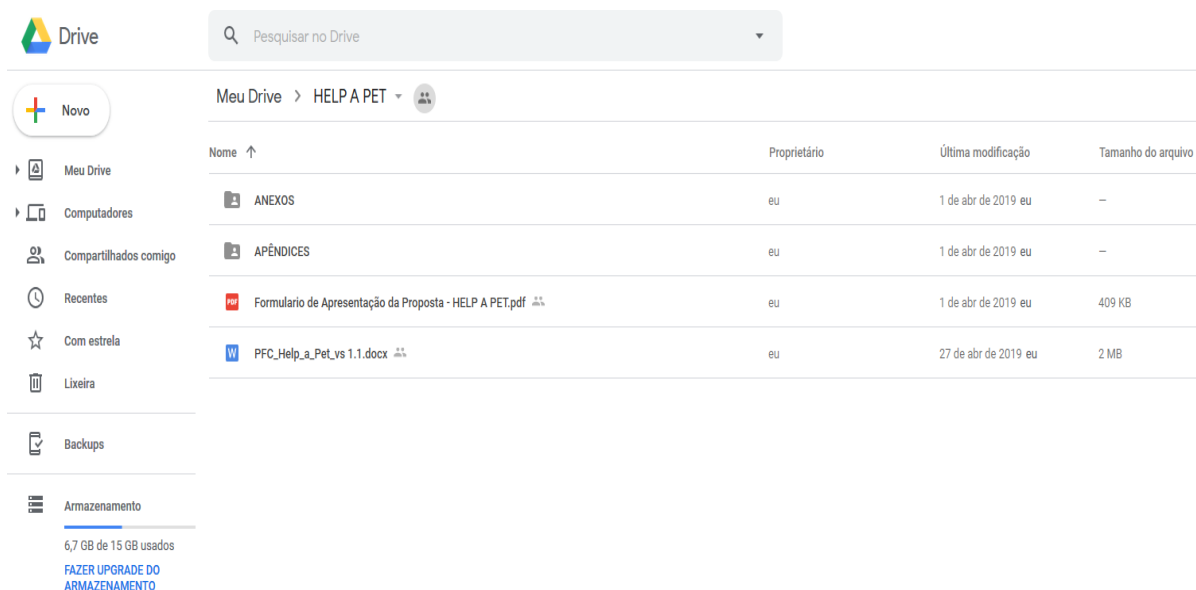


FIGURA 16: Armazenamento do projeto *Help a Pet* usando *Google Drive*

Já para o controle de versões dos códigos-fonte é utilizado o serviço do *GitHub*, que é um repositório que implementa o padrão de versionamento *Git*. Uma das opções do *GitHub* é a criação de repositórios privados, gratuitos, com limite de até três desenvolvedores por projeto, sendo ideal para projetos com portes menores e certo nível de sigilo envolvido. Cada integrante da equipe também possui seu espaço de desenvolvimento próprio, chamado de *branches*, o que possibilita que o desenvolvimento seja feito paralelamente. Então, um membro da equipe é responsável por fazer a integração com o *branch* principal, sendo também responsável pela resolução de possíveis conflitos de integração que venham a surgir.

No *GitHub* a equipe também tem acesso a um painel *Kanban*, uma ferramenta criada por Taiichi Ohno, da *Toyota*. *Kanban* é uma forma simples de gestão à vista que fixa cartões de tarefas em um quadro ou mural, possibilitando que as pessoas possam verificar rapidamente a situação do trabalho.

Para o controle de versionamento, todos da equipe *Help a Pet* podem modificar os códigos-fonte, desde que aja uma requisição de mudança e ela seja autorizado pelo Comitê de Controle de Mudanças do projeto.

Na Figura 15 é mostrado o uso do *GitHub* para controle de versionamento do código fonte do projeto.

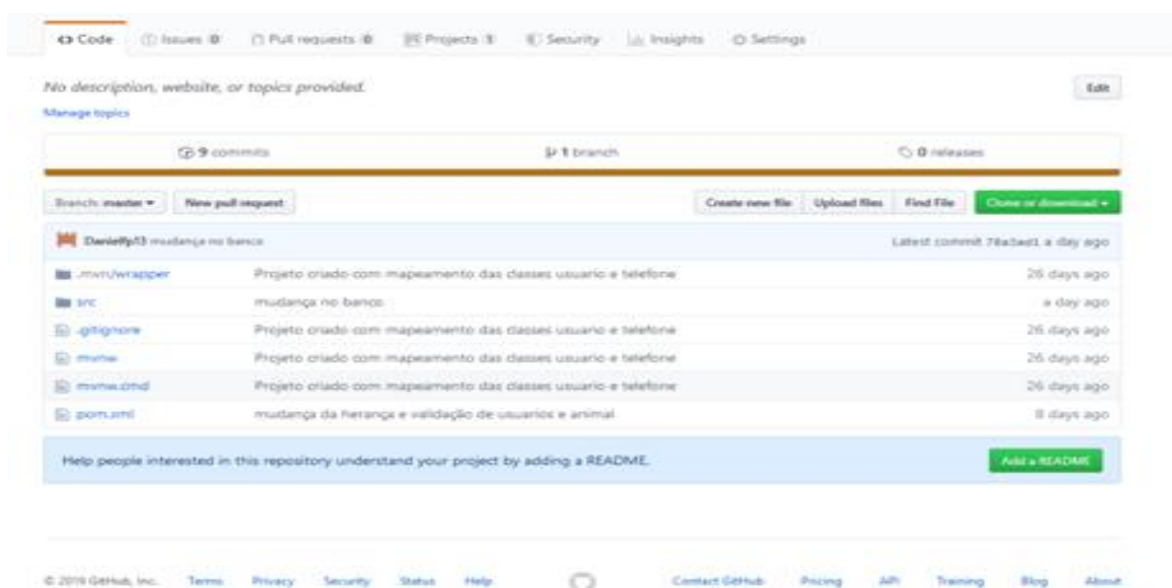


FIGURA 17: Controle de versão do projeto *Help a Pet* usando *Github*

4.2.3.2 Controle das Mudanças no Escopo

Controle das mudanças no escopo deve ser transparente à equipe, organizado e padronizado, também descrito como:

Controle integrado de mudanças é o processo de revisar todas as solicitações de mudança, aprovar as mudanças e gerenciar as mudanças sendo feitas nas entregas, ativos de processos organizacionais, documentos do projeto e no plano de gerenciamento do projeto, e comunicar a disposição dos mesmos (PMI, 2013, p. 94).

Deve-se manter um histórico de atualizações que consta todas as modificações que foram realizadas em relação à versão anterior.

No projeto *Help a Pet* registra-se todo histórico de atualizações e mudanças, no qual constam as modificações realizadas em relação à versão anterior. Toda e qualquer mudança no escopo do projeto deve ser previamente solicitada ao Comitê de Controle de Mudanças, formado por André Silva Lopes, Daniel Fernando Pereira e Taíses da Silva Mendes. Essas mudanças devem ser discutidas de forma verbal e considerar a importância, o prazo disponível e o tempo necessário para sua realização, definindo assim a sua aprovação ou rejeição.

As principais solicitações de mudanças, com suas respectivas situações encontram-se no Quadro 3.

Solicitação de mudança	Descrição	Data	Situação
Adição de nova tecnologia para desenvolvimento do <i>front-end</i> .	Sugerida a utilização do <i>framework</i> Angular para desenvolvimento do <i>front-end</i> (referente ao RNF 09 e RNF 10).	05/05/2019	Aprovada pelo Comitê de Mudanças
Mudança da tecnologia para desenvolvimento do <i>front-end</i> .	Foi alterada para a utilização do <i>Bootstrap</i> para desenvolvimento do <i>front-end</i> .	30/08/2019	Aprovada pelo Comitê de Mudanças
Cancelamento da mudança tecnologia no desenvolvimento do <i>front-end</i> .	Foi cancelado por conta do <i>back-end</i> ter sido desenvolvido para ser acesso via chamadas da API REST. O Angular voltou a ser utilizado.	23/09/2019	Aprovada pelo Comitê de Mudanças.

QUADRO 3: Propostas de mudanças para o projeto *Help a Pet*

4.2.4 Gestão da Qualidade

Para satisfazer o objetivo proposto ao um projeto de forma positiva, são necessárias políticas de qualidade específicas a cada tipo de entrega, conforme explicado:

O gerenciamento da qualidade do projeto aborda o gerenciamento do projeto e suas entregas. Ele se aplica a todos os projetos, independentemente da natureza das suas entregas. As medidas e técnicas de qualidade são específicas do tipo de entrega produzida pelo projeto. Por exemplo, o gerenciamento da qualidade das entregas de software pode usar abordagens e medidas diferentes das utilizadas na construção de uma usina nuclear. Nos dois casos, deixar de cumprir os requisitos pode ter consequências negativas e graves para uma ou todas as partes interessadas do projeto (PMI, 2013, p. 227).

Esta seção apresenta detalhadamente toda a atividade de gestão da qualidade aplicada a este projeto.

Grande parte da gestão de qualidade é responsabilidade do gerente de projeto, Lucas Soares de Oliveira que deve ter conhecimento de todo escopo, as normas a serem seguidas, controlar e monitorar o andamento do projeto, certificando-se de que os requisitos estejam sendo atendidos, assim como as necessidades das partes envolvidas e que os prazos estejam sendo respeitados. A gestão da qualidade também deve ser aplicada por todos os integrantes da equipe.

Durante todo projeto, a equipe deve adotar as seguintes diretrizes de qualidade:

- a) para o documento deve utilizar as diretrizes para elaboração de trabalhos científicos da FAI, que é uma adaptação das normas da Associação Brasileira de Normas Técnicas (ABNT);
- b) na modelagem de artefatos específicos deve ser feita de acordo com o padrão *Unified Modeling Language* (UML);
- c) deve usar as seguintes ferramentas para controle de versão: *Google Drive*, *Git* e *Github*;
- d) a codificação deve seguir as convenções definidas pela *Java Code Conventions*.

No Apêndice E, é encontrado o quadro com critérios de qualidade preenchidos. Nele se encontram todos os itens levados em consideração para obter a qualidade deste documento.

4.2.5 Gestão dos Riscos

Esta seção apresenta os riscos do projeto que foram identificados pela equipe durante a sua realização. Dessa forma, com o uso de técnicas específicas, torna-se possível aumentar probabilidade e o impacto positivo e diminuir a probabilidade e o impacto negativo no projeto.

PMI (2013, p. 309) explica que:

O gerenciamento dos riscos do projeto inclui os processos de planejamento, identificação, análise, planejamento de respostas e controle de riscos de um projeto. Os objetivos do gerenciamento dos riscos do projeto são aumentar a probabilidade e o impacto dos eventos positivos e reduzir a probabilidade e o impacto dos eventos negativos no projeto.

Os riscos do projeto são eventos ou condições incertas, podendo ser positivos ou negativos e podem ou não acontecer. Portanto, deve-se gerenciá-los de forma que em caso de riscos positivos, eles devem ser explorados ou melhorados e, ao mesmo tempo, no caso de riscos negativos, eles devem ser mitigados ou prevenidos.

O PMI (2013, p. 309) cita seis processos para o gerenciamento de riscos, que são:

- a) planejar o gerenciamento dos riscos - o processo de definição de como conduzir as atividades de gerenciamento dos riscos de um projeto;
- b) identificar os riscos - o processo de determinação dos riscos que podem afetar o projeto e de documentação das suas características;
- c) realizar a análise qualitativa dos riscos - o processo de priorização de riscos para análise ou ação posterior com a avaliação e combinação de sua probabilidade de ocorrência e impacto;
- d) realizar a análise quantitativa dos riscos - o processo de analisar numericamente o efeito dos riscos identificados nos objetivos gerais do projeto;
- e) planejar as respostas aos riscos - o processo de desenvolvimento de opções e ações para aumentar as oportunidades e reduzir as ameaças aos objetivos do projeto;
- f) controlar os riscos – o processo de implementar planos de respostas aos riscos, acompanhar os riscos identificados, monitorar riscos residuais, identificar novos riscos e avaliar a eficácia do processo de gerenciamento dos riscos durante todo o projeto.

O PMI (2013, p. 344 - 345) ainda aborda quatro estratégias para riscos negativos ou ameaças e quatro estratégias para riscos positivos ou oportunidades, apresentadas a seguir:

Estratégia para riscos negativos ou ameaças:

- a) prevenir - a prevenção de riscos é uma estratégia de resposta ao risco em que a equipe do projeto age para eliminar a ameaça ou proteger o projeto contra o seu impacto. Ela envolve a alteração do plano de gerenciamento do projeto para eliminar totalmente a ameaça;

- b) transferir - a transferência de riscos é uma estratégia de resposta ao risco em que a equipe do projeto transfere o impacto de uma ameaça para terceiros, juntamente com a responsabilidade pela sua resposta. Transferir o risco simplesmente passa a responsabilidade de gerenciamento para outra parte, mas não o elimina;
- c) mitigar – a mitigação de riscos é uma estratégia de resposta ao risco em que a equipe do projeto age para reduzir a probabilidade de ocorrência ou impacto do risco. Ela implica na redução da probabilidade e/ou do impacto de um evento de risco adverso para dentro de limites aceitáveis;
- d) aceitar - a aceitação de risco é uma estratégia de resposta pela qual a equipe do projeto decide reconhecer a existência do risco e não agir, a menos que o risco ocorra. Essa estratégia é adotada quando não é possível ou econômico abordar um risco específico de qualquer outra forma. Essa estratégia indica que a equipe do projeto decidiu não alterar o plano de gerenciamento do projeto para lidar com um risco, ou não conseguiu identificar outra estratégia de resposta adequada.

Estratégia para riscos positivos ou oportunidades:

- a) explorar - a estratégia explorar pode ser selecionada para riscos com impactos positivos quando a organização deseja garantir que a oportunidade seja concretizada. Essa estratégia procura eliminar a incerteza associada com um determinado risco positivo, garantindo que a oportunidade realmente aconteça;
- b) melhorar - a melhoria é usada para aumentar a probabilidade e/ou os impactos positivos de uma oportunidade. Identificar e maximizar os principais impulsionadores desses riscos de impacto positivo pode aumentar a probabilidade de ocorrência;
- c) compartilhar - compartilhar um risco positivo envolve a alocação integral ou parcial da responsabilidade da oportunidade a um terceiro que tenha mais capacidade de explorar a oportunidade para benefício do projeto;
- d) aceitar - aceitar uma oportunidade é estar disposto a aproveitá-la caso ela ocorra, mas não persegui-la ativamente.

Esses dados devem ser coletados progressivamente pela equipe durante a execução das fases do projeto. A planilha de acompanhamento consta no Apêndice F deste documento.

5 ESPECIFICAÇÃO E ANÁLISE DOS REQUISITOS

Neste capítulo são apresentados os requisitos do projeto *Help a Pet*, com as visões funcionais e dos dados do sistema de software, em nível conceitual (ou de análise).

5.1 REQUISITOS DO SISTEMA DE SOFTWARE

Nesta seção são apresentados os requisitos funcionais e não funcionais do sistema. Respeitando a classificação que Sommerville (2007) fez, requisitos funcionais definem a forma no qual o sistema responderá em determinadas situações de acordo com os dados inseridos. Já os requisitos não funcionais definem as restrições do sistema, que podem ser várias como de tempo de execução, segurança, usabilidade, dentre outros.

5.1.1 Requisitos Funcionais (RF)

Nesta seção são apresentados os seguintes módulos do sistema e seus respectivos requisitos: módulo de autenticação, módulo de pedido de adoção, módulo de administração e módulo de relatório.

- a) Módulo de autenticação
 - 1. Cadastrar adotante;
 - 2. Cadastrar instituição;
 - 3. Autenticar usuário;
 - 4. Recuperar senha;
 - 5. Alterar senha;
 - 6. Aceitar termo de uso;
- b) Módulo de pedido de adoção
 - 7. Manter cadastro de animais;
 - 8. Realizar busca de animais;
 - 9. Realizar pedido de adoção;
 - 10. Gerar termo de adoção;
- c) Módulo de administração
 - 11. Visualizar cadastro de instituição;
 - 12. Liberar cadastro de instituição;

d) Módulo de relatórios

13. Gerar relatório de instituições cadastradas;
14. Gerar relatório de animais cadastrados;
15. Gerar relatório de adoções realizadas.

a) Módulo de autenticação

RF 01 – Cadastrar adotante

Esse requisito refere-se ao cadastro de usuários que desejam ser adotantes. Esse cadastro deverá ser realizado pelos próprios interessados, quando entrarem no sistema ou se preferirem quando for adotar um animal.

Para se cadastrar no sistema, o adotante deverá informar os seguintes dados:

(1) nome, (3) sobrenome, (3) Cadastro de Pessoa Física (CPF), (4) Registro Geral (RG), (5) logradouro, (6) bairro, (7) número da residência, (8) município, (9) UF, (10) Código de Endereçamento Postal (CEP), (11) e-mail válido, (12) senha, (13) telefones para contato e (14) data de nascimento.

O sistema deverá assegurar que apenas usuários com mais de 18 anos de idade possam se cadastrar e que, somente após a confirmação do cadastro, o usuário poderá adotar um animal.

Prioridade: essencial.

RF 02 – Cadastrar instituição

Esse requisito refere-se ao cadastro de instituição no sistema, que deverá ser realizado pelo responsável pela entidade protetora dos animais.

Para se cadastrar no sistema, a instituição deverá informar os seguintes dados:

(1) nome, (2) e-mail válido, (3) senha, (4) município, (5) CEP, (6) UF, (7) bairro, (8) logradouro, (9) número da residência, (10) telefones para contato e (11) CNPJ.

O cadastro desse tipo de usuário só será confirmado após a verificação e validação dos dados informados pela instituição.

A liberação do cadastro de instituição será realizada por um administrador do sistema (RF12), que já deverá estar previamente cadastrado no sistema.

Prioridade: essencial.

RF 03 - Autenticar usuário

Esse requisito refere-se ao modo como um usuário já cadastrado na aplicação irá realizar sua identificação. O usuário deverá fornecer (1) e-mail e (2) senha para ter acesso às funcionalidades que lhe são autorizadas.

Prioridade: essencial.

RF 04 - Recuperar senha

Esse requisito refere-se à possibilidade de que um usuário do sistema possa recuperar sua senha de autenticação, caso a tenha perdido. Para realizar a operação de recuperação de senha, o usuário já deverá estar previamente cadastrado no sistema e fornecer o seu e-mail de autenticação. O sistema fará um novo registro de senha através de um *link* que será enviado para o e-mail desse usuário, previamente, cadastrado no sistema.

Prioridade: essencial.

RF 05 - Alterar senha

O sistema deverá fornecer uma opção para que um usuário possa alterar sua senha de autenticação, quando achar conveniente.

Para realizar a operação de alteração de senha, o usuário já deverá estar previamente cadastrado no sistema e fornecer os seguintes dados: senha atual e nova senha.

Prioridade: essencial.

RF 06- Aceitar termo de uso

O sistema deverá fornecer um termo de uso para os usuários do sistema. Trata-se de um documento que traz em seu texto as condições e regras para que um usuário do sistema possa utilizar o serviço que está sendo oferecido.

O sistema apresentará um termo de uso diferente para cada tipo de usuário, e os usuários só poderão se cadastrar no sistema se concordarem com o termo de uso.

Prioridade: essencial.

b) Módulo de pedido de adoção**RF 07 – Manter cadastro de animais**

Esse requisito estabelece como a instituição, previamente cadastrada no sistema, deverá realizar o cadastro dos animais pertencentes a ela e que estão disponíveis para adoção. A instituição deverá informar os seguintes dados sobre o animal: (1) nome do animal, (2) raça, (3) ano de nascimento estimado, (4) espécie, (5) gênero, (6) porte, (7) fotos, (8) se já foi esterilizado, (9) se já foi vacinado, (10) se já foi vermifugado, (11) descrição do animal, (12) cor e (13) data de recolhimento no abrigo pela instituição.

O sistema deverá oferecer a opção para que as instituições que já tiverem animais cadastrados possam alterar os dados dos animais ou excluí-los.

Prioridade: essencial.

RF 08 – Realizar busca de animal

Este requisito refere-se à forma dos usuários visualizarem os dados dos animais que estarão disponíveis para adoção.

Para facilitar a busca de animais, o sistema deverá oferecer filtros de pesquisa de localização como município, UF e bairro.

E também filtros sobre os dados dos animais, como raça, idade, espécie, gênero, cor, porte, se já foi esterilizado, se foi vacinado, dentre outros parâmetros complementares.

Prioridade: essencial.

RF 09 – Realizar pedido de adoção

Esse requisito refere-se à maneira pela qual um adotante poderá realizar a adoção de um animal. Ao escolher um animal para adoção, o adotante deverá fazer um pedido de adoção, que será registrado pelo sistema. O sistema deverá guardar os seguintes dados (1) data de início, (2) data de encerramento e (3) situação do processo de adoção.

Prioridade: essencial.

RF 10 – Gerar termo de adoção

O sistema deverá fornecer a opção para que a instituição possa gerar um termo de adoção que deverá ser assinado pelo adotante e pelo responsável pela adoção.

O termo de adoção deverá conter os seguintes dados:

- a) do adotante (1) nome, (2) CPF, (3) RG, (4) endereço, (5) bairro, (6)nº, (7) município, (8) UF, (9) CEP, (10) e-mail, (11) telefones para contato, (12) se possui animal, (13) e quantos.
- b) do animal: (1) nome do animal, (2) raça, (3) data de nascimento, (4) espécie, (5) gênero, (6) porte, (7) esterilizado, (8) vacinado, (9) vermifugado;
- c) E os seguintes termos:
 - 1. “Ao adotar o animal acima descrito declaro-me apto para assumir a guarda e a responsabilidade sobre este animal, eximindo o doador de toda e qualquer responsabilidade por quaisquer atos praticados pelo animal a partir desta data.”
 - 2. “Declaro ainda estar ciente de todos os cuidados que este animal exige no que se refere à sua guarda e manutenção, além de conhecer todos os riscos inerentes à espécie e raça no convívio com humanos, estando apto a guardá-lo e vigiá-lo, comprometendo-me a proporcionar boas condições de alojamento e alimentação, assim como, espaço físico e/ou atividades regulares que possibilite o animal se exercitar”.

3. “Responsabilizo-me por preservar a saúde e integridade do animal e a submetê-lo aos cuidados médico-veterinários sempre que necessário para este fim”.
4. “Comprometo-me a não transmitir a posse deste animal a outrem sem o conhecimento do doador”.
5. “Comprometo-me a permitir o acesso do doador ao local onde se encontra o animal para visitas e averiguação de suas condições, podendo ser pré-agendadas ou não”.
6. “Tenho conhecimento de que caso seja constatada alguma situação inadequada para o bem-estar do animal ou descumprimento das normas, perderei a sua guarda, ficando ainda sujeito às penas previstas na legislação vigente. Declaro-me assim, ciente das normas acima, as quais aceito e assino o presente Termo de Responsabilidade, assumindo plenamente os deveres que dele constam, bem como outros relacionados à posse responsável e que não estejam incluídos neste Termo”.

Prioridade: essencial.

c) Módulo de administração

RF 11 – Visualizar cadastro de instituição

O sistema deverá notificar o administrador de que uma instituição realizou um cadastro no sistema e também deverá prover acesso aos dados informados pela instituição para que o administrador possa validar esses dados e liberar o cadastro da instituição a fim de que ela possa utilizá-lo.

Prioridade: essencial.

RF 12 – Liberar cadastro de instituição

Esse requisito refere-se à verificação e validação dos dados enviados pela instituição no (RF02), caso os dados fornecidos pela instituição sejam válidos, o administrador do sistema

deverá liberar o cadastro para a instituição e o sistema enviará um e-mail para a instituição informando que o cadastro foi autorizado.

Prioridade: essencial.

d) Módulo de relatórios

RF 13 – Gerar relatório de instituições cadastradas

Esse requisito refere-se ao modo de como o administrador poderá gerar um relatório das instituições cadastradas no sistema. Para gerar esse relatório será possível realizar filtragem por UF, município, data de cadastro e situação do cadastro.

Prioridade: importante.

RF 14 – Gerar relatório de animais cadastrados

Esse requisito refere-se ao modo de como a instituição poderá gerar um relatório de todos os seus animais cadastrados no sistema. Para gerar esse relatório será possível realizar filtragem por espécie, raça, porte, cor, gênero, nome, se já foi adotado, e idade estimada.

Prioridade: importante.

RF 15 – Gerar relatório de adoções realizadas

Esse requisito refere-se ao modo de como a instituição poderá gerar um relatório de todas as adoções realizadas por ela em determinado período. Para gerar esse relatório será possível realizar filtragem por município, espécie, raça, porte, cor, gênero, data de início e data de término.

Prioridade: importante.

5.1.2 Requisito não Funcionais (RNF)

Os requisitos não funcionais são propriedades emergentes que o sistema deve atender, como acessibilidade, segurança, confiabilidade e tempo de resposta. Esses requisitos não estão relacionados diretamente com as funcionalidades do sistema, mas o afetam de

forma eminente.

Os requisitos não funcionais trazem consigo o conceito de qualidade, sendo assim, quanto mais requisitos não funcionais forem atendidos, maior será a qualidade do sistema de software.

5.1.2.1 Requisitos de Produto

a) Requisitos de Confiabilidade

RNF 01 – Disponibilidade 24 X 7 para servidores de aplicação

A aplicação *Web* deverá estar disponível ao usuário 24h por dia em sete dias da semana. A disponibilidade requerida será de 99,98% do tempo ao longo do ano.

Prioridade: essencial.

RNF 02 – Confiabilidade dos dados

O sistema deverá aplicar verificações nos valores de dados fornecidos pelos usuários, de modo a evitar dados inconsistentes e manter a integridade dos dados mantidos.

Prioridade: essencial.

b) Requisitos de Eficiência

RNF 03 – Recuperação do servidor

Em caso de falha, o servidor deverá ser restaurado e estar disponível em até 24h após a parada.

Prioridade: essencial.

c) Requisitos de Portabilidade

RNF 04 – Navegadores web

A aplicação *Help a Pet* deverá ser manipulada por meio dos seguintes navegadores: *Google Chrome* versão 73.0.3683.103; *Firefox* versão 66.0.2 ou *Microsoft Edge* versão 44.17763.1.0.

Prioridade: essencial.

d) Requisitos de Usabilidade**RNF 05 – *Design* responsivo nas interfaces gráficas**

A aplicação utilizará um *design* responsivo, desta forma a interface de usuário poderá ser vista de diversas formas e contextos (dispositivos móveis – *smartphones* e *tablets*, *notebooks*, *desktops*). O *design* responsivo deve exibir com qualidade e comportamento adequado nos mais variados tamanhos de telas dos dispositivos de acesso.

Prioridade: essencial.

RNF 06 – Interface de usuário simplificada

A aplicação deverá ter uma interface de usuário de fácil entendimento, para que sua aprendizagem seja rápida e sua experiência seja agradável durante a navegação. A aplicação deve apresentar orientações para uso e os campos de entrada devem ser demarcados com faixas de valores esperadas.

Prioridade: essencial.

e) Requisitos de Acessibilidade**RNF 07 – Interface para usuários com deficiência**

A interface com o usuário deve ser adequada com adaptações e personalizações que permitam sua utilização por usuários com necessidades especiais (por exemplo, identificação de cores em textos de páginas para daltônicos), seguindo as orientações específicas de acessibilidade de interfaces de usuário.

Prioridade: essencial.

5.1.2.2 Requisitos Organizacionais**a) Requisitos de Entrega****RNF 08 – Dados de entrega**

Ao longo do projeto deverão ser realizadas quatro entregas do sistema, conforme previsto no Capítulo 4. A versão beta do sistema deverá ficar pronta para apresentação na Feira de Tecnologia da FAI (Faitec), realizada em outubro de 2019.

Prioridade: essencial.

b) Requisitos de implementação

RNF 09 – Linguagens de programação

A componente servidora (*back-end*) será desenvolvida em Java utilizando *Spring Model View Controller* (MVC) e a componente *Web* será desenvolvida utilizando *Typescript*, *Javascript*, *HyperText Markup Language* (HTML) e *Cascading Style Sheets* (CSS).

Prioridade: essencial.

RNF 10 – Frameworks

A componente cliente (*front-end*) será implementada utilizando *Angular* versão 8.

Prioridade: essencial.

RNF 11 – Ambiente de Desenvolvimento

A aplicação deverá ser implementada utilizando as *Integrated Development Environment* (IDEs) *Visual Code* para desenvolvimento *front-end*, a IDE do *SpringBoot* para desenvolvimento com a Linguagem Java e o *PostgreSQL* para Sistema de Gerenciamento de Banco de Dados (SGBD).

Prioridade: essencial.

RNF 12 – Ferramentas *Computer-Aided Software Engineering* (Case)

Devem ser utilizados o *Github* para controle de versão, *SpringBoot* como IDE de desenvolvimento, *Ideas Modeler* para modelagem de dados, caso de uso e diagrama de classes, *Balsamiq Mockups* para prototipação e criação de modelos de sistemas *Web*, *Dia* para a criação do Diagrama Entidade e Relacionamento (DER).

Prioridade: essencial.

c) Requisitos de Padrões

RNF 13 – Padrão de Documentação

O documento técnico deve seguir as normas para trabalhos científicos da ABNT e Diretrizes para Elaboração de Trabalhos Acadêmicos da FAI.

Prioridade: essencial.

RNF 14 – Padrões de codificação

O padrão de código fonte seguirá as convenções propostas pela *Java Code Conventions* para a linguagem *Java*.

Prioridade: essencial.

RNF 15 – Padrão *Model View Controller* (MVC)

O padrão da arquitetura do sistema deverá ser a MVC, que tem como objetivo separar a aplicação em três camadas: camada de controle (*controller*), camada de interação (*view*) e camada de manipulação de dados (*model*).

Prioridade: essencial.

5.1.2.3 Requisitos Externos

a) Requisitos de Interoperabilidade

RNF 16 – Hardware da aplicação servidora e do banco de dados

O servidor deverá ter no mínimo as seguintes configurações: processador i7, 8 *Gbytes* de *RAM* e 1 *Tbytes* de HD.

Prioridade: essencial.

RNF 17 – Sistema Operacional para aplicação servidora

O sistema operacional deverá ser *Linux Ubuntu* 16.04.2 para as aplicações servidoras.

Prioridade: essencial.

RNF 18 – Sistema Gerenciador de Banco de Dados (SGBD)

O SGBD deverá ser PostgreSQL versão 11.

Prioridade: essencial.

RNF 19 – *Container* de aplicação

Como *container* de aplicação será usado o Apache Tomcat versão 9.

Prioridade: essencial.

RNF 20 – Servidor de e-mail

O servidor de e-mails *Simple Mail Transfer Protocol* (SMTP) do *Google Gmail* deverá ser usado para envio de mensagens (mensagem de boas-vindas e redefinição de senhas).

Prioridade: essencial.

b) Requisitos Éticos**RNF 21 – Confidencialidade de documentos**

Não poderão ser expostos ao público os dados particulares enviados pelo usuário para validação.

Prioridade: essencial.

c) Requisitos de Privacidade e Segurança**RNF 22 – Segurança no tráfego de dados sigilosos**

Os dados sigilosos devem estar protegidos por um mecanismo de criptografia para garantir a integridade e deve-se assegurar sua persistência de forma correta, usando *Hyper Text Transfer Protocol Secure* (HTTPS) como protocolo de comunicação.

Prioridade: essencial.

RNF 23 – Autenticação e controle de acesso

Somente usuários autenticados no sistema terão acesso aos módulos de acesso privado. Deverão fornecer seu e-mail e senha para autenticação e acesso ao sistema.

Prioridade: essencial.

RNF 24 – Privacidade e acesso aos dados

Dados privados do usuário (e-mail e senha) deverão ser mantidos em sigilo visando maior privacidade dentro da aplicação. Somente o administrador *master* do SGBD terá total acesso e permissão para alterar a estrutura do banco de dados.

Prioridade: essencial.

RNF 25 – Política de senhas

O sistema deverá aceitar senhas que atendam às seguintes regras: ter no mínimo 8 caracteres, ter no máximo 20 caracteres, uso de letras maiúsculas e minúsculas, números ou caracteres especiais intercalados na senha.

Prioridade: essencial.

RNF 26 – Termo de uso da aplicação

O usuário (adotante ou instituição) somente poderá ter sua conta ativa se aceitar os termos de uso com as condições e as regras de uso da aplicação. O usuário (instituição) deverá enviar documentações via e-mail para ganhar acesso à aplicação.

Prioridade: essencial.

5.2 ANÁLISE DOS REQUISITOS

Para a análise dos requisitos são utilizados o modelo de casos de uso da UML e o Modelo Entidade e Relacionamento (MER). Esses modelos fornecem bases conceituais para a abstração de informações relevantes sobre o sistema.

5.2.1 Modelo de Casos de Uso

O modelo de caso de uso descreve as principais funcionalidades do sistema e a interação dessas funcionalidades com os usuários do mesmo sistema. No diagrama, não são aprofundados detalhes técnicos do sistema, somente da relação entre usuário e sistema.

Segundo Pressman e Maxin (2016, pag. 173), um caso de uso descreve “cenário de uso específico em uma linguagem simples sob o ponto de vista de um ator definido”.

5.2.1.1 Diagrama de Casos de Uso

Diagramas de Casos de Uso são compostos basicamente por quatro partes:

- a) Cenário: sequência de eventos que acontecem quando um usuário interage com o sistema.
- b) Ator: usuário do sistema, podendo ser pessoa ou um tipo de sistema;
- c) Caso de uso: é uma tarefa ou uma funcionalidade realizada pelo ator;
- d) Comunicação: é a interação de um ator com um caso de uso.

O diagrama de casos de uso do sistema pode ser encontrado no Apêndice G.

5.2.1.2 Descrição dos Casos de Uso

A descrição de caso de uso pode ser vista como os cenários possíveis que podem ocorrer entre ator e caso de uso. Nessa descrição são levantados as pré-condições, pós-condições e os fluxos de eventos necessários para realizar dessa função.

A descrição dos casos de uso se encontra no Apêndice H.

5.2.2 Modelo Conceitual dos Dados

Modelo conceitual de dados é constituído por artefatos centrais de arquitetura de dados, sendo definidos como representações gráficas dos requisitos de informação relevantes a uma determinada área de assunto. Ele possui alto nível de abstração, não sofrendo

interferência de fatores tecnológicos e fatores de projeto em sua construção. Neste projeto, utiliza-se o MER para elaborar o modelo conceitual dos dados.

O modelo conceitual dos dados do sistema deste projeto pode ser encontrado no Apêndice I.

5.2.3 Modelo Inicial da Interface de Usuário

O modelo caracteriza uma perspectiva para a interface de usuário como tendo um componente físico, que o usuário percebe e manipula, e outro conceitual, que o usuário interpreta, processa e raciocina.

Nesta seção é apresentado modelo inicial de telas do sistema que o usuário manipulará, não sendo seu *design* definitivo.

O modelo inicial da interface de usuário pode ser encontrado no Apêndice J.

5.3 MÉTRICAS PARA ESTIMATIVAS DE ESFORÇO

Neste projeto, a partir da Fase 2, constam as estimativas baseadas em Pontos de Casos de Uso (PCU). A estimativa de esforço por Pontos de Função constam a partir da Fase 3.

As planilhas com resultado das estimativas de esforço encontram-se no Apêndice K deste projeto. Além disso, a síntese das métricas é apresentada no Quadro 4.

Métrica	Fase 2	Fase 3 (reestimativa)	Fase 4 (reestimativa)
PCU - Schneider e Winters	4578 Hh	5761 Hh	5604Hh
PCU – Karner	3270 Hh	4115 Hh	4003Hh
Pontos por Função	-	-	2889Hh

QUADRO 4: Métricas de estimativa do projeto

6 ARQUITETURA E PROJETO DO SISTEMA DE SOFTWARE

A arquitetura e projeto do sistema de software são atividades que definem, sob o ponto de vista de elementos do sistema, as interfaces, comportamentos e os componentes que devem constar no sistema.

Este capítulo apresenta a visão estrutural, visão comportamental, visão de dados e o projeto da interação humano-computador e de sistemas distribuído *Help a Pet*.

6.1 VISÃO ESTRUTURAL

A visão estrutural de um projeto tem como prioridade a descrição estática das estruturas do sistema, como classes, atributos, operações e as prováveis associações entre tais elementos.

São apresentados nesta seção os seguintes diagramas: diagrama de classes, diagrama de objetos e diagrama de pacotes.

6.1.1 Diagrama de Classes

O diagrama de classes da UML tem como objetivo mostrar as classes dos objetos manipulados pelo sistema, seus atributos, operações e relacionamentos existentes entre as classes (PRESSMAN; MAXIN, 2016).

O diagrama de classes do deste projeto encontra-se no Apêndice L deste documento.

6.1.2 Diagrama de Objetos

O diagrama de objetos é o modelo que mostra as instâncias das classes contidas no diagrama de classes, mostrando um conjunto de objetos e seus relacionamentos. Estes diagramas são importantes para construir os aspectos estáticos do sistema, sendo compostos por objetos e seus vínculos.

O diagrama de objetos deste projeto encontra-se no Apêndice M deste documento.

6.1.3 Diagrama de Pacotes

O diagrama de pacotes ou também conhecidos como diagramas de módulos têm como conceito agrupar logicamente os pedaços dos sistemas (nível superior) com dependência entre eles (interação de pacotes), ou seja, pacotes podem depender de outros pacotes. Esses elementos agrupados podem ser diagramas, classes, outros pacotes, entre outros.

O diagrama de pacotes do projeto *Help a Pet* demonstra os pacotes dos componentes servidores (*back-end*) *web*, *frameworks* e tecnologias externas como *Spring MVC* e *Hibernate*.

O diagrama de pacotes deste projeto encontra-se no Apêndice N deste documento.

6.2 VISÃO COMPORTAMENTAL

A visão comportamental é a representação do comportamento do sistema, com as associações entre os objetos e as interações realizadas entre eles por meio do envio de requisições e mensagens.

São apresentados nesta seção os seguintes diagramas: diagrama de sequência, diagrama de visão geral de interação e diagrama de atividades.

6.2.1 Projeto das Interações entre Objetos

6.2.1.1 Diagrama de Sequência

O diagrama de sequência é usado para modelar as interações entre os atores e os objetos em um sistema e as interações do próprio objeto. Um diagrama de sequência mostra a sequência de interações que ocorrem durante um caso de uso em particular (SOMMERVILLE, 2007).

No Apêndice O deste projeto encontra-se os diagramas de sequência do caso de uso “Manter Cadastro de Animais”.

6.2.1.2 Diagrama de Visão Geral de Interação

O diagrama de visão geral de interação mostra a exibição das interações gerais do sistema, fazendo a relação entre interações demonstradas por diagramas de sequência específicos. O diagrama de visão geral de interação para o caso de uso “Manter Cadastro de Animais” encontra-se no Apêndice P deste documento.

6.2.2 Diagrama de Atividades

O diagrama de atividades é um gráfico que mostra o fluxo de controle de uma atividade para outra, mostrando como será o funcionamento do software (em nível micro ou macro), como será a execução de alguma de suas partes, como será a atuação do sistema na realidade de negócio na qual ele está inserido. No Apêndice T desse projeto consta o diagrama de atividades do caso de uso “cadastrar usuário adotante”

6.3 VISÃO DE DADOS

Nesta seção encontra-se o modelo operacional (ou lógico) do banco de dados, com objetivo de mostrar a base de dados do sistema, assim como os relacionamentos existentes entre as suas tabelas, também consta ainda o dicionário de dados de todas as tabelas.

6.3.1 Modelo Lógico

Na elaboração do modelo lógico, deve-se levar em conta a modelagem conceitual de dados, implementando recursos como adequação de padrão, nomenclatura, definindo as chaves primárias e estrangeiras, normalização, integridade referencial.

O modelo lógico dos dados do sistema pode ser consultado no Apêndice Q.

6.3.2 Dicionário de Dados do Modelo Lógico

O dicionário de dados é uma lista organizada que contém definições e representações de todos os atributos persistidos do sistema, incluindo as especificações de cada campo, divididos em suas respectivas tabelas.

O dicionário de dados deste projeto encontra-se no Apêndice R deste documento.

6.3.3 Estrutura Física do Banco de Dados

Este é o nível mais baixo de abstração (mais perto do modelo físico), pois descreve o modo como os dados são armazenados e recuperados pelo SGDB. A estrutura física do banco de dados do sistema, construída com o SGDB PostgreSQL, encontra-se no Apêndice S deste documento.

Estrutura	Documento
Scripts DDL	Vide Apêndice U
Scripts DML	Vide Apêndice V

QUADRO 5: Estrutura física do banco de dados

6.4 VISÃO FÍSICA

A visão física é decorrente da modelagem dos aspectos físicos que contemplam decomposições do sistema em módulos, ou seja, obtém-se a visão do estágio final do sistema de software, que está sendo acondicionado em um ambiente computacional.

6.4.1 Diagrama de Componentes

Um diagrama de componentes permite demonstrar a captura da estrutura física da implementação, construída como parte da especificação da arquitetura.

Um diagrama de componentes do sistema encontra-se no Apêndice Z deste documento.

6.4.2 Frameworks adotados

Framework é um conjunto de códigos abstratos e/ou genéricos, geralmente classes, desenvolvidos em alguma linguagem de programação, que se relacionam entre si para disponibilizar funcionalidades específicas para programação de softwares. O *Help a Pet* utiliza os seguintes *frameworks*:

- a) *Spring Boot*: segundo Java Avançado (2018), é um projeto da *Spring* elaborado para facilitar o processo de configuração e publicação de nossas aplicações. Escolhem-se os módulos que se desejam utilizar (através dos *starters*) e os inclui no arquivo “pom.xml” do projeto. Esses, basicamente, são dependências que agrupam outras dependências;
- b) *SpringMVC*: segundo Afonso (2017), é um *framework* que ajuda no desenvolvimento de aplicações *Web*. Ele já tem todas as funcionalidades para atender as requisições HTTP, delegar responsabilidades de processamento de dados para outros componentes e preparar a resposta que precisa ser dada. É uma excelente implementação do padrão MVC;
- c) *Hibernate*: Segundo Devmedia(2009), é um *framework* de consulta e persistência objeto/relacional de alto desempenho. Uma das soluções de Mapeamento Objeto Relacional (ORM) mais flexíveis e poderosas no mercado, ele faz o mapeamento de classes Java para tabelas de banco de dados e de tipos de dados Java para tipos de dados SQL;
- d) *Swagger*: segundo Aquiles e Ferreira (2016), é uma das principais ferramentas utilizadas para modelagem, documentação e geração de código para APIs do estilo REST;
- e) *Spring Security*: segundo o Spring.io (2019), é uma estrutura de autenticação e controle de acesso poderosa e altamente personalizável. É o padrão de fato para proteger aplicativos baseados em *Spring*. O *Spring Security* é uma estrutura que se concentra no fornecimento de autenticação e autorização para aplicativos Java;
- f) *Spring Data*: Segundo o Spring.io (2019), a missão da *Spring Data* é fornecer um modelo de programação familiar e consistente baseado em *Spring* para acesso a dados, mantendo as características especiais do armazenamento de dados subjacente. Isso facilita o uso de tecnologias de acesso a dados, bancos de dados relacionais e não relacionais, estruturas de redução de mapa e serviços de dados baseados em nuvem.

6.4.3 Design Patterns Aplicados

Os *Design Patterns* são também conhecidos como padrões de projetos. Exemplos de *design patterns* utilizados durante a codificação do sistema encontram-se no Quadro 6.

Tipo	<i>Design Pattern</i>	Exemplos
Estrutural	MVC	Arquitetura do sistema como um todo
Comportamental	DTO	AdotanteNewDTO.java
Comportamental	Iterator	AdotanteRepository.java
Arquitetural	<i>Dependency Injection</i>	InstituicaoService.java

QUADRO 6: *Design Patterns* adotados

- a) MVC: é um padrão de arquitetura de software, utilizado para organizar e distribuir o código em camadas. No *Help a Pet*, o conceito MVC é aplicado na componente servidora;
- b) DTO: é um padrão de projeto de software usado para transferir dados de um local a outro na aplicação, sem lógica de negócios em objetos. No *Help a Pet* esse padrão é utilizado na inserção e atualização dos dados;
- c) O padrão *Repository* separa a lógica de acesso a dados e mapeia essa lógica para entidades na lógica de negócio. Ele trabalha com as entidades de domínio e realiza a lógica de acesso a dados. No *Help a Pet*, esse padrão é utilizado para fazer a persistência e acesso dos dados no banco de dados.
- d) Injeção de dependência: é uma técnica utilizada para diminuir o acoplamento entre classes. No *Help a Pet*, utilizam-se as notações *@Autowired* permitem que o *Spring* injete as dependências em classes específicas.

6.4.4 Convenções e Guias para Codificação

As convenções de código são uma boa prática de programação. Ajudam, por exemplo, na manutenção ou evolução de um sistema. A violação dessas “regras” prejudica em muito o entendimento de um código fonte e com convenções conhecidas é possível produzir um código legível, “limpo”, que demonstra qualidade e de maneira profissional.

As convenções na codificação são facilitam a legibilidade e a manutenção do código-fonte. Exemplos de casos que se adotam guias para codificação: padrões de nomenclatura de pacote, classes, métodos e variáveis. O *Help a Pet* utiliza essas convenções no seu código-fonte, de acordo com as melhores práticas indicadas pela *Java Code Conventions*, da *Oracle*.

Os pacotes, por exemplo, são nomeados com todas as letras em minúsculas. Já para nomear as classes, utiliza-se o padrão *Camel Case* onde a primeira letra de cada palavra inicia-se com maiúscula. Quando a classe possui um nome composto, a primeira letra de cada palavra começa com letra maiúscula. Os métodos começam com letra minúscula e, caso o nome do método seja composto por mais de uma palavra, a primeira letra da segunda palavra e das palavras posteriores começa com letra maiúscula.

6.4.5 Análise de Complexidade Algorítmica

O trecho de código visto na Figura 18 tem como objetivo de cadastrar um usuário instituição no banco de dados, enviar um e-mail de confirmação para a instituição que realizou o cadastro e enviar um e-mail para todos os administradores do sistema informando-os que um novo cadastro de instituição foi realizado.

Como o método possui dois laços de repetição encadeados onde trata-se de um algoritmo com a complexidade $O(n \times m)$, onde n é o número de usuários cadastrados no sistema e o m é o número de perfis que cada usuário possui.

```

@Transactional
public Instituicao insert(Instituicao obj) {
    obj = instituicaoRepository.save(obj);
    emailService.sendOrderConfirmationEmail(obj);
    List<Usuario> usuarioList = usuarioRepository.findAll();
    for (Usuario usuario : usuarioList) {
        for (Perfil perfil : usuario.getPerfil()) {
            if (perfil.getCod() == 1) {
                emailService.sendReleaseRegister(obj, usuario);
            }
        }
    }
    return obj;
}

```

FIGURA 18: Método para inclusão de um novo usuário da instituição

O trecho de código visto na Figura 19 tem como objetivo de gerar um *token* com data de expiração para o usuário quando o mesmo se autenticar no sistema.

Como esse método não possui nenhum laço de repetição e as suas instruções serão executadas apenas uma única vez a complexidade desse algoritmo é constante $O(1)$.

```

public String generateToken(String username) {
    return Jwts.builder().//gere token
        .setSubject(username)
        .setExpiration(new Date(System.currentTimeMillis() + expiration))
        .signWith(SignatureAlgorithm.HS512, secret.getBytes())
        .compact();
}

```

FIGURA 19: Método para gerar um *token* de segurança

O trecho de código visto na Figura 20 tem como objetivo gerar uma nova senha com 10 caracteres para o usuário.

Como o método possui apenas 1 laço de repetição com interações fixas. As instruções dos algoritmos são executadas um número fixo de vezes, no caso, trata-se de um algoritmo com complexidade constante $O(1)$.

```

private String newPassword() {
    char[] vet = new char[10];
    for (int i = 0; i < 10; i++) {
        vet[i] = randomChar();
    }
    return new String(vet);
}

```

FIGURA 20: Método para gerar uma nova senha para o usuário

6.5 PROJETO DO SISTEMA DISTRIBUÍDO

De acordo com Coulouris, Dollimore e Kindberg (2007, p.15), um sistema distribuído é aquele no qual os componentes localizados em computadores interligados em rede, se comunicam e coordenam suas ações apenas passando mensagens. Essa definição leva-se as seguintes características dos sistemas distribuídos: concorrência de componentes, falta de um relógio global e falhas de componentes independentes.

6.5.1 Procedimentos para Tratamentos dos Desafios

Nesta seção são abordados os principais requisitos que um sistema distribuído precisa apresentar para que haja interoperabilidade funcional. Estes requisitos são: heterogeneidade, escalabilidade, abertura, segurança, manuseio de falhas, concorrência e transparência. Cada item terá uma breve análise e a solução encontrada no projeto do *Help a Pet* para atender esses requisitos.

6.5.1.1 Heterogeneidade

Segundo Coulouris, Dollimore e Kindberg (2007, p.28), a Internet permite aos usuários acessarem serviços e executarem aplicativos por meio de um conjunto heterogêneo de computadores e redes. A heterogeneidade (isto é, variedade e diferença) se aplica aos seguintes aspectos:

- a) redes: para que haja comunicação é necessário a utilização de protocolos de Internet;

- b) hardwares de computador: computadores que possuem placa de rede *Ethernet* têm uma implementação diferente daqueles que possuem placas de outros padrões, isto acontece também com arquiteturas de microprocessadores (Intel, ARM, etc.);
- c) sistemas operacionais: embora os sistemas operacionais de todos computadores na Internet precisam incluir uma implementação de protocolos de Internet, os diferentes sistemas operacionais trazem consigo diferentes formas de programação e interface para esses protocolos;
- d) linguagens de programação: cada linguagem de programação possui suas singularidades, seja no tratamento de vetores, registros e variáveis. É necessário tratar essas diferenças para que não haja erros de informações e haja uma comunicação efetiva;
- e) implementação de diferentes desenvolvedores: é necessário que diferentes desenvolvedores utilizem de padrões comuns no desenvolvimento para que haja comunicação entre os diferentes sistemas.

O *Help A Pet* é uma solução distribuída. A aplicação servidora (*back-end*) é implementada utilizando a linguagem de programação Java e ela, assim como a aplicação *front-end*, são executadas por meio do *container web* Apache Tomcat.

Também é utilizada neste sistema a arquitetura *Representational State Transfer* (REST), para acesso aos serviços providos pela aplicação servidora, linguagem Java e estilo de programação SOA/REST. O sistema utiliza o protocolo *Hypertext Transfer Protocol Secure* (HTTPS) para comunicação de dados entre as aplicações.

6.5.1.2 Escalabilidade

De acordo com Coulouris, Dollimore e Kindberg (2007, p.31), um sistema é descrito como escalável, se ele permanece eficiente quando há um aumento significativo no número de recursos e no número de usuários. Além disso, para resolver o desafio da escalabilidade, o desempenho dos algoritmos e o acesso de dados compartilhado devem ser otimizados para que não haja gargalos no sistema. Para lidar com a escalabilidade, neste sistema, utilizam-se boas práticas de programação, com algoritmos otimizados. Em relação ao banco de

dados, existem rotinas de manutenção e criação de índices e também podem ser utilizadas características escaláveis do *Apache Tomcat*.

6.5.1.3 Abertura

De acordo com Tanenbaum e Steen (2007, p.4), um sistema distribuído aberto é um sistema que oferece serviços de acordo com regras padronizadas que descrevem a sintaxe e a semântica desses serviços. No caso de sistemas distribuídos, em geral os serviços são especificados por meio de interfaces, que costumam ser descritas em uma linguagem de definição de interface (*Interface Definition Language – IDL*).

A principal característica de um sistema aberto é a documentação das principais interfaces do software com as principais funções, como utilizá-las em outras aplicações e principalmente a divulgação da API para que outros desenvolvedores possam utilizar dos recursos oferecidos pelo sistema.

No Help a Pet, utiliza-se o *Swagger* que é um *framework* para descrição, consumo e visualização de serviços RESTful. O objetivo dele é permitir que a documentação possa evoluir no mesmo ritmo da implementação, já que pode ser gerada automaticamente com base em anotações do código. O *Swagger* tem um módulo de interface de usuário que permite aos programadores interagirem com as APIs em *sandbox* de forma muito intuitiva, sem exigir conhecimento da implementação ou mesmo dos parâmetros e opções (que são explícitas na documentação).

6.5.1.4 Segurança

Segundo Coulouris, Dollimore e Kindberg (2007, p.30), muitos recursos de informação que se tornam disponíveis e são mantidos em sistemas distribuídos têm um alto valor intrínseco para seus usuários. Portanto, sua segurança é de considerável importância. A segurança de recursos de informação têm três componentes: confidencialidade (proteção contra exposição para pessoas não autorizadas), integridade (proteção contra alteração ou dano) e disponibilidade (proteção contra interferência com os meios de acesso aos recursos).

Neste projeto, utiliza-se o *JSON Web Token* (JWT), que de acordo com Adriano (2017), é um sistema de transferência de dados que pode ser enviado via requisição POST ou em um cabeçalho HTTP (*header*) de maneira “segura”, essa informação é assinada digitalmente por um algoritmo HMAC, ou um par de chaves pública/privada, usando método de criptografia *Rivest – Shamir – Adleman* (RSA). Pode-se ver na Figura 21 um cenário onde será requisitado um *token* por meio do verbo HTTP POST, que irá devolver um *token* validado para que as próximas requisições que utilizem os verbos HTTP possam utilizá-lo.

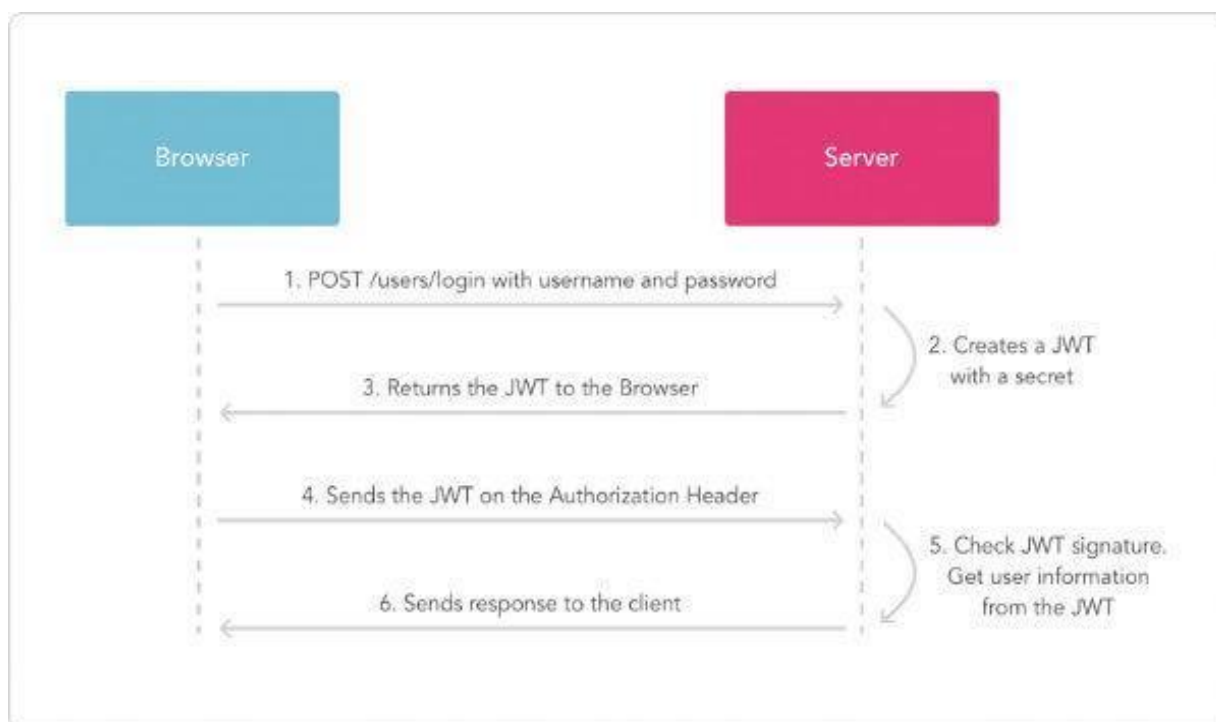


FIGURA 21: Diagrama de sequência usando JWT

FONTE: Adriano (2017)

O sistema Help a Pet também utiliza o *BCrypt*, que segundo Boterhoven (2016) é baseado no algoritmo criptográfico da cifra de bloco *Blowfish* e assume a forma de uma função de *hash* adaptável. Usando um fator-chave, o *BCrypt* pode ajustar o custo do *hash*. Com as alterações do fator-chave, a saída de *hash* pode ser influenciada. Dessa forma, o *BCrypt* permanece resistente, especialmente, a um tipo de quebra de senha chamado *rainbow table* que, segundo o Blogspot(2010) é uma tabela de consulta que utiliza um algoritmo de transação memória-tempo, permitindo recuperar o texto original de uma senha de uma

senha digerida (*hash*), gerada por uma função criptográfica de digestão (*hashing*). É utilizado normalmente para que um ataque contra senhas digeridas seja possível.

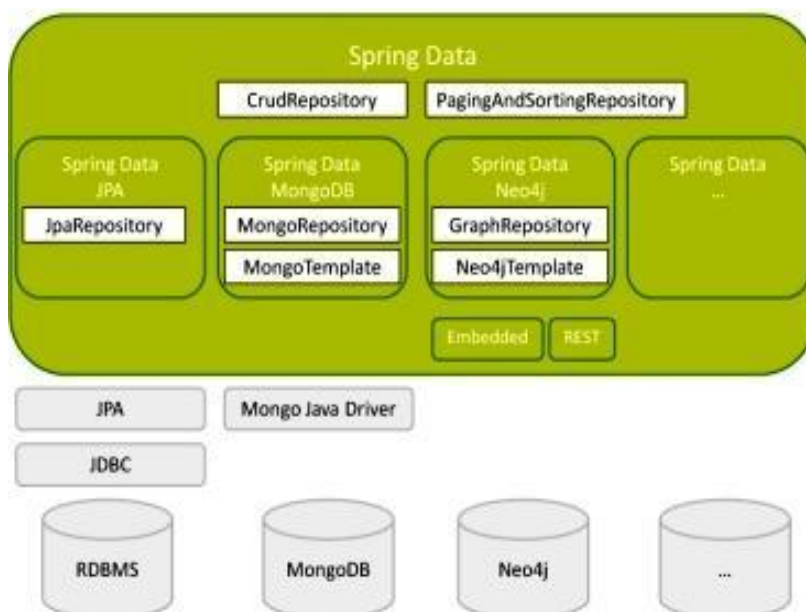
Utiliza-se o *Spring Security* que é uma estrutura de autenticação e controle de acesso poderosa e altamente personalizável. É o padrão de fato para proteger aplicativos baseados em *Spring*. O *Spring Security* é uma estrutura que se concentra no fornecimento de autenticação e autorização para aplicativos escritos em Java. Como todos os projetos do *Spring*, o poder real do *Spring Security* é encontrado na facilidade com que pode ser estendido para atender aos requisitos personalizados.

6.5.1.5 Manuseio de Falhas

Colouris, Dollimore e Kindeberg (2007) afirmam que falhas em sistemas distribuídos são parciais, pois todos eles possuem componentes que funcionam independentemente, portanto, mesmo com a falha de um componente, os outros podem continuar operando normalmente, dificultando assim o manuseio de falhas. É essencial que todo componente seja desenvolvido com o objetivo de que ele, mesmo ocorrendo falhas dos componentes que depende, funcione ou trate destas falhas apropriadamente.

O *Help a Pet* trata apenas as falhas em nível de aplicação e relacionadas a qualquer erro de acesso ao banco de dados. Para isso, utiliza técnicas apropriadas para o tratamento de exceções que possam ocorrer em tempo de execução e o controle de transações no SGDB.

Segundo Trelle (2012), objetivo do *Spring Data* é unificar e facilitar o acesso a diferentes tipos de repositórios de persistência, tanto sistemas de banco de dados relacionais quanto repositórios de dados NoSQL. Com todo tipo de armazenamento de persistência, seus repositórios (também conhecido como *Data Access Objects*) geralmente oferecem operações CRUD (*Create-Read-Update-Delete*) em objetos de domínio único, métodos localizadores, classificação e paginação. O *Spring Data* fornece interfaces genéricas para esses aspectos bem como implementações específicas de armazenamento de persistência. A Figura 22 mostra a arquitetura do *Spring Data*.

FIGURA 22: *Spring Data*

FONTE: Trelle (2012)

6.5.1.6 Concorrência

Segundo Colouris, Dollimore e Kindeberg (2007), em um ambiente concorrente, cada recurso deve ser projetado para oferecer consistência nos estados de seus dados. É essencial que todos os recursos estejam disponíveis, com o maior desempenho possível e para o maior número de usuários possíveis simultaneamente.

Como forma de controlar o máximo possível de conexões simultâneas no acesso ao sistema, utiliza-se o *Apache Tomcat* na componente servidora. Para acesso e controle das requisições ao banco de dados, utiliza-se o *Spring Data Java Database Connectivity* (JDBC) específico para o SGDB PostgreSQL. Trata-se de um módulo do *framework* da família do *Spring Data*. Ele facilita a implementação de repositórios baseados em JDBC. Este módulo lida com suporte aprimorado para camadas de acesso a dados baseadas em JDBC, o que facilita a criação de aplicativos com tecnologias *Spring* que usam tecnologias de acesso a dados.

Spring Data JDBC não oferece armazenamento em cache, carregamento lento, gravação atrasada ou muitos outros recursos do JPA. Isso faz do *Spring Data JDBC* um ORM simples.

6.5.1.7 Transparência

Conforme Colouris, Dollimore e Kindeberg (2007), o desafio da transparência tem o objetivo de tornar aspectos da distribuição invisíveis para o desenvolvedor, a fim de que ele se preocupe apenas com o projeto de seu sistema em particular. O objetivo de tornar ocultos certos aspectos da distribuição é para que este seja percebido como um sistema único em vez de uma coleção de componentes independentes.

Um sistema com o objetivo de atender às necessidades dos clientes pode realizar diversas operações relacionadas ao banco de dados, utilizar a API sem a necessidade de entendimento da distribuição, suas regras técnicas e de implementação, apenas fornecendo os dados necessários.

O *Help A Pet* utiliza métodos que podem ser executados concorrentemente e recursos compartilhados sem a interferência entre si, realizados via funções do *Apache Tomcat* e pelo SGBD PostgreSQL e o Spring Data que fornece interfaces genéricas para esses aspectos bem como implementações específicas de armazenamento de persistência.

6.5.2 Procedimentos para Tratamentos dos Desafios

O *Help a Pet* possui uma arquitetura baseada em cliente/servidor. Para cada requisição de um cliente é o servidor que irá fazer todo o processamento e retornar as respostas, por meio de protocolos de rede.

O servidor é responsável por validar as requisições do cliente, processar, persistir os dados e retornar o resultado. O cliente (um navegador) é o canal de comunicação com o usuário, sendo por intermédio dele que ele terá acesso ao sistema, além de prover a interface gráfica de usuário necessária para a interação.

Os processos servidores são constituídos por:

- a) P1 - processo SGBD PostgreSQL: é o processo responsável por fazer o gerenciamento de acessos ao banco de dados. O banco de dados é chamado de *HelpAPet*;

6.6 PROJETO DA INTERAÇÃO HUMANO-COMPUTADOR

6.6.1 Perfil do Usuário

Observando as principais características das plataformas onde os componentes do software *Help a Pet* estarão para acesso, foi possível analisar e identificar os perfis dos prováveis usuários.

a) Visitante do *website*

Faixa etária: acima de 18 anos.

Sexo: indiferente.

Características: são usuários que possuem alguma familiaridade na utilização de sistemas computacionais existentes, por já saberem utilizar navegador para compras em *e-commerce* e redes sociais.

b) Usuário que visa a adoção de animais:

Faixa etária: acima de 18 anos.

Sexo: indiferente.

Características: são usuários com pouca ou nenhuma familiaridade na utilização de sistemas computacionais. Para atender a esta categoria, o sistema *Help a Pet* deve fornecer uma interface prática e intuitiva, acentuando os passos necessários para que se tenha uma experiência positiva e bem-sucedida;

c) Usuário da Instituição:

Faixa etária: acima de 18 anos.

Sexo: indiferente.

Características: são usuários que têm familiaridade na utilização de sistemas computacionais existentes. O sistema *Help a Pet* funcionará como uma ferramenta facilitadora, visando o melhor desempenho, pois tem funções bem definidas e projetadas para este perfil desse colaborador.

No Apêndice S constam as entrevistas com possíveis usuários do sistema.




6.6.2 Aspecto Visual da Interface de Usuário

Para elaboração do aspecto visual, são utilizados componentes que trazem conforto para interação do usuário com o sistema. Esses componentes são representados por meio de botões, ícones, imagens e qualquer elemento visual que facilite a navegação.

Nesta seção, são apresentadas as principais características relacionadas ao aspecto visual.

6.6.3 Ícones

Para facilitar a navegação, o *Help a Pet* utiliza de ícones presentes em *websites* e aplicações conhecidas pela maior parte dos usuários:

- a)  ícone de página inicial. Ao selecionar este ícone, o usuário autorizado consegue acessar a página inicial.
- b)  ícone para entrar na página como um usuário. Ao selecionar este ícone, o usuário consegue acessar a página destinada a ele, para poder efetuar funções que são exclusivas para quem é cadastrado.
- c)  ícone de pesquisa. Ao selecionar esse ícone, eu pesquiso os animais, com suas respectivas características que foram escolhidas pelo usuário.

6.6.4 Fontes Tipográficas

As fontes têm por objetivo facilitar a compreensão e tornar a experiência de leitura mais agradável do usuário. Diferentes tamanhos de fonte para os diferentes tamanhos de tela devem ser utilizados para tornar a leitura e navegação agradável ao usuário. A fonte escolhida foi Arial 12.

6.6.5 Cores

As cores escolhidas para o *Help a Pet* são branca e preta. Branco simboliza pureza, integridade e clareza e por isso é uma cor muito usada na maioria dos *websites* de *e-Commerce*. Um bom uso dele é incluir espaço em branco (vazio) em torno de seu conteúdo,

de modo que os clientes não se sintam “encaixotados”. A cor preta é associada com força, poder e dominação. Quando usado no comércio eletrônico, ele envia uma mensagem de confiança aos potenciais clientes, misturando com o branco que é a cor que mais predomina na aplicação, para não deixar que a interface fique sombria, quebrando a cor escura do preto.

6.6.6 Heurística de Usabilidade

O projeto *Help a Pet* segue as heurísticas de usabilidade de propostas por Jacob Nielsen. A Figura 24 mostra a consistência e padrões visuais (texto, cor e desenho do elemento).

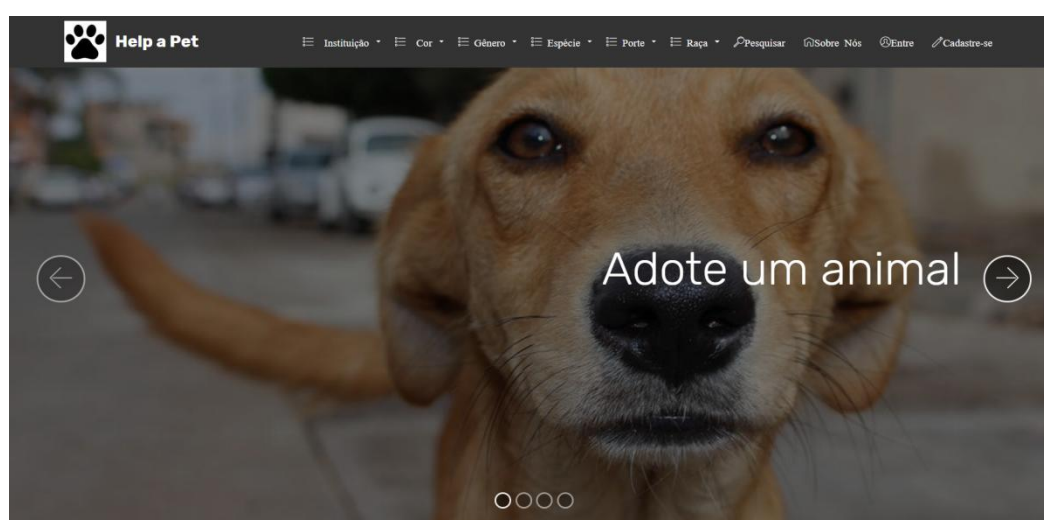


FIGURA 24: Consistência e padrões visuais

A Figura 25 mostra a correspondência entre o sistema e o mundo real, pois existem símbolos, por exemplo, que sempre são utilizados para representar determinadas ações.

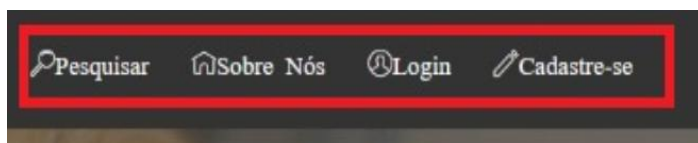
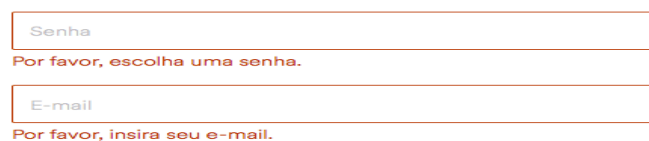


FIGURA 25: Correspondência entre o sistema e o mundo real

A Figura 26 mostra a ajuda dada aos usuários para reconhecerem, diagnosticarem ou recuperarem se de erros. As mensagens de erros devem ser pequenas e em linguagem simples, indicando precisamente o problema e sugerindo construtivamente uma forma para o usuário chegar a uma solução.



Senha

Por favor, escolha uma senha.

E-mail

Por favor, insira seu e-mail.

FIGURA 26: Mensagens de ajuda nas entradas de dados

7 PLANO DE TESTES

Teste de software é o processo de execução de um artefato visando determinar se ele atingiu suas especificações e funcionou corretamente no ambiente para o qual foi projetado. O seu objetivo é revelar falhas em um produto, para que as causas dessas falhas sejam identificadas e possam ser corrigidas pela equipe de desenvolvimento antes da entrega para uso do usuário final.

Este capítulo é apresenta um plano de testes para o sistema *Help a Pet*.

7.1 FINALIDADE

O plano de testes aqui apresentado tem a finalidade verificar o correto funcionamento de algumas funcionalidades oferecidas pelo *Help a Pet*.

7.2 ESCOPO

O plano de testes deste documento limita-se a 7 requisitos funcionais, demonstrando todo fluxo de execução de cada um desses requisitos, desde a criação, edição e exclusão dos dados.

7.2.1 Referências a Documentos Relevantes

Tipo do Material	Referência
Requisitos Funcionais e não funcionais.	Seção 5.1 deste documento.

QUADRO 7: Referências documentadas

7.2.2 Ambiente para a Realização de Testes

Equipamento	Marca/Modelo/Configuração	Finalidade
Computador	Processador i5 7200u, 8GB de memória RAM e 500GB de HD.	Testar o funcionamento do sistema e de seus componentes.

QUADRO 8: Equipamento para realização de testes

Software/Versão	Fabricante	Finalidade
Google Chrome	Google.	Navegador.
Windows 10	Microsoft.	Plataforma de Sistema Operacional.
PostgreSQL 11.2	<i>PostgreSQL Global Development Group</i>	Sistema gerenciador de banco de dados.
Java SE 8u112	<i>Oracle Corporation</i>	O Java SE é um ambiente de desenvolvimento para aplicações na linguagem de programação Java.

QUADRO 9: Software para realização de testes

7.3 ESPECIFICAÇÃO DOS CASOS DE TESTES

Nesta seção são apresentados os requisitos que serão testados, a rastreabilidade entre os requisitos e os casos de testes, e a descrição dos casos de testes que constam neste documento.

7.3.1 Item a Testar

Identificação do Item	Descrição
1	[RF 01] - Cadastrar adotante
2	[RF 02] - Cadastrar instituição
3	[RF 03] - Autenticar usuário
4	[RF 05] - Alterar senha
5	[RF 11] - Visualizar cadastro de instituição
6	[RF 12] - Liberar cadastro de instituição
7	[RF 07] - Manter cadastro de animais

QUADRO 10: Itens a testar

7.3.2 Rastreabilidade entre Requisitos e Casos de Teste

Identificação do Requisito	Caso(s) de teste(s) aplicável(is)
[RF 01] - Cadastrar adotante	CT01 - Realizar cadastro de adotante
[RF 02] - Cadastrar instituição	CT02 - Realizar cadastro de instituição
[RF 03] – Autenticar usuário	CT03 - Realizar acesso como adotante CT04 - Realizar acesso como instituição
[RF 05] – Alterar senha	CT05 - Realizar alteração de senha de usuário adotante

	CT06 - Realizar alteração de senha de usuário da instituição
[RF 11] – Visualizar cadastro de instituição	CT07 - Visualizar lista de intuições cadastradas aguardando liberação
[RF 12] – Liberar cadastro de instituição	CT08 - Liberar cadastro de instituição
[RF 07] – Manter cadastro de animais	CT09 - Realizar cadastro dos dados de um animal CT10 – Editar dados de um animal CT11 - Excluir dados de um animal

QUADRO 11: Rastreabilidade entre requisitos e casos de teste

7.3.3 Descrição dos Casos de Testes

As descrições dos casos de testes previstos no Quadro 10 encontram-se no Apêndice X.

7.4 RESULTADOS

Os casos de teste apresentam resultados compatíveis com as entradas, portanto, há coerência entre os requisitos funcionais e as funções apresentadas na aplicação *Help a Pet*. Além disso, a aprovação dos testes feitos comprova e resulta no bom funcionamento dessas funcionalidades oferecidas pelo sistema.

8 PLANO DA IMPLANTAÇÃO

O plano de implantação tem por objetivo garantir que o sistema de software fique disponível para uso de forma satisfatória seus usuários, proporcionando acesso a todas as funcionalidades de forma consistente e íntegra.

Neste capítulo são apresentadas a metodologia de implantação, treinamento previsto para que os usuários utilizem corretamente o sistema, cronograma de implantação, toda documentação de apoio à implantação do sistema e uma visão da implantação em um cenário específico.

8.1 METODOLOGIA

Esta seção apresenta a metodologia usada para implantação do sistema e os processos necessários para que o sistema seja implantado corretamente.

8.1.1 Descrição da Metodologia

Para o funcionamento correto do sistema, é necessário seguir uma série de passos ordenados. São eles:

1. Designar um responsável ou equipe responsável pela implantação;
2. Levantamento de requisitos de *hardware* e *software*;
3. Definir um servidor com capacidade necessária para a aplicação;
4. Configurar os componentes no servidor *Apache Tomcat*;
5. Criar uma instância no banco de dados do servidor *PostgreSQL*;
6. Fazer o *deploy* da aplicação no servidor.

8.1.2 Matriz de Responsabilidade

Atividades	Responsável
Planejamento	Taíses da Silva Mendes
Definição da equipe de implantação	Taíses da Silva Mendes
Levantamento de recursos necessários de <i>hardware</i>	Daniel Fernando Pereira
Levantamento de recursos necessários de <i>software</i>	Daniel Fernando Pereira
Definição de estratégias para conversão e migração de dados	Daniel Fernando Pereira
Programação dos treinamentos	André Silva Lopes
Preparação dos testes de aceitação	Taíses da Silva Mendes
Execução	Equipe
Configuração da infraestrutura de Tecnologia da Informação	André Silva Lopes
Instalação	Daniel Fernando Pereira
Conversão e migração de base de dados	Daniel Fernando Pereira
Treinamento	André Silva Lopes
Realização de testes de aceitação	Taíses da Silva Mendes
Avaliação	André Silva Lopes
Acompanhamento pós-implantação	Equipe
Reunião final da implantação	Equipe

QUADRO 12: Papéis e responsabilidades na implantação

8.2 TREINAMENTO PREVISTO

O treinamento tem como objetivo capacitar o usuário a desempenhar todas as funções que o sistema apresentará e tirar o máximo proveito delas, trazendo todas as vantagens e funcionalidades a sua disposição.

Os treinamentos previstos e apresentados no Quadro 13 capacitam o usuário a desempenhar todas as funções na aplicação, de modo correto, para que possa sanar todas as necessidades à qual este sistema foi criado.

Treinamento	Conteúdo	Grupo de Usuários
Demonstração das funcionalidades do sistema para os usuários do tipo instituição.	Ensinar a realização do cadastro de instituição no sistema, os cadastro dos animais abrigados e a emitir relatórios.	Usuários do tipo instituição.
Demonstração das funcionalidades do sistema para os usuários do tipo adotante.	Ensinar a realização do cadastro de adotante no sistema, realizar buscas de animais e adotar um animal. Esse tipo de treinamento deve ser por meio de manuais/vídeos.	Usuários do tipo adotante

QUADRO 13: Treinamento previsto

8.3 CRONOGRAMA DE IMPLANTAÇÃO

Nesta seção são definidos os tempos necessários para que todo o processo de implantação seja realizado. O Quadro 14 apresenta a tarefa e o tempo para implantação em horas.

Tarefa	Duração	Período
Instalação do Servidor	35 horas	-
Configuração do Banco de dados	5 horas	-
Outras configurações	15 horas	-
Treinamento dos usuários	5 horas	-
Tempo total estimado	60 horas	-

QUADRO 14: Cronograma de atividades da implantação

8.4 DOCUMENTOS DE APOIO À IMPLANTAÇÃO

O Quadro 15 apresenta os documentos oferecidos para apoiar o processo de implantação e para uso do sistema.

Documento	Referência
Manual de instalação e configuração do Apache Tomcat	Anexo B.
Manual de instalação do Postgresql	Anexo B.
Manual do Usuário	Apêndice W

QUADRO 15: Documentos de apoio à implantação

8.5 VISÃO DA IMPLANTAÇÃO

No Apêndice Y encontra-se o diagrama de implantação para melhor visualização das necessidades da implantação em um cenário específico.

9 CONCLUSÃO

Todas as instituições entrevistadas durante a realização deste projeto afirmaram utilizar as redes sociais para se comunicarem com a sociedade. Também confirmaram a utilidade de um sistema de software capaz de oferecer suporte operacional às suas atividades.

Nesse sentido, com o auxílio das pesquisas, entrevistas e reuniões sobre o projeto *Help a Pet*, houve avanços quanto à prioridade dos requisitos funcionais e não funcionais do sistema de software a ser desenvolvido.

Baseando-se nas necessidades elencadas pelas instituições entrevistadas, foram definidos os objetivos deste projeto, cujo alcance é demonstrado a seguir.

A página principal do sistema *Help a Pet* mostra algumas fotos de animais que estão disponíveis para adoção e mensagens que incentivam a adoção, a conscientização da adoção, o trabalho voluntário e as doações (que são as principais necessidades das instituições entrevistadas). A Figura 27 mostra exemplos dessas mensagens.

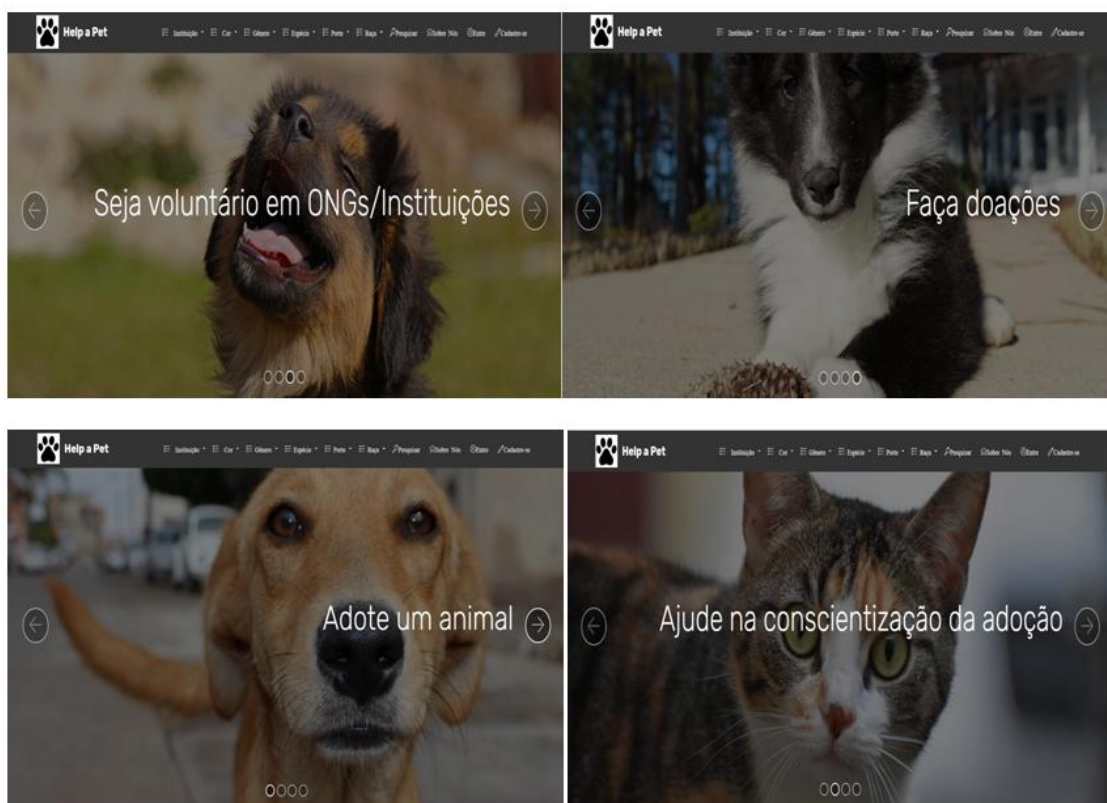


FIGURA 27: Mensagens que incentivam a adoção, o trabalho voluntário e as doações

Na Figura 28 é possível ver os animais em destaque que estão disponíveis para a adoção, com um breve texto sobre cada um deles.

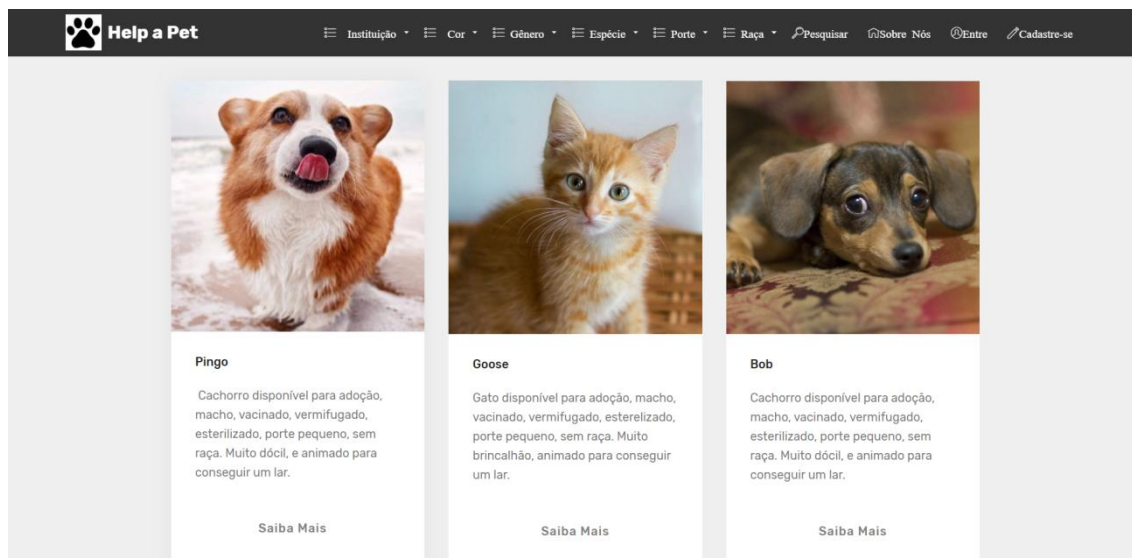


FIGURA 28: Animais em destaque para adoção

Um dos temas discutidos na revisão bibliográfica são os maus-tratos contra os animais e as penalidades previstas em lei contra esse tipo de crime. Assim, a página inicial do sistema faz uma abordagem à Lei Federal no. 9.605, que dispõe sobre as sanções penais e administrativas derivadas de condutas e atividades lesivas ao meio ambiente, e a proposta de alteração dessa lei para ampliar as sanções aplicáveis. As Figuras 29 e 30 mostram os textos abordados.

Denuncie os maus-tratos!

A denúncia de maus-tratos é legitimada pelo Art. 32, da Lei Federal nº. 9.605, de 12.02.1998 (Lei de Crimes Ambientais). Em 2018, o Senado Federal aprovou um projeto de lei de autoria do senador Randolfe Rodrigues, que amplia as sanções para crimes de maus-tratos a animais, alterando a pena de 3 meses a 1 ano de detenção para 1 a 4 anos de detenção, com a possibilidade de multa mantida.

Saiba Mais



FIGURA 29: Página sobre a lei de crimes ambientais

Denuncie os maus-tratos aos animais

O abandono e os maus-tratos em animais é crime e, por isso, deve ser denunciado e punido

Maus-tratos

Entende-se por maus-tratos o ato de submeter alguém ao tratamento cruel, trabalho forçado e/ou privação de alimentos ou cuidados, prejudicando-lhe a saúde física ou mental. Esse crime é praticado pelos mais variados tipos de pessoas e os motivos envolvem aspectos culturais, sociais e psicológicos, sendo muitas vezes praticado sem a consciência de que tal ato é prejudicial. A denúncia de maus-tratos é legitimada pelo Art. 32, da Lei Federal nº. 9.605, de 12.02.1998 (Lei de Crimes Ambientais). Em 2018, o Senado Federal aprovou um projeto de lei de autoria do senador Randolfe Rodrigues, que amplia as sanções para crimes de maus-tratos a animais, alterando a pena de 3 meses a 1 ano de detenção para 1 a 4 anos de detenção, com a possibilidade de multa mantida.

O que pode ser considerado como maus-tratos

- Manter os pets em lugares anti-higiênicos ou em locais que impeçam sua respiração, movimento ou descanso;
- Deixar o cão ou gato exposto ao sol por longos períodos de tempo, ou, ao contrário, sem qualquer tipo de iluminação;
- Obrigar o pet a trabalhos excessivos, inclusive em competições que possam causar pânico, estresse ou esforço acentuado;
- Golpear, mutilar ou ferir voluntariamente qualquer órgão do pet (com exceção do procedimento de castração);
- Não providenciar assistência veterinária em casos de acidentes ou de doença;
- Não garantir alimento e água para o pet.
- Abandono de cães e gatos.
- Capturar animais silvestres;
- Promover violência como rinhas, farra-do-boi, dentre outros.

FIGURA 30: Página sobre o que pode ser considerado como maus-tratos

A Figura 31 mostra a página para cadastro dos dados de uma pessoa interessada em adotar um animal do abrigo.

Criar Conta

☒ Adotante
 ☐ Instituição

Primeiro Nome

Sobrenome

CPF

RG

Data de Nascimento

Email

Senha

Confirmar Senha

Logradouro

Numero

CEP

Bairro

UF

Município

FIGURA 31: Página para cadastro da pessoa interessada na adoção

A Figura 32 mostra os dados de um animal que está no abrigo à espera por adoção.



Cor: Mesclado
 Espécie: Viralata
 Gênero: Macho
 Porte: Pequeno
 Vacinado: Sim
 Vermifugado: Sim
 Castrado: Sim

Descrição: Muito brincalhão, animado para conseguir um lar.



ADOTAR

FIGURA 32: Página que mostra os dados do animal que está no abrigo

A Figura 33 mostra a página que contém o termo de adoção preenchido.



TERMO DE ADOÇÃO E RESPONSABILIDADE
 Abandonar ou maltratar animais é crime! Pena: 3 meses a 1 ano de detenção e multa (Lei 9605/98)

Dados pessoais do adotante:
 Nome: Rafael Silva
 RG: MG-87953934
 Cidade: Pouso Alegre
 Nº: 354
 Email: rafael@gmail.com
 CPF: 63543368141
 Endereço: Rua Santa Efigênia
 Estado: MG
 CEP: 98765-000
 Bairro: Santa Barbara
 Telefone: 36533456

Dados do Animal:
 Nome: Bob
 Espécie: Cachorro
 Cor: Preto
 Raça: vira-lata
 Porte: Grande
 Idade: 11.0
 Gênero: Macho
 Vacinado: Sim
 Esterelizado: Não
 Vermifugado: Sim

Mediante a assinatura deste termo me comprometo a:

- * Ao adotar o animal acima descrito declaro-me apto para assumir a guarda e a responsabilidade sobre este animal, eximindo o doador de toda e qualquer responsabilidade por quaisquer atos praticados pelo animal a partir desta data.
- * Declaro ainda estar ciente de todos os cuidados que este animal exige no que se refere à sua guarda e manutenção, além de conhecer todos os riscos inerentes à espécie e raça no convívio com humanos, estando apto a guarda-lo e vigia-lo, comprometendo-me a proporcionar boas condições de alojamento e alimentação, assim como, espaço físico e/ou atividades regulares que possibilite o animal se exercitar.
- * Responsabilizo-me por preservar a saúde e integridade do animal e a submetê-lo aos cuidados médico-veterinários sempre que necessário para este fim.
- * Comprometo-me a não transmitir a posse deste animal a outrem sem o conhecimento do doador.
- * Comprometo-me a permitir o acesso do doador ao local onde se encontra o animal para visitas e averiguação de suas condições, podendo estas serem pré-agendadas ou não.
- * Tenho conhecimento de que caso seja constatada situação inadequada para o bem estar do animal e descumprimento das normas, perderei a sua guarda, ficando ainda sujeito às penas previstas na legislação protetora dos animais.

Declaro-me assim, ciente das normas acima, as quais aceito, assinando o presente Termo de Responsabilidade, assumindo plenamente os deveres que dele constam, bem como outros relacionados à posse responsável e que não estejam incluídos neste Termo.

Assinatura do responsável pela adoção _____ Assinatura do adotante _____
 _____ de _____ de 20 _____

FIGURA 33: Termo de adoção e responsabilidade

Uma melhoria no sistema será a possibilidade de adicionar contatos e propagandas de clínicas veterinárias ou de comércio de ração, visando arrecadar recursos para as instituições que precisam de ajuda financeira e também suprir os custos de hospedagem do sistema no provedor de Internet. Uma abordagem sobre colocar uma “vaquinha online” que algumas instituições utilizam também poderá ser considerada no futuro, bem como o cadastro de denúncias.

O trabalho das ONGs, a importância em se adotar um animal e a necessidade de se combater os maus-tratos sofridos por eles foram elementos compreendidos por todos os membros da equipe ao longo deste projeto. Também se verificou que a implantação de um sistema como o *Help a Pet* pode propiciar muitos benefícios à sociedade.

REFERÊNCIAS

- ADOTAFÁCIL. **Adota fácil**. Google Play, 2019. Disponível em: <<https://play.google.com/store/apps/details?id=com.adotafacil.daniel.adotafacil>>. Acesso em: 19 Mar. 2019.
- ADOTA PET GO. **Adote um animal próximo a você**. Google Play, 2019. Disponível em: <<https://play.google.com/store/apps/details?id=com.labup.adotapetv2>>. Acesso em: 19 Mar. 2019.
- ADRIANO, T. **Json Web Token – Conhecendo o JWT na teoria e na prática**. Imasters, 2017. Disponível em <<https://imasters.com.br/desenvolvimento/json-web-token-conhecendo-o-jwt-na-teoria-e-na-pratica>>. Acesso em: 01 Set. 2019.
- AFONSO, Alexandre. **O que é Spring MVC?**, Algaworks, 2017. Disponível em: <<https://blog.algaworks.com/spring-mvc/>>. Acesso em: 21 de Ago. 2019.
- AGÊNCIA DO SENADO. **Senado Aprova Aumento de Pena para o Crime de Maus-tratos a Animais**. Brasília: Senado Notícias, 2018. Disponível em: <<https://www12.senado.leg.br/noticias/materias/2018/12/11/senado-aprova-aumento-de-pena-para-o-crime-de-maus-tratos-a-animais>>. Acesso em: 09 Fev. 2019.
- AMIGO NÃO SE COMPRA. **Um blog sobre cuidados com seu pet**. Rio de Janeiro: 2019. Disponível em: <<https://www.amigonaosecompra.com.br/>>. Acesso em: 19 Mar. 2019.
- AMOR VIRA-LATA. **Home**. Caxias do Sul: 2019. Disponível em: <<http://www.amorviralata.com.br/>>. Acesso em: 19 Mar. 2019.
- APROBATO FILHO, Nélon. **O Couro e o Aço: sob a mira do moderno: a “aventura” dos animais pelos “jardins” da Paulicéia, final do século XIX/início do século XX**. Tese de doutorado em História Social, Universidade do Estado de São Paulo, São Paulo, 2006. Disponível em: <<http://www.teses.usp.br/teses/disponiveis/8/8138/tde-16072007-113730/pt-br.php>>. Acesso em: 11 Mar. 2019.
- AQUILES, A.; FERREIRA, R. **Modelando APIs REST com Swagger**. Caelum, 2016. Disponível em <<https://blog.caelum.com.br/modelando-apis-rest-com-swagger/>>. Acesso em: 21 Ago. 2019.
- BLOGSPOT. **O que é Rainbow Table?**. 2010. Disponível em <<https://o-que-quer-dizer.blogspot.com/2010/12/rainbow-table.html>>. Acesso em: Nov. 2019.
- BOTERHOVEN, D. **Por que você deve usar o BCrypt para hash senhas**. Medium Corporation, 2016. Disponível em <<https://medium.com/@danboterhoven/why-you-should-use-bcrypt-to-hash-passwords-af330100b861>>. Acesso em: 01 Set. 2019.

BRASIL. Lei Nº 9.605, de 12 de fevereiro de 1998. **Diário Oficial [da] União**, Brasília, DF, 13 Fevereiro 1998. Disponível em:
<http://www.planalto.gov.br/ccivil_03/Leis/L9605.htm>. Acesso em: 09 Fev. 2019.

CALHAU, L. B. Meio Ambiente e Tutela Penal nos Maus-Tratos contra Animais. **Fórum de Direito Urbano e Ambiental**, Belo Horizonte, 4 mar./abr. 2005. Disponível em:
<<http://sisnet.aduaneiras.com.br/lex/doutrinas/arquivos/Meio%20Ambiente.pdf>>. Acesso em: 11 Mar. 2019.

CARVALHO, Fábio Câmara Araújo de. **Gestão de projetos**. São Paulo: Pearson Education do Brasil, 2012.

COULOURIS, G.; DOLLIMORE, J.; KINDBERG, T. **Sistemas Distribuídos – Conceitos e Projetos**. 4. ed. Porto Alegre: Bookman, 2007. Disponível em:
<<http://www.inf.ufsc.br/~bosco.sobral/ensino/ine5645/coulouris.pdf>>. Acesso em: 24 Ago. 2019.

DELABARY, B. F. **Aspectos que Influenciam os Maus-Tratos Contra Animais no Meio Urbano**. Revista Eletrônica em Gestão, Educação e Tecnologia Ambiental, v. 5, n. 5, p. 835-840, 2012. Disponível em:
<<https://periodicos.ufsm.br/reget/article/viewFile/4245/2813>>. Acesso em: 09 Fev. 2019.

DELEGACIA ELETRÔNICA DE PROTEÇÃO ANIMAL. **Depa**. São Paulo: 2019. Disponível em: <<http://www.ssp.sp.gov.br/depa>>. Acesso em 11 Mar. 2019.

DEVMEDIA. **Desenvolvendo com Hibernate**, 2009. Disponível em:
<<https://www.devmedia.com.br/desenvolvendo-com-hibernate/14756>>. Acesso em: 21 Ago. 2019.

FERRARI, Rafael. **Entrevista sobre Zoonoses**. Santa Rita do Sapucaí: [S.n.], 2019.

GRISCI, C. L. I.; LOCATELLI, P. A. P. C.; D'ARISBO, A. A vida na lógica do Trabalho Imaterial: Análise do Acolhimento Temporário de Animais. **Gestão e Planejamento**, v. 18, jan./dez. 2017.

JAVA AVANÇADO. **O que é Spring Boot?**. 2018. Disponível em:
<<https://www.javaavancado.com/o-que-e-spring-boot/>>. Acesso em: 21 Ago. 2019.

JUCÁ, Ricardo. **Foto ilustrativa**. Facebook, 2018. Disponível em:
<<https://www.facebook.com/drricardojuca/photos/a.350484288337353/2107687389283692/?type=3&theater>>. Acesso em: 28 Mar. 2019.

MOUTINHO, F. F. B.; NASCIMENTO, E. R.; PAIXÃO, R. L. **Percepção da Sociedade sobre a Qualidade de Vida e o Controle Populacional de Cães Não Domiciliados**. Cienc. anim. bras., Goiânia, v.16, n. 4, p. 574-588, 2015. Acesso em: 09 Mar. 2019.

MINISTÉRIO DA SAÚDE. **Vigilância de Zoonoses**. Brasília: [S.n], 2018. Disponível em: <<http://portalms.saude.gov.br/noticias/zoonoses/44152-vigilancia-de-zoonoses>>. Acesso em: 18 Mar. 2019.

NAÇÕES UNIDAS BRASIL. **Conheça os Novos 17 Objetivos de Desenvolvimento Sustentável da ONU**. 2017. Disponível em <<https://nacoesunidas.org/conheca-os-novos-17-objetivos-de-desenvolvimento-sustentavel-da-onu/>>. Acesso em: 21 Ago.2019.

NIELSEN, J. **Ten Usability Heuristics**. USA: [S.n], 1994. Disponível em: <<https://www.nngroup.com/articles/ten-usability-heuristics/>>. Acesso em: 03 set. 2019.

ONG RESGACTI. **Foto ilustrativa**. 2019. Disponível em: <<https://www.facebook.com/RESGACTI/>>. Acesso em: 03 Mar. 2019.

OPPENHEIMER, Raphaella. **Entrevista sobre a Yellow Brick**. Santa Rita do Sapucaí: [S.n.], 2019.

PORTINARI, Natália. Senado Aprova Aumento da Pena de Maus-Tratos aos Animais. **O Globo**, 2018. Disponível em: <<https://oglobo.globo.com/sociedade/senado-aprova-aumento-da-pena-de-maus-tratos-aos-animais-23298327>>. Acesso em: 18 Mar. 2019.

PRESSMAN, Roger S.; MAXIN, Bruce R. **Engenharia de Software: uma abordagem profissional**. 8. ed. Porto Alegre: McGrawHill, 2016.

PROJECT MANAGEMENT INSTITUTE (PMI). **Guia PMBOK: um Guia do Conhecimento em Gerenciamento de Projetos**. 5. ed. São Paulo: Saraiva, 2013.

RESGACTI. **Sobre a Resgacti**. 2019. Disponível em: <<http://www.resgacti.org.br/>>. Acesso em: 03 Mar. 2019.

SANTANA L. R.; MARQUES, M. R. **Maus-Tratos e Crueldade Contra Animais nos Centros de Controle de Zoonoses: Aspectos Jurídicos e Legitimidade Ativa do Ministério Público para Propor Ação Civil Pública**. Salvador, 2001. Disponível em: <http://www.mp.go.gov.br/portalweb/hp/9/docs/maus_tratos_ccz_de_salvador.pdf>. Acesso em: 09 Fev. 2019.

SCHEFFER, G. K. **Maus-tratos aos Animais: Uma Perspectiva Criminológica**. Canal Ciências Criminais, 2018. Disponível em: <<https://canalcienciascriminais.com.br/maus-tratos-animais/>>. Acesso em: 12 Mar. 2019.

SILVA NETA, B. F.; LEITE, J. M. B.; SOUZA, F. S.; LISBÔA, R. S. **Ação de uma Organização Não Governamental de Proteção Animal na Cidade de Manaus**. In: III CONGRESSO BRASILEIRO DE BIOÉTICA E BEM-ESTAR ANIMAL, Anais... Curitiba, 2014. Disponível em: <<http://portal.cfmv.gov.br/uploads/anaisIII.pdf>>. Acesso em: 11 Mar. 2019.

SILVA, W. **Análise de Complexidade de Algoritmos**. LikeSys, 2018. Disponível em <<http://www.likesys.com.br/algoritmos/analise-de-complexidade-de-algoritmos/>>. Acesso em: 27 Out. 2019.

SOMMERVILLE, Ian. **Engenharia de Software**. 8. ed. São Paulo: Pearson Addison - Wesley, 2007.

SPRING.IO. **Spring Data**. 2019. Disponível em <<https://spring.io/projects/spring-data>>. Acesso em: 01 Set 2019.

SPRING.IO. **Spring Security**. 2019. Disponível em <<https://translate.google.com/translate?hl=pt-BR&sl=en&u=https://spring.io/projects/spring-security&prev=search>>. Acesso em: 01 Set 2019.

TOLEDO, L.F.; GIRARDI, G. **Polícia registra 21 casos de maus-tratos por dia**. Exame, 2016. Disponível em: <<https://exame.abril.com.br/brasil/policia-anota-21-casos-de-maus-tratos-a-animais-por-dia/>>. Acesso em: 11 Mar. 2019.

TOMAZELA J. M. São Paulo registra 25 mil casos de maus-tratos a animais por dia. O **Estado**, 2018. Disponível em: <<https://sao-paulo.estadao.com.br/noticias/geral,sao-paulo-registra-25-casos-de-maus-tratos-a-animal-por-dia,70002636411>>. Acesso em: 13 Mar. 2019.

TRELLE, Tobias. **Spring Data – One API To Rule Them All?** InfoQ, 2012. Disponível em: <<https://www.infoq.com/articles/spring-data-intro/>>. Acesso em: 21 Ago. 2019.

UNRIC. Centro Regional de Informação das Nações Unidas. **17 Objetivos de Desenvolvimento Sustentável**. 2019. Disponível em <<https://www.unric.org/pt/17-objetivos-de-desenvolvimento-sustentavel>>. Acesso em: 21 Ago. 2019.

VALERIANO, Dalton. **Moderno gerenciamento de projetos**. São Paulo: Prentice Hall, 2005.

YELLOW BRICK. **Sobre o Evento**. Facebook, 2019. Disponível em:
<https://www.facebook.com/pg/yellowbrickteam/about/?ref=page_internal>. Acesso em:
30 Mar. 2019.

OBRAS CONSULTADAS

ANDRADE, T. F. **Validação com Bean validation**. Blog.Algaworks, 2013. Disponível em <<https://blog.algaworks.com/validacao-com-bean-validation/>>. Acesso em: 01 Set. 2019.

BENYON, David. **Interação Humano-Computador**. 2. ed. São Paulo: Pearson, 2011.

OLIVEIRA, E. C. M. **Convenções de Código Java**. Linha de código, 2019. Disponível em <<http://www.linhadecodigo.com.br/artigo/865/convencoes-de-codigo-java.aspx>>. Acesso em: 01 Set. 2019.

OPUS-SOFTWARE. *Design Patterns – O que são e quais os benefícios?*. 2018. Disponível em: <<https://www.opus-software.com.br/design-patterns/>>. Acesso em: 21 Ago. 2019.

PHPIT. **O que é um Framework? Para que serve?**. 2016. Disponível em <<http://www.phpit.com.br/artigos/o-que-e-um-framework.phpit>>. Acesso em: 21 Ago. 2019.

REDE GLOBO. Mais de 30 milhões de Animais Abandonados Esperam uma Chance. **TV Tribuna**, SP, 2018. Disponível em: <<https://redeglobo.globo.com/sp/tvtribuna/Caominhada/noticia/mais-de-30-milhoes-de-animais-abandonados-esperam-uma-chance.ghtml>>. Acesso em: 10 Fev. 2019.

APÊNDICE A - ENTREVISTAS

A entrevista realizada com as representantes das Instituições parceiras está disponível em formato PDF e encontra-se na pasta APÊNDICES do arquivo anexado a este documento.

APÊNDICE B – ESTRUTURA ANALÍTICA DO PROJETO

A EAP do projeto está disponível no formato JPEG e encontra-se na pasta APÊNDICES do arquivo anexado a este documento.

APÊNDICE C – CRONOGRAMA E DICIONÁRIO EAP

O cronograma de atividades do projeto está disponível na pasta APÊNDICES do arquivo anexado a este documento.

APÊNDICE D – RELATÓRIO DE DESEMPENHO

O relatório de desempenho está disponível na pasta APÊNDICES do arquivo anexado a este documento.

APÊNDICE E – LISTA DE VERIFICAÇÃO DA QUALIDADE

A planilha de verificação da qualidade deste projeto está disponível na pasta APÊNDICES do arquivo anexado a este documento.

APÊNDICE F – ACOMPANHAMENTO DOS RISCOS

A planilha onde consta o acompanhamento dos riscos do projeto está disponível na pasta APÊNDICES do arquivo anexado a este documento.

APÊNDICE G – DIAGRAMAS DE CASOS DE USO

Os diagramas de casos de uso do projeto estão disponíveis na pasta APÊNDICES do arquivo anexado a este documento.

APÊNDICE H – DESCRIÇÃO DOS CASOS DE USO

A descrição dos casos de uso do projeto está disponível na pasta APÊNDICES do arquivo anexado a este documento.

APÊNDICE I – DIAGRAMA ENTIDADE-RELACIONAMENTO

O diagrama entidade-relacionamento está disponível na pasta APÊNDICES do arquivo anexado a este documento.

APÊNDICE J – MODELO INICIAL DE INTERFACE DE USUÁRIO

Os modelos iniciais de interface estão disponíveis na pasta APÊNDICES do arquivo anexado a este documento.

APÊNDICE K – ESTIMATIVA DE ESFORÇO

As estimativas de esforço para o desenvolvimento do sistema estão disponíveis na pasta APÊNDICES do arquivo anexado a este documento.

APÊNDICE L – DIAGRAMA DE PACOTES

O diagrama de pacotes está disponível na pasta APÊNDICES do arquivo anexado a este documento.

APÊNDICE M – DIAGRAMA DE CLASSES

O diagrama de classes está disponível na pasta APÊNDICES do arquivo anexado a este documento.

APÊNDICE N – DIAGRAMA DE OBJETOS

O diagrama de objetos está disponível na pasta APÊNDICES do arquivo anexado a este documento.

APÊNDICE O – DIAGRAMA DE SEQUÊNCIA

Os diagramas de sequência estão disponíveis na pasta APÊNDICES do arquivo anexado a este documento.

APÊNDICE P – DIAGRAMA DE VISÃO GERAL DE INTERAÇÃO

Os diagramas de visão geral de interação estão disponíveis na pasta APÊNDICES do arquivo anexado a este documento.

APÊNDICE Q – MODELO LÓGICO

O modelo lógico (ou operacional) está disponível na pasta APÊNDICES do arquivo anexado a este documento.

APÊNDICE R – DICIONÁRIO DE DADOS

O dicionário de dados está disponível na pasta APÊNDICES do arquivo anexado a este documento.

APÊNDICE S – ENTREVISTAS (PERFIL DO USUÁRIO)

As entrevistas para perfil de usuário estão disponíveis na pasta APÊNDICES do arquivo anexado a este documento.

APÊNDICE T – DIAGRAMA DE ATIVIDADES

O diagrama de atividades está disponível na pasta APÊNDICES do arquivo anexado a este documento.

APÊNDICE U – SCRIPT DDL

O script *Data Definition Language* (DDL) está disponível na pasta APÊNDICES do arquivo anexado a este documento.

APÊNDICE V – SCRIPT DML

O script *Data Manipulation Language* (DML) está disponível na pasta APÊNDICES do arquivo anexado a este documento.

APÊNDICE W – MANUAL DO USUÁRIO

O manual do usuário está disponível na pasta APÊNDICES do arquivo anexado a este documento.

APÊNDICE X – DESCRIÇÃO E RESULTADO DOS CASOS DE TESTE

A descrição e os resultados dos casos de teste estão disponíveis na pasta APÊNDICES do arquivo anexado a este documento.

APÊNDICE Y – DIAGRAMA DE IMPLANTAÇÃO

O diagrama de implementação está disponível na pasta APÊNDICES do arquivo anexado a este documento.

APÊNDICE Z – DIAGRAMA DE COMPONENTES

O diagrama de componente está disponível na pasta APÊNDICES do arquivo anexado a este documento.

ANEXO A – TERMO DE ADOÇÃO

Os termos de adoção disponibilizados pelas instituições estão disponíveis no formato PDF e encontra-se na pasta ANEXOS do arquivo anexado a este documento.

ANEXO B – MANUAIS DE INSTALAÇÃO E CONFIGURAÇÃO

Os manuais de instalação e configuração estão disponíveis no formato PDF e encontra-se na pasta ANEXOS do arquivo anexado a este documento.