

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

CURSO DE SISTEMAS DE INFORMAÇÃO

LUCAS SILVA E DIAS

MATHEUS FELIPE DE SOUZA

VINICIUS DE ALMEIDA GONÇALVES

DEU PET

SANTA RITA DO SAPUCAÍ - MG

2022

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

CURSO DE SISTEMAS DE INFORMAÇÃO

LUCAS SILVA E DIAS

MATHEUS FELIPE DE SOUZA

VINICIUS DE ALMEIDA GONÇALVES

DEU PET

Projeto de final de curso apresentado a 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

2022

FOLHA DE APROVAÇÃO

[Na impressão final do documento, esta folha será substituída pela assinada pela Banca examinadora final.]

HISTÓRICO DE REVISÕES

Continua.

Data	Versão	Autor (es)	Descrição
21/12/2022	2.7	Vinicius Almeida	Correção apontamentos gerais
12/12/2022	2.6	Banca de defesa	Apontamentos gerais.
19/11/2022	2.5	Lucas Dias e Vinicius Almeida	Correção dos apontamentos e acertos dos textos para complementação da fase 4.
09/11/2022	2.4	Eunice Gomes	Apontamentos e acertos dos textos.
29/10/2022	2.3	Lucas Dias, Matheus Felipe e Vinicius Almeida	Correção da Fase 3 e entrega da Fase 4.
30/09/2022	2.2	Eunice Gomes	Correções nos textos e apontamentos para complementação da Fase 3.
10/09/2022	2.1	Lucas Dias, Matheus Felipe e Vinicius Almeida	Entrega da Fase 3.
25/06/2022	1.8	Lucas Dias, Matheus Felipe e Vinicius Almeida	Correção das indicações da banca de qualificação.
17/06/2022	1.7	Eunice Gomes, Silvana Lima e Wanderley Souza	Indicações da banca de qualificação.

Conclusão.

Data	Versão	Autor (es)	Descrição
04/06/2022	1.6	Lucas Dias, Matheus Felipe e Vinicius Almeida	Correção dos apontamentos da Fase 2.
25/05/2022	1.5	Eunice Gomes	Apontamentos para correção da Fase 2.
21/05/2022	1.4	Lucas Dias, Matheus Felipe e Vinicius Almeida	Correção dos apontamentos da Fase 1 e inclusão da Fase 2.
06/05/2022	1.3	Eunice Gomes	Apontamentos da Fase 1.
23/04/2022	1.2	Lucas Dias Matheus Felipe Vinicius Almeida	Correção dos apontamentos da Fase 1.
02/04/2022	1.1	Eunice Gomes	Apontamentos parciais para correção da Fase 1.
26/03/2022	1.0	Lucas Dias Matheus Felipe Vinicius Almeida	Entrega da Fase 1.

AGRADECIMENTOS

A Deus, pelo dom de nossas vidas, e por nos ajudar a ultrapassar todos os obstáculos. Ele que esteve conosco durante todo este caminho, dando força de vontade e coragem para superar os desafios e problemas.

À nossa família que nos apoiou e incentivou com paciência e compreensão. Nossas namoradas Aline Fernandes, Júlia do Prado e Sara Braga que nos ouviram e aconselharam nos momentos mais difíceis.

Gostaríamos de agradecer a todos os professores desta instituição de ensino que em muito contribuíram para a realização deste trabalho. Professores que com seus ensinamentos tornaram a nossa formação acadêmica possível. Agradecemos à professora e coordenadora que nos guiou pelo caminho deste projeto final de curso, sem o qual nada disso seria possível, à Eunice Gomes de Siqueira, nosso agradecimento especial. Obrigado pela dedicação e tempo despendido em nosso auxílio na realização deste projeto, o mundo precisa de mais professores como você.

Também queremos agradecer à FAI - Centro de Ensino Superior em Gestão, Tecnologia e Educação, seu corpo docente e seus funcionários que contribuíram direta e indiretamente para a conclusão deste trabalho. Todos demonstraram estar comprometidos com a qualidade e excelência do ensino.

RESUMO

A conscientização da população quanto à quantidade de animais não domiciliados, ao abandono e à superpopulação dos abrigos é essencial para a melhoria desse cenário e convívio em sociedade. A educação sob a guarda responsável e cuidados dos animais, o acesso às informações populacionais dos abrigos e a adesão às campanhas de bem-estar animal devem ser priorizadas. Assim, neste documento, encontra-se o detalhamento do projeto Deu Pet que teve como objetivo desenvolver um sistema de software para as Organizações Não Governamentais (ONGs) de proteção aos animais e para os cidadãos de modo que esses possam ter informações a respeito. A metodologia de trabalho contemplou estudos exploratórios, entrevistas com representantes de ONGs, gerenciamento do projeto, modelagem, codificação e testes de sistema seguindo o Paradigma Orientado a Objetos. Como resultados, obtiveram-se os artefatos necessários para a disponibilidade de um sistema de software para demonstração e uso do público de interesse.

Palavras-chave: Organização Não Governamental. Animal não domiciliado. Campanha de bem-estar animal. Projeto Final de Curso. Sistemas de Informação.

LISTA DE FIGURAS

FIGURA 1 - Foto do animal preso dentro do carro por 1 hora	22
FIGURA 2 - Mecanismo de cache	28
FIGURA 3 - Sistema em camadas.	29
FIGURA 4 - Telas do aplicativo Adota Pet Go.	31
FIGURA 5 - Telas do aplicativo Adota Fácil.	32
FIGURA 6 - Página principal do website Amigo não se compra	33
FIGURA 7 - Objetivos do Desenvolvimento Sustentável	38
FIGURA 8 - ODS 15	38
FIGURA 9 - Grupo de processos de gerenciamento de projetos.	40
FIGURA 10 - O Modelo Incremental	42
FIGURA 11 - Controle de versão do projeto usando o Github	47
FIGURA 12 - Logotipo para os componentes do Deu Pet.	72
FIGURA 13 - Exemplo de cores aplicadas no texto.	72
FIGURA 14 - Exemplo de textos com alto contraste.	72
FIGURA 15 - Análise do contraste entre as cores de fundo e de texto.	73
FIGURA 16 - Página inicial do aplicativo <i>Web</i> .	74
FIGURA 17 - Tela inicial do aplicativo <i>mobile</i>	75
FIGURA 18 - Menu principal do aplicativo <i>Web</i>	75
FIGURA 19 - Menu principal do aplicativo <i>mobile</i>	76
FIGURA 20 - Diálogo de confirmação de cadastro de animal.	77
FIGURA 21 - Tela de autenticação.	79
FIGURA 22 - Diálogo de confirmação de exclusão de conta.	79
FIGURA 23 - Orientação ao usuário para alterar dados.	80
FIGURA 24 – Documentação com Swagger.	85
FIGURA 25 - Sequência de autenticação com JWT.	86
FIGURA 26 – Um possível diagrama de distribuição.	89
FIGURA 27 - Diagrama de componentes do aplicativo <i>Web</i>	90
FIGURA 28 - Laço de repetição para recuperar todas as instituições	93
FIGURA 29 - Laço de repetição para recuperar todas as campanhas	94
FIGURA 30 - Laço de repetição para recuperar todas as imagens dos animais	94
FIGURA 31- Diagrama de implantação	102
FIGURA 32 - Deu Pet <i>mobile</i>	104

FIGURA 33- Tela de acesso Deu Pet Web 105

FIGURA 34 - 1º Lugar Prêmio Municipal de Inovação 2022 105

LISTA DE QUADROS

QUADRO 1 - Entrevista parcial com Rafael Ferrari de Souza, presidente da SPA de Santa Rita do Sapucaí - MG25

QUADRO 2 - Estimativa de esforços 45

QUADRO 3- Complexidade algorítmica 93

QUADRO 4 - Documentos relevantes para testes 95

QUADRO 5 - Equipamentos para a realização dos testes 96

QUADRO 6 - Softwares para a realização dos testes 96

QUADRO 7 - Papéis e responsabilidades na implantação 99

QUADRO 8 - Treinamentos previstos 100

QUADRO 9 - Cronograma de atividades da implantação 101

QUADRO 10 - Recursos de apoio à implantação 101

LISTA DE ABREVIATURAS E SIGLAS

ABNT	Associação Brasileira de Normas Técnicas
API	Application Programming Interface
ART	Artigo
CASE	Computer-Aided Software Engineering
CPF	Cadastro de Pessoa Física
DER	Diagrama Entidade e Relacionamento
DAO	Data Access Object
FAI	FAI - Centro de Ensino Superior em Gestão, Tecnologia e Educação
FAITEC	Feira de Tecnologia e Empreendedorismo da FAI
GO	Goiânia
HTML	HyperText Markup Language
HTTP	HyperText Transfer Protocol
IDE	Integrated Development Environment
MER	Modelo Entidade Relacionamento
MG	Minas Gerais
MVC	Model-View-Controller
MVP	Minimum Viable Product
Nº	Número
ODS	Objetivos do Desenvolvimento Sustentável

OMS	Organização Mundial de Saúde
ONG	Organizações Não Governamental
OSCIP	Organização da Sociedade Civil de Interesse Público
PCU	Pontos de Casos de Uso
PDF	Portable Document Format
PFC	Projeto Final de Curso
PMBok	Project Management Body of Knowledge
PMI	Project Management Institute
RAM	Random Access Memory
REST	Representation State Transfer
RF	Requisito Funcional
RNF	Requisito Não Funcional
SMTP	Simple Mail Transfer Protocol
SPA	Sociedade Protetora dos Animais
SGBD	Sistema Gerenciador de Banco de Dados
SPA	Sociedade Protetora dos Animais
UML	Unified Modeling Language
URL	Uniform Resource Location

SUMÁRIO

1 INTRODUÇÃO	18
2 REVISÃO BIBLIOGRÁFICA	20
2.1 POPULAÇÃO DE ANIMAIS NÃO DOMICILIADOS	20
2.2 PAPEL DAS ONGS NA PROTEÇÃO DOS ANIMAIS	22
2.3 MÉTODOS E TÉCNICAS APLICADOS AO PROJETO.....	26
2.3.1 Arquitetura Cliente-Servidor.....	26
2.3.2 Arquitetura REST.....	26
2.3.2.1 <i>Stateless</i> (Sem Estado)	27
2.3.2.2 Cache	28
2.3.2.3 Sistema em Camadas	29
2.3.4 Web Components	30
2.3.5 Spring MVC.....	30
2.4 TRABALHOS RELACIONADOS	30
2.4.1 Adota Pet GO	31
2.4.2 Adota Fácil.....	32
2.4.3 Amigo não se Compra.....	33
3 OBJETIVO DO PROJETO.....	34
3.1 FORMULAÇÃO DO PROBLEMA.....	34
3.2 OBJETIVOS	35
3.3 JUSTIFICATIVA	36

3.4 PÚBLICO DE INTERESSE	36
3.5 NÍVEIS DE DECISÃO E GRUPOS FUNCIONAIS ATENDIDOS	36
3.6 ADERÊNCIA AOS OBJETIVOS DO DESENVOLVIMENTO SUSTENTÁVEL	37
3.7 MODELO DE NEGÓCIO PROPOSTO.....	39
4 MÉTODOS GERENCIAIS	40
4.1 PLANO DE ELABORAÇÃO E GERENCIAMENTO DO PROJETO.....	40
4.2 MODELO DE CICLO DE VIDA.....	41
4.2.1 Modelo Incremental.....	41
4.3 RECURSOS NECESSÁRIOS	43
4.3.1 Recursos Humanos.....	43
4.3.2 Recursos de Software.....	43
4.3.3 Recursos de Hardware.....	44
4.4 RELATÓRIO DE DESEMPENHO.....	44
4.5 ESTIMATIVAS DE TAMANHO E ESFORÇO.....	44
4.6 GERENCIAMENTO DE CONFIGURAÇÃO	45
5 ESPECIFICAÇÃO E ANÁLISE DOS REQUISITOS DO SISTEMA	48
5.1 REQUISITOS DO SISTEMA DE SOFTWARE	48
5.1.2 Requisitos não Funcionais	60
5.1.2.2 Requisitos Organizacionais	61
5.1.2.3 Requisitos Externos	63
5.1.3 Principais Regras de Negócio	65

5.2 ANÁLISE DOS REQUISITOS	65
5.2.1 Visão Funcional	65
5.2.2 Modelo Conceitual dos Dados	66
5.2.3 Modelo Inicial da Interface de Usuário.....	66
6 ARQUITETURA E PROJETO DO SISTEMA DE SOFTWARE	67
6.1 VISÃO ESTRUTURAL	67
6.1.1 Diagrama de Pacotes.....	67
6.1.2 Diagramas de Classes.....	68
6.2 VISÃO COMPORTAMENTAL	68
6.2.1 Projeto das Interações entre Objetos	68
6.2.2 Diagramas de Atividades.....	69
6.3 VISÃO DOS DADOS	69
6.3.1 Modelo Lógico	69
6.3.2 Dicionário de Dados do Modelo Lógico	69
6.4 PROJETO DA INTERAÇÃO HUMANO-COMPUTADOR	70
6.4.1 Perfil de Usuário.....	70
6.4.2 Projeto da Interface de Usuário.....	71
6.4.3 Heurísticas de Usabilidade	76
6.4.4 Projeto da Acessibilidade	81
6.5 PROJETO DO SISTEMA DISTRIBUÍDO	82
6.5.1 Procedimentos para Tratamento dos Desafios	82

6.5.2 Tecnologias e Arquiteturas de Distribuição	88
7 IMPLEMENTAÇÃO DO SISTEMA DE SOFTWARE	90
7.1 COMPONENTES DO SISTEMA DE SOFTWARE	90
7.2 TECNOLOGIAS DE IMPLEMENTAÇÃO	91
7.2.1 Linguagens de Programação e <i>Frameworks</i> Adotados.....	91
7.2.2 <i>Design Patterns</i> Aplicados	91
7.2.3 Convenções e Guias para Codificação.....	91
7.2.4 Estrutura Física do Banco de Dados	92
7.3 ANÁLISE DE COMPLEXIDADE ALGORÍTMICA	92
8 PLANO DE TESTES	95
8.1 FINALIDADE	95
8.2 ESCOPO	95
8.2.1 Referências aos Documentos Relevantes,.....	95
8.2.2 Ambiente para a Realização dos Testes	96
8.3 ESPECIFICAÇÃO DOS CASOS DE TESTES	96
8.4 RESULTADOS DOS TESTES	97
9 PLANO PARA IMPLANTAÇÃO	98
9.1 METODOLOGIA	98
9.1.1 Descrição da Metodologia.....	98
9.1.2 Matriz de Responsabilidade	99
9.2 TREINAMENTO PREVISTO	100

9.3 CRONOGRAMA DE IMPLANTAÇÃO	100
9.4 RECURSOS DE APOIO À IMPLANTAÇÃO	101
9.5 VISÃO DA IMPLANTAÇÃO	102
10 CONCLUSÃO.....	103
REFERÊNCIAS	107
OBRAS CONSULTADAS	113
APÊNDICE A – PLANO DE ELABORAÇÃO E GERENCIAMENTO DO PROJETO	114
APÊNDICE B – RELATÓRIO DE DESEMPENHO	115
APÊNDICE C – VISÃO FUNCIONAL	116
APÊNDICE D – VISÃO DOS DADOS	117
APÊNDICE E – MODELO INICIAL DA INTERFACE DE USUÁRIO	118
APÊNDICE F – VISÕES ESTRUTURAL E COMPORTAMENTAL.....	119
APÊNDICE G – ENTREVISTAS COM USUÁRIOS	120
APÊNDICE H – ESTIMATIVA DE TAMANHO E ESFORÇO	121
APÊNDICE I – MODELO DE NEGÓCIO.....	122
APÊNDICE J – CASOS DE TESTES	123
APÊNDICE K – RECURSOS DE APOIO À IMPLANTAÇÃO	124
APÊNDICE L – ENTREVISTA COM A SOCIEDADE PROTETORA DOS ANIMAIS	125

1 INTRODUÇÃO

Nas paisagens urbanas, além da presença de edifícios e viadutos, cães e gatos não domiciliados são tão frequentes que já se tornam integrantes desse meio. Mesmo sem contagens precisas, a Organização Mundial da Saúde (OMS) estima que milhões de animais, sendo a maioria cães e gatos, estão em estados de abandono, maus tratos ou desamparados.

Para a subsistência de muitos animais em situação de rua, são necessárias ações da vigilância sanitária municipal, bem como o auxílio da sociedade protetora dos animais, abrigos e demais organizações não governamentais (ONGs). Contudo, devido à alta demanda, superlotação, falta de recursos e de controle de saúde ambiental, é importante aplicar ferramentas capazes de contribuir com tais atividades e acelerar o processo de adoção de um animal, visando aliviar a carga de trabalho dessas instituições.

Perante o exposto, este projeto, nomeado Deu Pet, visa o desenvolvimento de um sistema de *software* que será disponibilizado em três componentes, sendo eles um aplicativo *Web*, um aplicativo para plataforma móvel (*mobile*) e uma *Application Programming Interface* (API). Por meio dele, será facilitada a divulgação dos animais que estão nos abrigos à espera de doação e também será possível o acompanhamento e inscrição do usuário em campanhas gerais relacionadas ao bem-estar dos animais e que são promovidas pelas instituições protetoras.

A metodologia envolve uma revisão bibliográfica sobre a problemática em questão, aprofundamento na prática das tecnologias requeridas para o desenvolvimento do sistema, aplicação de processos de gerência com referência à literatura da área, especificação de requisitos e modelagem baseada na *Unified Modeling Language* (UML) e construção e testes de software orientado a objetos.

Este documento se divide em 10 capítulos, sendo que o primeiro capítulo apresentado é esta introdução.

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

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

O Capítulo 4 apresenta o plano e a gerência do projeto e aborda os métodos gerenciais aplicados neste trabalho.

O Capítulo 5 apresenta os requisitos funcionais e não funcionais do sistema de software Deu Pet, também mostra os modelos de casos de uso, conceitual dos dados e da interface de usuário.

O Capítulo 6 apresenta a arquitetura do sistema de software e as visões estrutural, comportamental, de dados e das interações.

O Capítulo 7 apresenta informações sobre a implementação do sistema de software e contém os componentes, tecnologias de codificação e uma análise de complexidade algorítmica.

O Capítulo 8 apresenta um dos planos de testes elaborado para avaliar o sistema e os resultados obtidos por meio deles.

O Capítulo 9 apresenta o plano da implantação, sendo descritas a metodologia de implantação, os treinamentos previstos para que os usuários utilizem corretamente o sistema, o cronograma, a documentação de apoio à implantação do sistema e uma visão da implantação.

O Capítulo 10 apresenta os principais resultados obtidos com a realização deste projeto, as lições aprendidas e as propostas de complementação do trabalho, sendo seguido das referências, obras consultadas, apêndices e anexos.

2 REVISÃO BIBLIOGRÁFICA

Este capítulo tem como objetivo apresentar o referencial teórico utilizado para que este projeto seja realizado. Demonstra-se a relação entre o homem e o animal domesticado a fim de entender as demandas dos interessados e traz informações a respeito das ONGs e instituições protetoras dos animais, para elucidar o tema e realizar uma abordagem mais incisiva nas necessidades apontadas.

Além dos conteúdos supracitados, também apresenta o embasamento teórico a respeito das tecnologias aplicadas.

2.1 POPULAÇÃO DE ANIMAIS NÃO DOMICILIADOS

Os animais domésticos possuem um papel importante na vida do homem desde há milhares de anos. Na realidade, desde a origem da civilização humana que as pessoas dependem da interação que estabelece com outras espécies animais, tendo essa relação sido inicialmente de predação, e posteriormente de domesticação (WALSH, 2009).

Contudo, uma das práticas muito recorrentes consiste no abandono de animais. Em especial, os domésticos como cães e gatos, o que afeta de maneira cada vez mais os grandes centros (TRINDADE *et al.*, 2013). Há um gradativo aumento da população de cães e gatos não domiciliados, do qual decorrem problemas pertinentes à saúde pública, trânsito, transtorno na coleta de lixo, além de ataques de animais ferozes em praças e logradouros de grande movimento. Neste sentido, são necessárias maneiras de abrigar esses animais e aplicar métodos que levem à diminuição da natalidade entre eles.

Abandonar ou maltratar animais é crime previsto pela Lei Federal nº. 9.605, de 12 de fevereiro de 1998, em seu art. 32: “Praticar ato de abuso, maus-tratos, ferir ou mutilar animais silvestres, domésticos ou domesticados, nativos ou exóticos” (BRASIL, 1998). E uma nova legislação, a Lei Federal nº 14.064, sancionada em setembro de 2020, aumentou a pena de detenção que era de até um ano para até cinco anos para quem cometer tal crime. Além disso, o rito processual passa à vara criminal, não mais ao juizado especial (BRASIL, 2020).

Sobre a população animal no Brasil, o Instituto Pet Brasil (2019) indica que a população de animais domésticos no Brasil é de cerca de 140 milhões de animais, entre cães, gatos, peixes,

aves e répteis e pequenos mamíferos. A maioria é de cachorros (54,2 milhões) e felinos (23,9 milhões), num total de 78,1 milhões de animais. Desses, 5% são animais em condição de vulnerabilidade, o que representa 3,9 milhões de pets. Do total da população em vulnerabilidade, cães representam 69% (2,69 milhões), enquanto os gatos correspondem a 31% (1,21 milhões).

E esse número elevado favorece maus tratos aplicados, em que os animais são surrados, envenenados e passam fome e frio. Devido a essas condições precárias, cães e gatos de rua se tornam transmissores de doenças para a população humana.

Os cães com maiores riscos de abandono são aqueles com problemas comportamentais, obtidos de abrigos ou a baixo custo, com idade igual ou superior a seis meses, não castrados e também os que não frequentaram cursos de obediência (adestramento) (PATRONEK *et al.*, 1995). De acordo com Santana e Marques (2001), esse quadro se agrava a cada dia, pois são milhares de cadelas e gatas parindo, aproximadamente, a cada três meses de gestação, dificultando o controle.

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 atualmente. 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.

Os humanos devem agir com os animais pelo princípio da analogia: “qualquer procedimento ou lesão que seja considerada dolorosa para seres humanos adultos, também o é para animais, mesmo quando não há uma evidência patente do comportamento doloroso” (SILVA *et al.*, p. 1, 2018).

Crime de maus-tratos aos animais é submeter ao abandono e agressões físicas, como: espancamento, mutilações, envenenamento; manter o animal preso a correntes ou cordas; manter o animal em locais não arejados, sem ventilação ou entrada de luz; manter o animal trancado em locais pequenos e sem o menor cuidado com a higiene; utilizar animais em

espetáculos que possam submetê-los a pânico ou estresse; capturar animais silvestres, dentre outras ações cruéis (SALLES, 2015).

De acordo com Santana (2020), um caso de repercussão pública expressiva ocorreu em Goiânia (GO), em 2020, quando um cachorro foi deixado trancado por mais de uma hora dentro de um carro, como mostra a Figura 1. O animal estava latindo com o calor. Comerciantes da região chamaram a Polícia Militar e o Corpo de Bombeiros para retirarem o cão. O caso foi registrado na polícia.



FIGURA 1 - Foto do animal preso dentro do carro por 1 hora

FONTE: Santana (2020)

2.2 PAPEL DAS ONGS NA PROTEÇÃO DOS ANIMAIS

A superpopulação de animais abandonados demanda, cada vez mais, a atuação de ONGs e de protetores independentes que normalmente abrigam mais animais do que tem condições financeiras de manter (SOUZA; REIS, 2020, p.1).

Boa parte das ações solidárias se deve ao fato dos movimentos que acontecem pela internet, como apelos, campanhas, e movimentos que protestam em prol de diversas causas.

Uma das pautas mais levantadas nos últimos tempos é a causa animal. Um mutirão de pessoas se mobiliza sempre que há algum tipo de notícia relacionado a maus tratos e abandono de

animais. Um exemplo de mobilização realizada, é a tentativa de proibição dos fogos de artifício, que para os nossos “*pets*”, são extremamente nocivos.

Parte dessas pessoas se comoveu com várias causas, tornando-se protetores independentes que se voluntariam a ajudar de várias formas as ONGs, como a Sociedade Protetora dos Animais (SPA), que através de redes sociais como o *Facebook* e o *Whatsapp* somam dezenas de pessoas voluntárias por cada ONG de cada município.

Segundo Moutinho, Serra e Valente (2019), o controle populacional se faz necessário por questões relacionadas à saúde coletiva, ao bem-estar animal e à ordem urbana. A grande quantidade de cães soltos nas ruas pode provocar acidentes de trânsito, destruição dos patrimônios público e privado, maus-tratos contra estes animais, doenças transmissíveis e agressões a pessoas.

Para controle populacional dos cães e gatos, eram utilizadas estratégias como a eutanásia o que veio a se tornar proibida em alguns estados do Brasil. Diversos municípios brasileiros dos estados de São Paulo, Rio de Janeiro e Minas Gerais adotaram a esterilização como estratégia de controle ao invés da eutanásia.

O abandono de animais se tornou um problema ainda maior para abrigos, ONGs e projetos que acolhem animais não domiciliados em todo o país. Levantamento da ONG Amapara, uma Organização da Sociedade Civil de Interesse Público (OSCIP) que presta ajuda às ONGs e aos protetores independentes da causa animal, concluiu que o número de bichos abandonados no Brasil subiu 61,6% entre julho de 2020 e fevereiro deste ano no país, em relação ao mesmo período do ano anterior (LINARD, 2022).

Para esclarecer mais o funcionamento de uma instituição, foi realizado no dia 15 de março de 2022, uma entrevista com Rafael Ferrari de Souza, responsável legal e presidente da Sociedade Protetora dos Animais (SPA) de Santa Rita do Sapucaí- MG. O entrevistado detalha alguns pontos importantes sobre o dia a dia da instituição, como mostrados no Quadro 1.

Continua.

	Sociedade Protetora dos Animais SRS
Bloco 1	
Quais animais são resgatados pela SPA?	Cães e gatos.
Quais os critérios para resgate desses animais?	Os cães a ser resgatados devem ser cães mordedores ou cadelas prenhas.
Quais os meios de contato para efetuação dos resgates hoje?	Todos os contatos são realizados via telefone, Whatsapp e Facebook.
Bloco 2	
Por onde conseguimos adotar os animais?	As adoções dos animais são por meio do contato com a SPA, via telefone, Whatsapp e Facebook.
Como funciona o processo de adoção dos animais?	Os critérios para adoção são: ser maior de 18 anos; passar por entrevista prévia (que funciona por formulário ou por ligação); assinar um termo de responsabilidade sob o animal (processo que ocorre após a aprovação da entrevista)
Caso o animal não consiga se adaptar, o que fazer?	Caso não dê certo a adoção, é possível devolver o animal, porque a adoção é como se fosse em forma consignada.

Conclusão.

Bloco 3	
Hoje a SPA participa ativamente das campanhas em prol da saúde animal?	Sim, a SPA realiza um trabalho conjunto com a prefeitura. Este ano serão realizadas 1000 castrações por investimento do município.
Como funciona a inscrição para essas campanhas?	A inscrição é feita de duas formas: pelo Google Forms ou direto na vigilância sanitária (para atender todos os públicos)
Quais são os critérios que fazem com que os animais entrem numa fila de baixa prioridade ou na reserva?	Os critérios são: idade maior que 8 anos; serem machos. Caso os animais sejam braquicéfalos (focinho curto/problema respiratório) são barrados diretamente na seleção.
Quais os critérios para entrarem na fila de alta prioridade?	Fêmeas que dividem o mesmo ambiente com um macho e entram na fila de alta prioridade.

QUADRO 1 - Entrevista parcial com Rafael Ferrari de Souza, presidente da SPA de Santa Rita do Sapucaí - MG

Em outro ponto chave da entrevista, que pode ser consultada na íntegra junto ao Apêndice L, Souza (2022, p.1) afirma que embora a dificuldade com relação a superlotações das ONGs, houve um aumento expressivo no número de ativistas da causa animal que buscam e lutam para conseguir mais espaço político e cada vez recolher mais recursos para que possam realizar campanhas de vacinação, castração, resgate e conscientização populacional que se torna um incentivo para que mais pessoas participem de forma voluntária a causa animal (SOUZA, 2022, p. 1).

2.3 MÉTODOS E TÉCNICAS APLICADOS AO PROJETO

A seguir são introduzidas algumas tecnologias de informação e comunicação importantes para a realização deste projeto.

2.3.1 Arquitetura Cliente-Servidor

Fielding (2000) define uma arquitetura cliente-servidor como um tipo de distribuição de responsabilidades de processamento, no qual um processo servidor é o fornecedor de um serviço e o processo cliente é o consumidor. Para isso, a resposta de uma requisição de um processo cliente em feito em um formato padronizado e por meio de protocolos específicos. O cliente ao solicitar um dado ao servidor pode não ter conhecimento da natureza desses dados. Esse tipo de acesso aos dados cria uma forma padronizada de solicitar e tratar os mesmos, simplificando a manutenção e portabilidade. Dessa forma, cria-se um cenário onde uma mesma base de dados pode ser compartilhada entre as aplicações, sejam elas *Web* ou *mobile*, por exemplo.

Uma das formas pelas quais uma arquitetura cliente-servidor pode ser implementada é destacada a seguir.

2.3.2 Arquitetura REST

Representation State Transfer (REST) é um estilo de arquitetura proposto por Fielding (2000) que visa relacionar de forma condizente a *Web* com os protocolos *Text Transfer Protocol* (HTTP) por meio de estilos arquiteturais.

Inúmeros serviços utilizados no cotidiano, como acesso a um portal *Web*, compras em lojas de *e-commerce*, *websites* de *streaming*, entre outros, demandam dos servidores o atendimento a inúmeras requisições de aplicações com as mais diversas tecnologias.

Visando a comunicação entre as partes de forma a unificar a maneira como os dados são requisitados, os *Web Services* são componentes de *software* que, por meio de padrões previamente estabelecidos, fornecem serviços específicos e promovem trocas de dados entre

sistemas, independentemente das linguagens de programação utilizadas na construção das aplicações envolvidas (DEITEL; DEITEL, 2010).

A arquitetura REST tem suporte dos principais ambientes de desenvolvimento de aplicações, o que proporciona uma gama de soluções em *Web Services* que podem atender as mais diversas necessidades. REST possibilita que os serviços sejam acessados por meio de requisições feitas por *Uniform Resource Location* – Localização Uniforme de Recursos (URL). Essa requisição é direcionada da aplicação cliente conforme definido, provendo orientações dinâmicas quanto à forma de construção e endereçamento das requisições (RIBEIRO; FRANCISCO, 2016).

Para este projeto, são tratados três pontos importantes da arquitetura REST, destacados a seguir.

2.3.2.1 *Stateless* (Sem Estado)

Stateless acontece quando as requisições feitas ao servidor não têm seus estados mantidos, criando a independência entre as requisições anteriores. Portanto, cada requisição cliente-servidor deve conter todos os dados necessários para que a solicitação possa ocorrer e conseqüentemente, impede que requisições usufruam de qualquer contexto previamente armazenado no servidor.

Essas características, segundo Fielding (2000), aumentam a transparência, confiabilidade e escalabilidade da aplicação, já que uma nova aplicação poderá requisitar dados sem a dependência de outra e a resposta não terá nem uma interferência da outra. Em uma situação quando um cliente deseja realizar uma solicitação ao servidor, o cliente deve primeiramente encaminhar um *token* contendo sua solicitação de acesso, assim o servidor o responde validando esse *token*, o qual posteriormente é utilizado pelo cliente para validar posteriores solicitações realizadas.

Em contrapartida, *stateless* provocam uma sobrecarga no desempenho da rede, devido ao fluxo de dados repetidos e de aumento por interação, já que os dados não podem ser armazenados em um estado anterior e inseridos em um contexto compartilhado (FIELDING, 2000).

Para contornar este problema, a arquitetura REST deve possibilitar maneiras para que as requisições sejam armazenadas em *cache*.

2.3.2.2 Cache

O *cache* é uma estrutura de armazenamento que visa promover o balanceamento das cargas de requisições para otimizar o acesso aos dados. O *cache* evita que o servidor responda às solicitações repetidas e gere um fluxo de dados desnecessário, reduzindo o congestionamento do canal de comunicação em rede. O *cache* pode ser usado também no servidor que salva recursos frequentemente acessados em memória ou disco, respondendo ao cliente em um tempo menor do que se fizesse uma consulta ao banco de dados ou servidor externo (KUWANO, 2018).

A limitação dessa abordagem é que só podem ser ‘cacheados’ métodos idempotentes (que não alteram os dados e retornam respostas idênticas ao longo do tempo). Em termos REST, consultas *GET* podem ser salvas em cache enquanto alteração de dados com *POST*, *PUT*, *DELETE* são exclusivamente processadas pelo servidor, e uma vez que alteram o estado dos dados, os dados não podem ser ‘reaproveitados’ pelo *cache* (KUWANO, 2018).

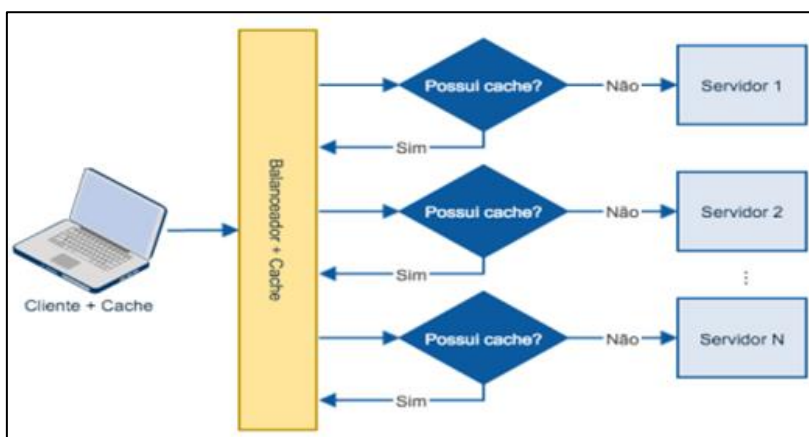


FIGURA 2 - Mecanismo de cache

Fonte: Dias (2016, p. 14).

Como demonstrado na Figura 2, caso um cliente efetue uma requisição ao servidor, ele tem esses dados armazenados em *cache* em sua máquina local, posteriormente, ao realizar uma consulta igual e, caso não haja alterações nos dados, é realizada uma consulta ao *cache* e

sendo igual, o servidor não irá prover novamente os mesmos dados, otimizando assim o acesso e evitando o congestionamento da rede.

Como preocupação, tem-se que para manter a confiabilidade dos dados é preciso realizar verificações constantes, já que o servidor pode ter seu estado alterado.

2.3.2.3 Sistema em Camadas

Um sistema arquitetado em camadas proporciona a possibilidade de se isolar a ação de cada componente, direcionando-os para apenas as camadas necessárias para a realização de suas operações. Este método aumenta a segurança a partir do encapsulamento dos serviços e o desempenho da aplicação a partir do balanceamento de carga de processamento dos componentes (VELASCO, 2018).

Como demonstrado na Figura 3, um sistema em camadas realiza primeiramente o acesso à camada de segurança e posteriormente é direcionado para a camada de *cache* onde ocorre a verificação de dados já armazenados. Caso a solicitação não esteja presente na camada de *cache*, o servidor encaminha a requisição para as camadas posteriores.

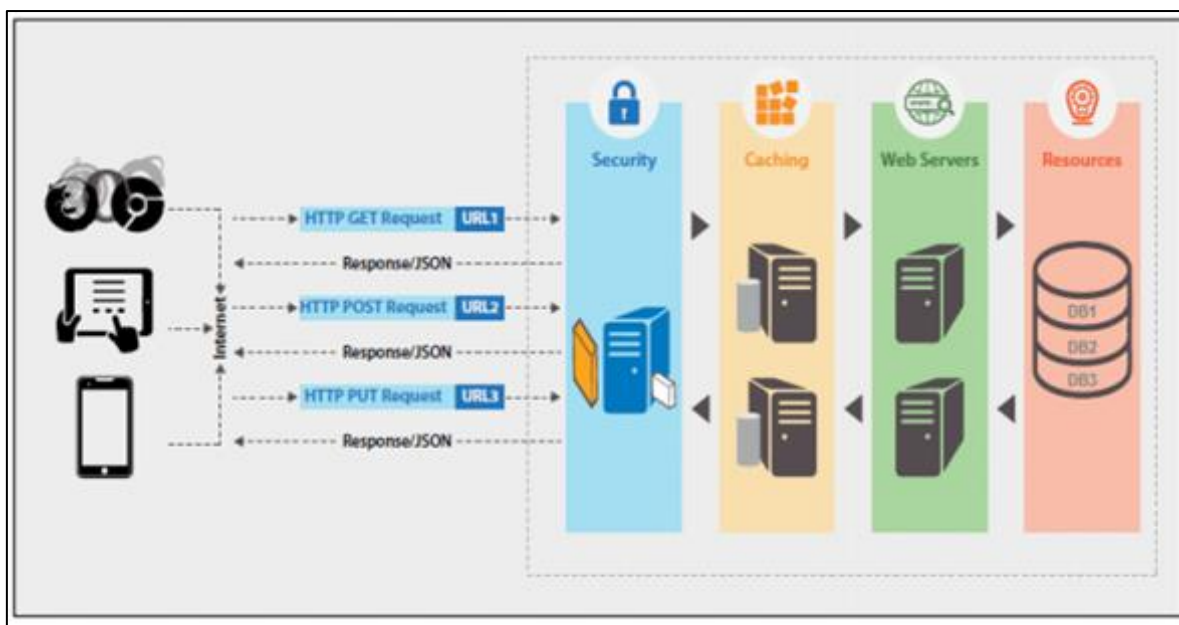


FIGURA 3 - Sistema em camadas.

Fonte: Deepak (2015).

2.3.4 Web Components

Web Components é o conjunto de diversas tecnologias reutilizáveis e customizáveis para aplicações *Web* que permitem a utilização dos elementos de forma independente do restante do restante do código (MOZILLA, 2018).

A prática de reutilização de códigos é algo recorrente por tornar o processo de desenvolvimento ágil e também por criar um padrão de *design* nos projetos. Contudo, em certos casos como a implementação HTML e seus variados tipos de *scripts* e estilos, o que pode tornar algo mais complexo. Os *web components* buscam uma maneira prática de solucionar esses problemas.

2.3.5 Spring MVC

O *framework Spring* é um dos *frameworks* utilizado no desenvolvimento de aplicações com linguagem de programação Java, sendo um subsistema *Spring*. Ele possibilita a implementação de um grande número de funções, tais como injeção de dependência, persistência de dados e uma implementação para o padrão *Model-View-Controller* (MVC) voltada para a criação de aplicações *Web* (DEV MEDIA, 2014). MVC visa modularizar os componentes e criar um fluxo de interação entre os mesmos sem que interfira na implementação de outras partes. Esta abordagem proporciona um desenvolvimento mais prático e distribuído.

O *framework Spring* possui uma vasta quantidade de bibliotecas flexíveis que possibilitam a criação de qualquer tipo de solução. A inversão de controles e injeção de dependências proporciona inúmeras possibilidades e recursos para projetos orientados a microserviços, assíncronos, em nuvem, entre outros (SPRING.IO, 2022).

2.4 TRABALHOS RELACIONADOS

Nesta seção são apresentadas aplicações de software similares às propostas apresentadas pelo projeto Deu Pet. São citadas três soluções que oferecem serviço de adoção de animais. São elas: Adota Pet GO, Adota Fácil e Amigo não se compra.

2.4.1 Adota Pet GO

O Adota Pet GO é um aplicativo *mobile*, de licença gratuita, criado por Marlon Henrique Ramalho Afonso, com apoio de Ruben Santos de Almeida, e orientado pelo seu professor Orion Teles.

De acordo com o Adota Pet Go (2022), trata-se de um aplicativo para “adoção de animais que segundo OMS, no Brasil, tem 30 milhões de animais vivendo nas ruas, sendo eles por maioria gatos e cachorros, então é esse o valor que teremos sempre em mente, e trabalharemos até que ele seja o mais próximo de zero possível”.

No aplicativo, um usuário pode publicar os dados de um animal para disponibilizá-lo para adoção. Pode-se cadastrar nome, fotos, descrição, espécie, raça e localização. Os usuários adotantes recebem uma lista de animais que estão mais próximos, usando o sistema de geolocalização do aplicativo e conseguem conversar com o cuidador temporário do animal por meio de um *chat* (ADOTA PET GO, 2022).

O aplicativo está disponível na Google Play, uma loja de aplicativos para dispositivos móveis com Android. A Figura 4 mostra algumas das telas da aplicação.

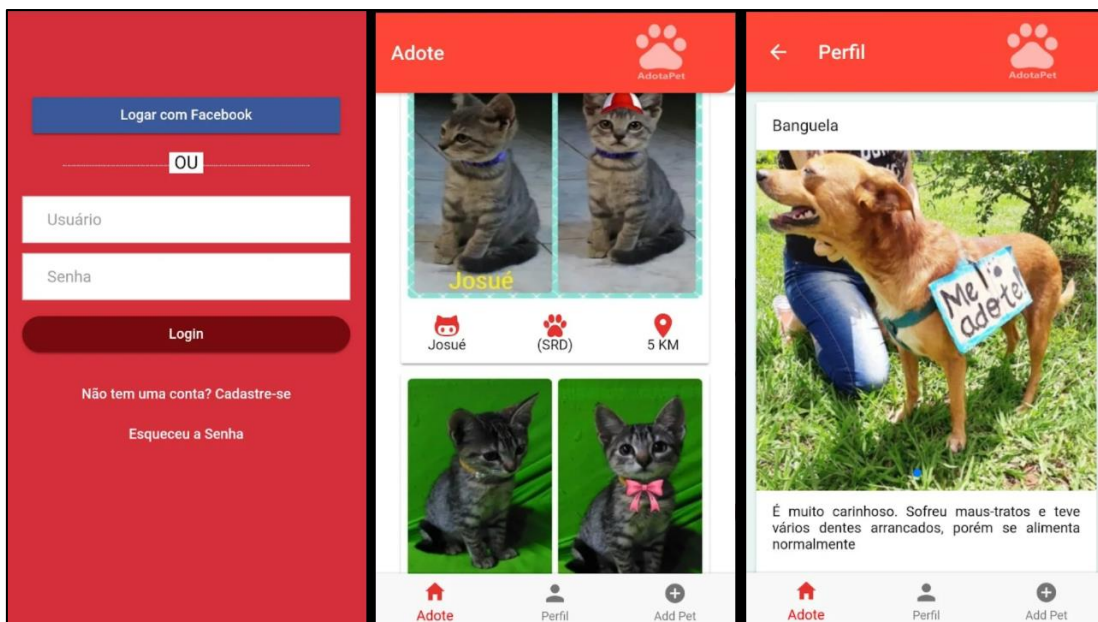


FIGURA 4 - Telas do aplicativo Adota Pet Go.

Fonte: Adota Pet GO (2022).

2.4.2 Adota Fácil

O Adota Fácil é uma aplicação *mobile*, de licença gratuita, criado por Daniel da Silva Gomes Neto, com o objetivo de facilitar a adoção de cães e gatos entre pessoas e ONGs no Brasil.

Ao se cadastrar ou entrar com os dados de uma rede social, o aplicativo listará todos os animais disponíveis para adoção e com possibilidade de busca por distância e raça que deseja adotar (ADOTA FÁCIL, 2022).

No site onde é possível realizar o *download* do aplicativo, consta que se uma pessoa não conseguir mais tomar conta de um animal, é possível publicar os detalhes desse e tentar encontrar alguém que o queira. Quando houver um interesse mútuo, ambas as partes poderão se conectar por meio do *chat* no aplicativo, facilitando assim, a adoção. Também é possível avaliar os usuários para que haja uma confiança antes de adotar o animal de alguém.

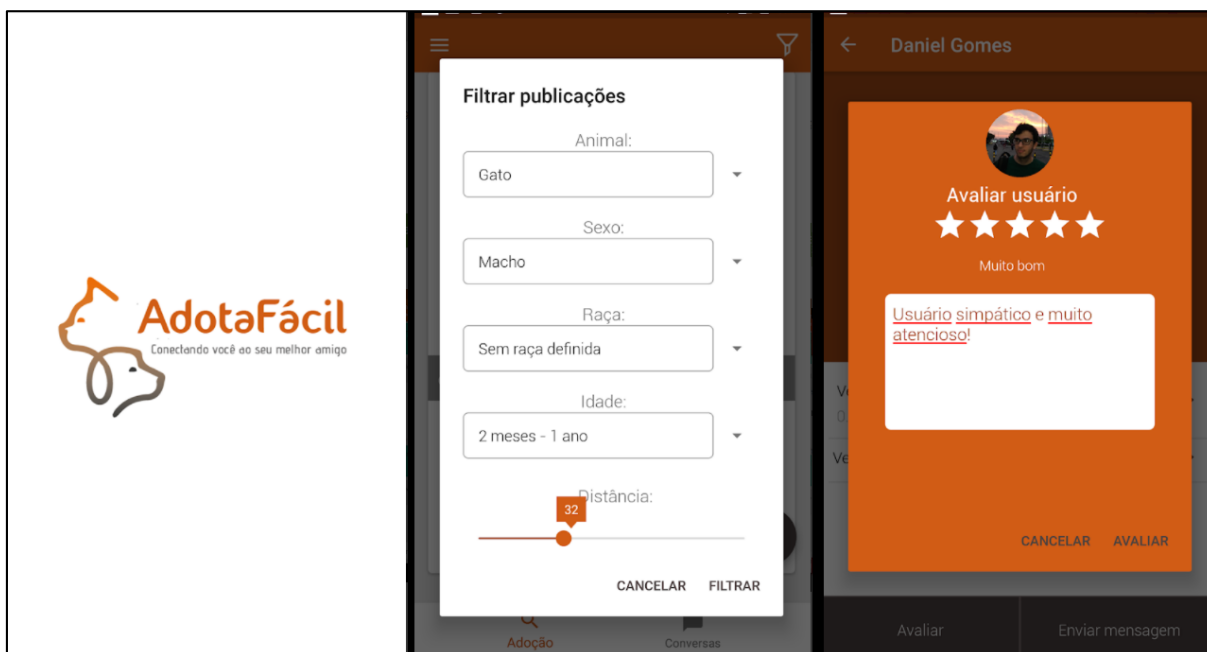


FIGURA 5 - Telas do aplicativo Adota Fácil.

Fonte: Adota Fácil (2022).

2.4.3 Amigo não se Compra

O Amigo não se Compra é uma associação privada com sede no município do Rio de Janeiro. A associação possui um *website* pela qual ONGs podem divulgar cães e gatos para adoção. Nesse *website*, encontra-se uma aplicação *Web* que funciona da seguinte forma:

- ONGs ou protetores fazem o seu cadastro gratuitamente e no mesmo instante já podem publicar os animais com detalhes sobre suas características e personalidades;
- pessoas que estão interessadas em adotar devem acessar o *website* e procurar um que tenha a ver com o perfil dele;
- quando se decidem, elas devem informar por meio da opção “quero adotar” e depois combinar com a ONG como buscar o animal.

A Figura 6 mostra a página principal do *website* da associação “Amigo não se compra”.

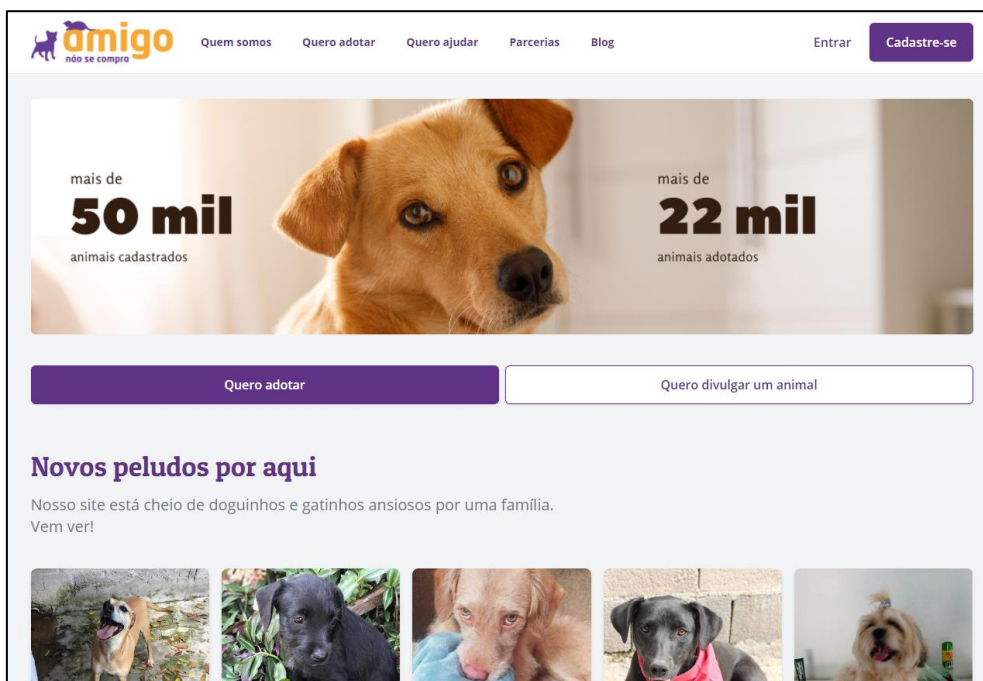


FIGURA 6 - Página principal do website Amigo não se compra

Fonte: Amigo não se compra (2022)

Apresentado o referencial teórico, no capítulo a seguir estão detalhados os objetivos do presente projeto.

3 OBJETIVO DO PROJETO

Neste capítulo é apresentada a formulação do problema, que enfatiza a necessidade da realização deste projeto, os objetivos almejados, a justificativa, o público de interesse e os níveis de decisão e grupos funcionais atendidos.

3.1 FORMULAÇÃO DO PROBLEMA

A excessiva quantidade de animais não domiciliados presentes na área urbana, de modo geral, acarreta em um problema de saúde pública e ambiental, pois pode levar à proliferação de zoonoses, problemas com a coleta de lixo e ataques a pedestres e condutores devido à agressividade dos animais. Não obstante esses desafios, a superlotação dos abrigos e pouca arrecadação de fundos para as instituições que realizam resgates são dificuldades a serem enfrentadas pela sociedade.

Uma entrevista realizada com Rafael Ferrari Souza, presidente da Sociedade Protetora de Animais (SPA) de Santa Rita do Sapucaí – MG, permitiu conhecer e analisar as atividades e entender as dificuldades dessas instituições. A entrevista pode ser consultada na íntegra junto ao Apêndice L.

Com base no Quadro 1, é possível averiguar que a visibilidade e meios de contato da SPA é toda realizada por meio de redes sociais e telefone, sem um recurso tecnológico mais sistêmico. Também é possível perceber que o processo de adoção é realizado manualmente pela instituição.

Outro desafio é o processo de castração gratuita de animais que é realizado pela Prefeitura Municipal, com apoio da SPA. Como o número de pedidos para castrações é alto, o processo é considerado moroso, uma vez que é preciso atender a alguns requisitos e mesmo dentro dos requisitos, existe uma fila de prioridade de acordo com a situação de cada animal. Por exemplo, em uma campanha onde existam 5 mil solicitações para castração, a SPA analisa individualmente cada solicitação, a fim de conferir se o animal atende aos requisitos e, então, classificar a solicitação na fila de prioridade. Para as pessoas interessadas, existe a dificuldade de precisar entrar no *website* da prefeitura para realizar uma solicitação e o desconhecimento dos requisitos que podem descartar imediatamente a participação de um determinado animal

da campanha de castração. Ademais, a população pode não conhecer a fila de prioridade e seus critérios para saber sua real chance de participar da campanha.

Os desafios aqui citados permitem entender que uma solução automatizada por meio de um sistema de software pode contribuir com a redução das dificuldades enfrentadas pelas instituições, de modo geral.

3.2 OBJETIVOS

O objetivo geral deste projeto de conclusão de curso é desenvolver um sistema de software, para uso com tecnologias *Web* e *mobile*, a fim de oferecer às ONGs, abrigos ou associações protetoras de animais, um sistema automatizado para a divulgação dos animais resgatados e disponíveis para adoção, bem como apresentar campanhas públicas ou privadas para o bem-estar animal.

Os objetivos específicos são:

- a) criar uma aplicação *Web* destinada a apoiar as instituições voltadas para o bem-estar animal;
- b) criar um aplicativo *mobile*, destinado às pessoas interessadas em adoção e ou nas campanhas de bem-estar animal.

Por meio de um aplicativo *Web*, será intermediado o processo de adoção e divulgação de campanhas de bem-estar animal pelas instituições protetoras. Pelo aplicativo *mobile* será possível que pessoas interessadas na adoção de animais encontrem aqueles que estão disponíveis para adoção em sua cidade; e que pessoas interessadas nas campanhas para animais se cadastrem para participar das listas de seleção.

O sistema de software deverá permitir aos seus usuários:

- a) o cadastro de animais resgatados pelas instituições protetoras;
- b) a divulgação de campanhas para animais pelas instituições protetoras;
- c) o acesso ao público que busca por adoção de animais;
- d) o acesso ao público que busca por campanhas de bem-estar para os animais.

3.3 JUSTIFICATIVA

Com base na entrevista apresentada no Quadro1, é possível averiguar que entre as principais dificuldades, então a falta de recursos financeiros e muitos animais vivendo nos abrigos, decorrentes da dificuldade na adoção dos mesmos.

Visto isso, um sistema que facilite a divulgação dos animais abrigados e as campanhas de bem-estar animal para o público interessado na adoção ou participação das campanhas, ajudará essas instituições a aumentar o fluxo de adoção, abrindo, assim, possibilidade de resgate de outros animais.

3.4 PÚBLICO DE INTERESSE

O sistema de software a ser desenvolvido tem como público de interesse ONGs, abrigos e associações protetoras dos animais e também pessoas que desejam adotar um animal ou participar das campanhas voltada para o bem-estar animal.

3.5 NÍVEIS DE DECISÃO E GRUPOS FUNCIONAIS ATENDIDOS

O sistema de software atuará apenas com informações em nível operacional, de acordo com o seguinte:

- a) o usuário interessado na adoção irá visualizar os dados dos animais, escolhendo aquele que mais lhe agrada. Entretanto, deverá se submeter a uma entrevista com a instituição para finalizar o processo de adoção;
- b) o usuário interessado na participação das campanhas de bem-estar irá sinalizar interesse por determinada campanha. Entretanto, será colocada em uma lista de seleção, onde a instituição responsável pela campanha, dado os critérios estabelecidos pela mesma, escolherá os animais mais urgentes para participação da campanha;
- c) os colaboradores da instituição irão disponibilizar as informações dos animais no abrigo, disponibilizar as campanhas para bem-estar animal disponíveis e estabelecer os critérios para participar das campanhas.

Os grupos funcionais atendidos são as ONGs, instituições protetoras dos animais e abrigos, que anunciarão seus animais para adoção e campanhas via a aplicação *Web* e os usuários adotantes ou interessados nas campanhas acessarão a aplicação *mobile* para escolher algum animal para adoção ou participar de alguma campanha promovida pelas entidades.

3.6 ADERÊNCIA AOS OBJETIVOS DO DESENVOLVIMENTO SUSTENTÁVEL

Os Objetivos de Desenvolvimento Sustentável (ODS) estão presentes em uma agenda mundial adotada durante a Cúpula das Nações Unidas sobre o Desenvolvimento Sustentável, realizada no ano de 2015. A agenda é composta por 17 objetivos e 169 metas a serem atingidas até o ano de 2030. Trata-se de um apelo global à ação para acabar com a pobreza, proteger o meio ambiente e o clima e garantir que as pessoas, em todos os lugares, possam desfrutar de paz e de prosperidade. Os temas podem ser divididos em quatro dimensões principais (ESTRATÉGIA ODS, 2016):

- a) ambiental: preservação e conservação do meio ambiente com ações que possam reverter o desmatamento, promover o uso sustentável dos ecossistemas e a adoção de medidas que sejam refetivas e urgentes no combate à mudança climática e seus impactos;
- b) social: voltada para as necessidades humanas, com ações que possam levar à melhor qualidade de vida, educação de qualidade e medidas institucionais que possam eliminar leis discriminatórias, promover o trabalho decente e aprovação de leis mais adequadas;
- c) econômica: com a redução da produção de resíduos e consumo de energia limpa, de fontes renováveis, consideradas eficientes e não poluentes;
- d) institucional: trata da capacidade real de colocar em prática os ODS. Levando em consideração, as crises sanitária e econômica, é possível prever que o avanço na implementação dos ODS e da Agenda 2030 tenha recebido um impacto significativo, tornando o desafio, que por si só não é simples, ainda maior.

A Figura 7 mostra os 17 objetivos elencados na Agenda 2030.



FIGURA 7 - Objetivos do Desenvolvimento Sustentável

Fonte: UNRIC (2019)

Espera-se que o projeto Deu Pet consiga colaborar para o alcance do objetivo “15 - Proteger a Vida Terrestre”, que consiste em 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.

O problema que ocorre no Brasil e no mundo consiste de maus-tratos e abandono de animais, a falta de atenção quanto à inspeção populacional dos animais não domiciliados, os poucos serviços governamentais para lidar com esterilização e controle de doenças, dentre outros. Por meio do sistema de software Deu Pet, espera-se abrandar tal problema, contribuindo com a preservação da biodiversidade e dos ecossistemas terrestres.



FIGURA 8 - ODS 15

Fonte: Nações Unidas Brasil (2020)

3.7 MODELO DE NEGÓCIO PROPOSTO

O modelo de negócio, proposto para sustentação da operação do sistema de software Deu Pet, foi elaborado por meio da ferramenta “Canvas” disponibilizada pelo Sebrae. O arquivo gerado está presente no Apêndice I.

4 MÉTODOS GERENCIAIS

No final da década de 1960, o *Project Management Institute* (PMI) foi criado com o intuito de reunir profissionais de gerenciamento de projetos, que desejavam uma troca de experiências e criar boas práticas entre si. Do contato mais próximo entre os membros do PMI, surgiu o *Project Management Body of Knowledge* (PMBok), que é um guia que contém boas práticas sobre como proceder na execução de todo ciclo de vida de um projeto.

Fundamentado no PMBoK, neste capítulo é exposto o plano de elaboração e gerenciamento do projeto, modelo de ciclo de vida, recursos necessários, relatório de desempenho, estimativas de tamanho e esforço e gerenciamento de configuração.

4.1 PLANO DE ELABORAÇÃO E GERENCIAMENTO DO PROJETO

O guia PMBoK afirma que gerenciamento de projetos é a “aplicação de conhecimentos, habilidades, ferramentas e técnicas às atividades do projeto a fim de cumprir os seus requisitos” (PMI, 2013, p.5).

Existem, assim, cinco grupos de processos de gerenciamento de projeto, sendo eles nomeados iniciação, planejamento, execução, monitoramento e encerramento. A Figura 9 mostra os processos necessários para atender os requisitos de projeto segundo o PMI.

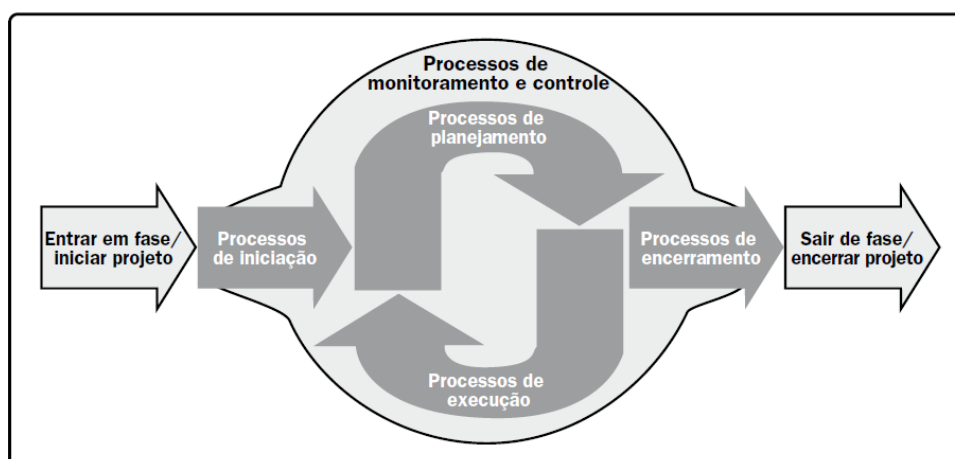


FIGURA 9 - Grupo de processos de gerenciamento de projetos.

Fonte: PMI (2013).

No Apêndice A, encontra-se o plano de elaboração e gerenciamento deste projeto, sendo elaborado com referência ao PMBok (PMI, 2013).

4.2 MODELO DE CICLO DE VIDA

Faz-se necessário criar um plano de processos contendo um conjunto de atividades e resultados os quais são associados ao que se produz em um produto de *software*. Logo, a estruturação de um conjunto de atividades define o processo de desenvolvimento. A organização previamente ao início da produção torna o projeto menos custoso e eleva o nível de qualidade do produto final (PRESSMAN; MAXIM, 2016).

Um processo pode conter várias atividades sendo as principais: especificação, projeto, implementação, validação, manutenção e evolução. Estas atividades geram as necessidades mínimas para que se possa obter um produto de software adequado ao que se almeja.

Para execução deste projeto, aplica-se o Modelo Incremental destinado a gerenciar as etapas de entregas principais.

4.2.1 Modelo Incremental

Para se trabalhar com o Modelo Incremental, o processo precisa possuir requisitos bem definidos e estruturados para que cada fase do projeto possua um acréscimo de melhorias constantes. Neste ponto, o modelo concilia elementos dos fluxos e processos tanto lineares quanto paralelos (Figura 10).

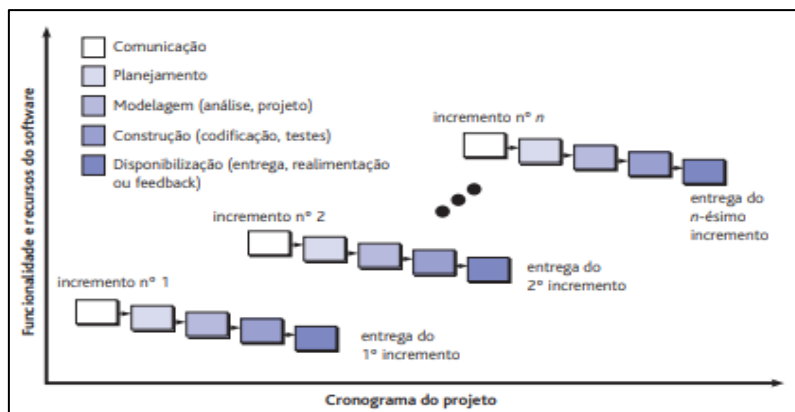


FIGURA 10 - O Modelo Incremental

Fonte: Pressman e Maxim (2016).

Para este projeto, há quatro incrementos os quais são detalhados a seguir.

O primeiro incremento tem como objetivo a elaboração e entrega parcial do primeiro artefato, com data definida para 23/04/2022. Trata-se de um documento que contempla os capítulos de introdução; revisão bibliográfica; objetivos do projeto; métodos gerenciais e suas quatro primeiras seções: plano de elaboração e gerenciamento do projeto (parcial), modelo de ciclo de vida, recursos necessários e o relatório de desempenho da Fase 1; especificação e análise dos requisitos e referências.

A entrega do segundo incremento, a ser realizado até 04/06/2022, contém a correção da documentação parcial da Fase 1; a seção de resumo; a complementação do plano de elaboração e gerenciamento do projeto, o relatório de desempenho da Fase 2, a estimativa de tamanho e esforços e a implantação do gerenciamento de configuração; arquitetura e projeto do sistema com as seções para visão estrutural, comportamental e de dados e perfil de usuário; e o capítulo de conclusão parcial; complemento das referências e, por fim, a entrega da codificação com ao menos 30% dos requisitos essenciais atendidos.

Entrega do terceiro incremento, a ser realizado até 10/09/2022, contém a correção da documentação parcial das Fases 1 e 2. A inserção de novos tópicos, como: aderência aos objetivos do desenvolvimento sustentável; o modelo de negócio proposto; o projeto da interface de usuário; as heurísticas de usabilidade; o projeto da acessibilidade; o projeto do sistema distribuído; os componentes do sistema de software; as tecnologias de implementação; as linguagens de programação e *frameworks* adotados; os *design patterns* aplicados; as convenções e guias para codificação; a estrutura física do banco de dados e, por fim, a entrega da codificação com ao menos 60% dos requisitos essenciais atendidos.

Entrega do quarto incremento, a ser realizado até 29/10/2022, contém a correção da documentação parcial das Fases 1, 2 e 3. A inserção de novos tópicos, como: o plano de testes; o plano para implantação; e 1ª. versão do vídeo demonstrativo do projeto. Trata-se do incremento da Fase 4 cujas atividades encerram este projeto.

4.3 RECURSOS NECESSÁRIOS

Para o desenvolvimento deste projeto, são necessários recursos humanos, de *software* e de *hardware*, conforme apresentados na seção a seguir.

4.3.1 Recursos Humanos

Para a realização desse projeto, são necessários três graduandos em Sistemas de Informação, sendo eles: Lucas Dias e Silva, Matheus Felipe Souza e Vinicius de Almeida Gonçalves.

É de responsabilidade de todos os membros, em conjunto: análise, projeto, confecção deste documento, programação de código-fonte e realização de testes com várias estratégias.

4.3.2 Recursos de Software

Os recursos de *software* necessários para este projeto:

- a) Sistema Operacional Windows 10 de 64 bits;
- b) IntelliJ IDEA 2021.3.3
- c) PostgreSQL 14;
- d) PgAdmin 4 v4.3;
- e) Dia 0.97.2
- f) Microsoft Excel 2013;
- g) Microsoft Word 2016;
- h) Google Drive;
- i) Visual Studio Code 1.32.3;
- j) Ideas Modeler 11.96;
- k) Postman 7.0.6;
- l) Git 2.21.0;
- m) Github;
- n) Java 11 LTS;
- o) Bootstrap 4.3.1;
- p) Sublime Text 3

- q) Discord;
- r) Google Forms;
- s) Adobe XD;
- t) Adobe Photoshop 2022 v23.3.1.426;
- u) HTML 5;
- v) JavaScript ECMAScript 2020.

4.3.3 Recursos de Hardware

Os recursos de *hardware* necessários para este projeto:

- a) 2 *laptops* com processador i5 de quarta geração (ou superior), memória RAM de 4 Gbytes (ou superior).
- b) 1 computador *desktop* com processador i5 de quarta geração (ou superior), memória RAM de 4 Gbytes (ou superior).
- c) 3 mouses ópticos;
- d) 1 teclado;
- e) 1 monitor de vídeo;
- f) 3 aparelhos celulares tipo *smartphone* com Android 8.0 (ou superior);

4.4 RELATÓRIO DE DESEMPENHO

O relatório de desempenho é um meio de dar transparência ao projeto, apresentando as medições do progresso em relação ao cumprimento das fases. Ele abrange a coleta e análise periódica do andamento real do projeto, confrontando-o com o esperado.

O relatório de desempenho do projeto Deu Pet pode ser encontrado no Apêndice B.

4.5 ESTIMATIVAS DE TAMANHO E ESFORÇO

A Técnica de Pontos de Casos de Uso (PCU) é aplicada para estimar o esforço em Homens hora (Hh) necessário para o desenvolvimento de um sistema de software orientada a objetos.

O Quadro 2 mostra a estimativa inicial realizada na Fase 2 e as reestimativas realizadas ao longo das fases seguintes.

Métrica	Fase 2 (Hh)	Fase 3 (Hh)	Fase 4 (Hh)
PCU - Karner	4234h	Mantida igual à Fase 2	Mantida igual à Fase 2
PCU - Schneider e Winters	4234h	Mantida igual à Fase 2.	Mantida igual à Fase 2.

QUADRO 2 - Estimativa de esforços

Na Fase 2, por prudência, optou-se em deixar uma certa folga nas atividades, visto que se levou em consideração a falta de experiência de alguns membros da equipe na programação para a plataforma *mobile* e também a falta de experiência com a gestão de projetos.

Na Fase 3, o ritmo o desenvolvimento foi mais constante. A documentação solicitada e a codificação da aplicação *Web* foi muito bem encaminhada. A estimativa de tamanho e esforço permaneceu análoga à Fase 2.

Na Fase 4, foram entregues os últimos capítulos solicitados, com parte do desenvolvimento *mobile* previsto para conclusão até a data de apresentação final. A estimativa de tamanho e esforço permaneceu análoga à Fase 2.

A planilha que deu origem aos resultados da estimativa de tamanho e esforço do projeto Deu Pet pode ser encontrada no Apêndice H.

4.6 GERENCIAMENTO DE CONFIGURAÇÃO

O gerenciamento de configuração é uma parte fundamental do projeto, pois ajuda a manter a consistência nas entregas produzidas, sendo possível revisar todas alterações e solicitações que ocorrem ao longo do tempo.


Neste projeto, as versões da documentação são identificadas conforme a seguinte nomenclatura: PFC_nomedoprojeto_v. Uma vez que:

- a) “PFC” é a sigla para “Projeto Final de Curso”;
- b) “nomedoprojeto” mostra o nome do projeto, neste caso “Deu Pet”;
- c) “v” indica a versão atual que o projeto se encontra.


Exemplo: PFC_Deu Pet_v1.0

Para o controle de versão da documentação e dos códigos-fonte deste projeto, é utilizado o serviço do *GitHub*, que é um repositório que usa controle de versionamento do *Git*. Ele permite que programadores ou qualquer usuário cadastrado na plataforma contribuam em projetos privados e ou *open source* de qualquer lugar do mundo. *GitHub* é amplamente utilizado por programadores para divulgação de seus trabalhos ou para que outros programadores contribuam também.

Para garantir a integridade do documento e código do projeto, o gerente do projeto Lucas Silva e Dias supervisiona e autoriza o envio de novas mudanças para o *Github*. Na Figura 11 é possível ver o uso do *Github* para controle de versionamento da documentação e código do projeto.



ViniciusNPDI doc: Adiconado documento da correcao
 ead5a91 15 minutes ago
🕒 81 commits

📁 PFC_Doc_2022	doc: Adiconado documento da correcao	15 minutes ago
📁 Projeto final de curso guia	doc: Diagrama de classes parcial adicionado	12 days ago
📄 .gitignore	Docs: Atualizado modelos de caso de uso, diagramas de interface do us...	last month
📄 Deu_Pet_Modelo_Caso_de_Uso_e_Mo...	Docs: Atualizado modelos de caso de uso, diagramas de interface do us...	last month
📄 Deu_Pet_Modelo_Caso_de_Uso_e_Mo...	Docs: Atualizado modelos de caso de uso, diagramas de interface do us...	last month
📄 Quadro entrevista help a pet.docx	doc: Adicionado o quadro entrevista help a pet	2 months ago
📄 README.md	Update README.md	3 months ago
📄 Sociedade Protetora dos Animais SR...	doc: Atualizado documentacao oficial e adicionado audio da entrevista	2 months ago
📄 Tarefas individuais Entrega 2.docx	Doc: Adicionado diagramas de pacote	12 days ago


README.md


Deu-Pet_documents

Repositório para documentação do projeto Deu Pet. Desenvolvido pelos alunos Lucas Dias, Matheus Sousa e Vinicius Almeida, do 4º ano de Sistema de Informação da FAI-MG para o PFC.


luksSD Update .gitignore
 49ea44c on 17 Feb
🕒 4 commits

📄 .gitignore	Update .gitignore	3 months ago
📄 README.md	Update README.md	3 months ago

README.md


Deu Pet

Projeto da aplicação Deu Pet, desenvolvido pelos alunos Lucas Dias, Matheus Sousa e Vinicius Almeida, do 4º ano de Sistema de Informação da FAI-MG para o PFC.

FIGURA 11 - Controle de versão do projeto usando o Github

No próximo capítulo são apresentados os requisitos a serem atendidos a fim de que os objetivos específicos deste projeto possam ser alcançados.

5 ESPECIFICAÇÃO E ANÁLISE DOS REQUISITOS DO SISTEMA

Este capítulo apresenta os requisitos funcionais, não funcionais e a análise desses por meio da construção de visões funcionais e de dados. Parte-se do escopo inicial do projeto Deu Pet para a realização da especificação e das modelagens de análise.

5.1 REQUISITOS DO SISTEMA DE SOFTWARE

A seguir são apresentados os requisitos funcionais e não funcionais do sistema de acordo com as histórias de usuários e suas necessidades. As prioridades são classificadas de acordo com as definições propostas por Sommerville (2013) em três tipos: essencial, sendo os requisitos base para o funcionamento do sistema; os requisitos importantes como requisitos necessários, porém que não afetam o funcionamento como um todo do sistema e, por fim, os desejáveis, que são os requisitos os quais são almejados para futuras implementações.

Os requisitos funcionais estão agrupados em módulos lógicos com seus respectivos requisitos detalhados a seguir.

- a) Módulo de autenticação
 - 1) Fazer cadastro de conta;
 - 2) Realizar autenticação;
 - 3) Recuperar senha;
 - 4) Alterar senha;
 - 5) Aceitar termo de uso;
- b) Módulo para as instituições
 - 6) Cadastrar instituição;
 - 7) Gerenciar Instituições;
 - 8) Listar instituições;
 - 9) Exibir detalhes da instituição;
 - 10) Editar dados da instituição;
 - 11) Excluir cadastro da instituição;
- c) Módulo para solicitação de adoção
 - 12) Cadastrar dados dos animais;

- 13) Gerenciar animais;
- 14) Listar animais;
- 15) Exibir detalhes do animal;
- 16) Editar cadastro dos animais;
- 17) Excluir cadastro dos animais;
- 18) Gerenciar solicitações de adoção de animais;
- 19) Excluir solicitações de adoção;
- 20) Demonstrar interesse em adoção;
- 21) Exibir informações para devolução de animal;
- d) Módulo para gerenciar perfil do usuário
 - 22) Exibir detalhes do perfil do usuário;
 - 23) Editar perfil;
 - 24) Alterar senha;
 - 25) Excluir perfil;
 - 26) Exibir informação para alteração de perfil da instituição;
- e) Módulo para as campanhas de bem-estar animal
 - 27) Cadastrar campanha;
 - 28) Gerenciar campanhas de bem-estar animal;
 - 29) Listar campanha de saúde;
 - 30) Exibir detalhes da campanha;
 - 31) Editar campanha;
 - 32) Excluir campanha;
 - 33) Divulgar campanha;
- f) Módulo de informação geral
 - 34) Exibir informações para denúncias;
 - 35) Exibir informações para sugestões ou críticas

a) Módulo de Autenticação

Este módulo agrupa os requisitos referentes às funcionalidades de controle de acesso ao sistema, pelo aplicativo *Web* para os administradores e os responsáveis das instituições e via o aplicativo *mobile* para os adotantes ou interessados em campanhas de bem-estar animal.

RF 01 - Fazer cadastro de usuário

Este requisito refere-se ao cadastramento dos dados do usuário interessado em adoção ou em campanhas gerais dos animais. Este requisito deverá ser atendido pelo aplicativo *mobile* e para o cadastro devem-se inserir os dados pessoais de caráter identificador como nome, e-mail, senha, telefone, logradouro, número, CEP e município. Esses precisarão ser preenchidos seguindo as diretrizes estabelecidas pelo sistema, que garantirá a não redundância dos dados e a verificação dos domínios aceitos. Cada pessoa possuirá somente uma conta no sistema com um identificador único e os dados poderão ser atualizados sempre que necessário ou excluídos quando o usuário desejar.

Prioridade: essencial.

RF 02 - Realizar autenticação

Este requisito refere-se à verificação dos dados para acesso à conta previamente criada para o usuário. A verificação se deve ao momento da autenticação, onde o usuário deverá informar o e-mail e senha da conta que deseja permissão para acesso, podendo existir a possibilidade de recuperação de conta em caso de esquecimento pelo usuário. Caso os dados de acesso não sejam encontrados pelo sistema, o usuário poderá solicitar seu cadastro. Este requisito deverá ser atendido em ambos os aplicativos, *Web* e *mobile*, e para todos os tipos de usuários presentes no sistema.

Prioridade: essencial.

RF 03 - Recuperar senha

Este requisito refere-se à possibilidade de ser efetuada a recuperação da senha de acesso à conta por um usuário previamente cadastrado no sistema. Para segurança, deverá ser informado o e-mail cadastrado para onde será enviada uma notificação através de um servidor *Simple Mail Transfer Protocol* (SMTP), com a nova senha automaticamente gerada pelo sistema, no qual deve ser trocada após realizar o novo acesso. Este requisito deverá ser atendido em ambos os aplicativos, *Web* e *mobile*, e para todos os tipos de usuários presentes no sistema.

Prioridade: essencial.

RF 04 - Alterar senha

Este requisito refere-se à possibilidade de o usuário que tem cadastro no sistema conseguir trocar a senha da conta. Para alterá-la, será preciso que se insira a senha atual e posteriormente uma nova senha, seguida de sua confirmação. Esta operação será possível mediante o acesso prévio efetuado ao sistema.

Prioridade: importante.

RF 05 - Aceitar os termos de uso

O sistema deverá fornecer aos usuários do tipo adotante/interessado e responsáveis pela instituição ao acessarem o sistema pela primeira vez, os termos de usos específicos para cada tipo de usuário. Esses termos referem-se às condições de uso do sistema e às regras gerais para utilização e os usuários só poderão utilizar das funcionalidades do sistema, caso aceitem as condições do termo.

Prioridade: essencial.

b) Módulo para as instituições protetoras dos animais

Este módulo contém as funcionalidades para cadastrar e gerenciar os perfis das instituições no sistema. Sus funções são de responsabilidade do usuário do tipo administrador e seu acesso será realizado apenas via o aplicativo *Web*.

RF 06 - Cadastrar dados da instituição

Este requisito refere-se ao cadastro das instituições por um usuário administrador. Para isso, será necessário informar dados como nome, telefone, CNPJ (ou CPF), logradouro, número, CEP e também anexar documentação dos termos de aceite assinado pela instituição a qual se deseja cadastrar. Realizado o cadastro, será gerada uma senha aleatória para acesso do usuário responsável pela instituição, a qual deverá ser trocada posteriormente (RF 24).

Prioridade: essencial.

RF 07 - Gerenciar instituições

Este requisito visa permitir o gerenciamento das instituições cadastradas no sistema. O usuário administrador terá acesso a uma lista contendo todas as instituições cadastradas (RF 08) e poderá acessar os dados completos dessas instituições (RF 09).

Prioridade: essencial.

RF 08 - Listar instituições

Este requisito visa permitir a exibição de uma lista contendo todas as instituições cadastradas no sistema. Esta lista irá exibir um resumo das informações de cada item da lista contendo nome da instituição, e-mail e telefone.

Requisito: essencial

RF 09 - Exibir detalhes da instituição

Este requisito refere-se à exibição dos dados das instituições previamente cadastradas (RF 06). Todos os usuários administradores poderão visualizar todos os dados cadastrados daquela determinada instituição. Além disso, poderá existir a edição do cadastro (RF 10) ou realizar a exclusão do mesmo (RF 11).

Prioridade: essencial.

RF 10 - Editar dados da instituição

Este requisito visa permitir a alteração do cadastro de uma instituição previamente cadastrada (RF 06), onde o administrador do sistema poderá alterar os dados de cadastro como nome, telefone, CNPJ (ou CPF), logradouro, número, CEP e município e também alterar ou inserir uma nova documentação de termos de aceite.

Prioridade: essencial.

RF 11 - Excluir os dados da instituição

Este requisito refere-se à possibilidade de exclusão de instituições previamente cadastradas (RF 06). Todos os usuários administradores poderão realizar a exclusão das instituições sempre que necessário.

Prioridade: essencial.

c) Módulo para indicação de interesse de adoção de animal

Este módulo engloba as funções relacionadas à manifestação do interesse pela adoção de animal, com tarefas relacionadas ao cadastro e gerenciamento dos animais, além da análise dos interesses em adoção são realizadas por meio do aplicativo *Web* pelo usuário responsável pela instituição e o usuário interessado em adotar ou interessado em campanhas de bem-estar animal poderá visualizar e demonstrar interesse em adotar um animal por meio do aplicativo *mobile*.

Prioridade: essencial.

RF 12 - Cadastrar dados dos animais

Este requisito refere-se ao cadastro dos animais por parte do responsável pela instituição. O usuário poderá cadastrar vários cães ou gatos, que serão automaticamente vinculados à instituição para a qual realizar-se-á o cadastro. Para fazer o cadastro dos animais, será necessário informar os dados de caráter identificador como nome, tipo, sexo, peso, porte, raça (se tiver), temperamento, pelagem primária, pelagem secundária, descrição do animal e foto. Apenas o usuário responsável pela instituição poderá efetuar o cadastro dos animais.

Prioridade: essencial.

RF 13 – Gerenciar dados dos animais

Este requisito visa o gerenciamento dos animais cadastradas no sistema (RF 12). O usuário responsável pela instituição terá acesso a uma lista contendo todas os animais cadastrados os quais pertençam a instituição (RF 14) e poderá acessar os detalhes desses animais (RF 15).

Prioridade: essencial.

RF 14 – Listar animais

Este requisito refere-se ao acesso por um usuário responsável pela instituição por meio do aplicativo *Web*, onde será exibida uma lista contendo todos os animais cadastradas no sistema os quais pertençam à instituição em questão. Esta lista irá exibir um resumo das informações de cada item da lista contendo nome do animal e sexo além de um acesso para exibição dos detalhes de cada animal (RF 14). Caso seja acessado por um usuário adotante, via o aplicativo *mobile*, serão todos animais os quais pertençam ao mesmo município desse usuário um a um,

sendo exibidos nome, foto, sexo, raça e temperamento e terá uma opção para visualizar os dados completos do animal e para exibir o próximo animal da lista.

Prioridade: essencial.

RF 15 – Exibir detalhes do animal

Este requisito visa exibir todos os dados de cadastro do animal contendo todas os dados dele. Caso seja acessado por meio do aplicativo *Web* por um usuário responsável pela instituição a qual cadastrou o animal, o mesmo terá uma opção de acesso para editar os dados desse animal (RF 16), excluir o cadastro do animal (RF 17) e outra opção para visualizar as solicitações de interesse em adoção para este animal (RF 18). Se acessado por um usuário do tipo adotante por meio do aplicativo *mobile*, o mesmo terá a possibilidade de demonstrar interesse em adotar o animal (RF 19).

Prioridade: essencial.

RF 16 – Editar dados do animal

Este requisito refere-se à edição dos dados dos animais previamente cadastrados (RF 12) que o responsável poderá editar com base na necessidade. Todos os dados poderão ser editados com exceção do número único atribuído ao animal.

Prioridade: essencial.

RF 17 – Excluir dados do animal

Este requisito refere-se à exclusão dos dados de um animal previamente cadastrado (RF 13) e só poderá ser realizado por usuários responsáveis pela instituição a qual o animal pertence e por meio do aplicativo *Web*.

Prioridade: essencial.

RF 18 – Gerenciar solicitações de adoção de animal

Requisito que visa ao usuário responsável pela instituição a fim de que seja possível exibir detalhes de cada animal (RF 15) pelo aplicativo *Web*. O responsável pela instituição poderá visualizar todas as solicitações de adoção relacionadas ao animal contendo os dados do

interessado em adotar como nome, telefone, e-mail e endereço e poderá excluir as solicitações (RF 19).

Prioridade: essencial.

RF 19 – Excluir solicitação de adoção

Este requisito refere-se a permitir a exclusão de solicitações de adoção realizados por usuários adotantes (RF 20). O acesso será permitido apenas aos usuários responsáveis pela instituição e por meio do aplicativo *Web*.

Prioridade: importante.

RF 20 – Demonstrar interesse em adoção

Este requisito refere-se à solicitação de interesse em adoção de um animal por parte dos usuários adotantes e realizada via o aplicativo *mobile*. O usuário que se interessar por um animal poderá ao acessar os detalhes do animal (RF 15), demonstrar interesse na adoção por meio da opção “Deu Pet”. Seus dados básicos como nome, telefone e e-mail serão enviados para análise do usuário responsável pela instituição.

Prioridade: essencial.

RF 21 – Exibir dados para a devolução de animal

Refere-se à possibilidade de um usuário do tipo adotante consultar os dados de um animal anteriormente adotado por ele. O usuário por meio do aplicativo *mobile* terá uma opção para visualizar os dados de contato, como nome da instituição, telefone e e-mail a qual o animal pertencia e as instruções para que possa realizar a devolução.

Prioridade: importante.

d) Módulo para gerenciamento do perfil de usuário

Este módulo contempla as funcionalidades referente ao gerenciamento de perfil de um usuário. Poderá ser acessado por meio do aplicativo *Web* pelos usuários responsáveis pelas instituições ou pelo aplicativo *mobile* pelos usuários adotantes ou interessados em campanhas

de bem-estar animal. O acesso a algumas funcionalidades será limitado a cada tipo de usuário onde um usuário adotante ou interessado poderá visualizar ou alterar dados e também excluir seu perfil. Já um usuário que seja responsável pela instituição poderá alterar apenas sua senha e visualizar instruções de como solicitar uma alteração dos dados do perfil ou excluir a conta de usuário.

RF 22 – Exibir detalhes do perfil

Este requisito refere-se à listagem de todos os dados de cadastro do usuário adotante/interessado ou da instituição controlada pelo usuário responsável. Poderá ser acessado por meio do aplicativo *mobile* pelos usuários do tipo adotante/interessados os quais terão acesso a edição de seu cadastro (RF 23), alteração de senha (RF 24) e exclusão da conta (RF 25). Os usuários responsáveis pelas instituições, via o aplicativo *Web*, terá acesso a uma sessão contendo os dados de contato com os administradores para que possam enviar uma solicitação de alteração ou exclusão de cadastro, além de poderem também alterar sua senha de acesso (RF 26).

Prioridade: essencial.

RF 23 – Editar dados do perfil

Este requisito refere-se à funcionalidade de edição dos dados do perfil. Essa opção deverá ser acessada por meio do aplicativo *mobile* pelos usuários do tipo adotante/interessado em campanhas de bem-estar animal. Contemplará a edição de todos os dados do usuário, exceto a número de identificação única e a senha a qual será disponibilizada uma opção de acesso para alteração da mesma (RF 24).

Prioridade: essencial.

RF 24 – Alterar senha da conta de usuário

Este requisito refere-se ao processo de alteração de senha de acesso para todos os tipos de usuários, podendo ser acessado por ambos os aplicativos *Web* e *mobile*. Também deverá ser disponibilizado para troca de senha em casos de recuperação de senha de acesso (RF 03). Usuários do tipo adotantes ou interessados poderão acessar a troca de senhas (RF 23) e usuários responsáveis pelas instituições poderão alterar a senha (RF 22).

Prioridade: essencial.

RF 25 – Excluir a conta de usuário

Este requisito refere-se à funcionalidade de excluir a conta do usuário previamente cadastrada no sistema. Poderá ser realizado por usuários do tipo adotantes/interessados por meio do aplicativo *mobile*. Para exclusão, deverá ser informada a senha de acesso do usuário para que seja validada a exclusão.

Prioridade: essencial.

RF 26 – Exibir dados para alteração de perfil do usuário

Este requisito refere-se à exibição dos dados dos usuários responsáveis pelas instituições. Será exibida a forma de contato com os administradores para que sejam enviadas as solicitações de alterações dos dados. Este requisito deverá ser atendido pelo aplicativo *Web*.

Prioridade: essencial.

e) Módulo das campanhas de bem-estar animal

Este módulo tem como objetivo a divulgação de campanhas de bem-estar animal, como as destinadas à vacinação contra raiva, castração e cuidados preventivos. As campanhas deverão ser cadastradas por usuários responsáveis pela instituição por meio do aplicativo *Web* e usuários interessados em campanhas poderão visualizar as informações dessas campanhas via o aplicativo *mobile*.

RF 27 – Cadastrar dados da campanha

Este requisito refere-se ao cadastro de campanhas de bem-estar dos animais. Deverá ser realizada por meio do aplicativo *Web* por usuários responsáveis pela instituição contendo os seguintes dados: título, descrição, requisitos, data de início, data de fim e poderá anexar uma imagem. Além disso, o usuário terá a possibilidade de adicionar um *link* externo para lista de espera da campanha. Será atribuído um identificador único para cada campanha que também possuirá um identificador da instituição a qual essa campanha pertencerá.

Prioridade: essencial.

RF 28 – Gerenciar dados da campanha

Este requisito refere-se a gerenciar as campanhas de bem-estar animal previamente cadastradas no sistema (RF 27). O usuário responsável pela instituição poderá acessar essa função por meio do aplicativo *Web*. O usuário terá acesso a uma lista com todas as campanhas cadastradas (RF 29) poderá visualizar os detalhes de cada campanha (RF 30) e também cadastrar uma nova (RF 27).

Prioridade: essencial.

RF 29 – Listar campanhas

Este requisito refere-se a apresentar uma listagem das campanhas cadastradas previamente por uma instituição (RF 27). Por meio do aplicativo *Web*, usuários responsáveis pelas instituições poderão visualizar suas campanhas cadastradas. Pelo aplicativo *mobile*, os usuários interessados nas campanhas poderão visualizar todas as campanhas cadastradas no município onde residem. Os dados exibidos na lista serão o título da campanha e a data de validade mais atual.

Prioridade: essencial.

RF 30 – Exibir detalhes da campanha

Este requisito refere-se à exibição dos dados da campanha previamente cadastradas no sistema (RF 27). Será acessível pelo usuário responsável pela instituição que terá acesso à edição desses dados (RF 31) e também por usuários interessados nas campanhas pelo aplicativo *mobile*, sendo que será possível visualizar a campanha e acessar o *link* do formulário para inscrição na fila para a campanha.

Prioridade: essencial.

RF 31 – Editar dados da campanha

Este requisito refere-se a realizar as alterações dos dados de uma campanha previamente cadastrada no sistema (RF 27). Poderá ser acessado apenas por usuário responsáveis pelas instituições. O usuário deverá ter a opção de confirmação das alterações, outro para cancelar e uma terceira para exclusão da conta (RF 32). Tais opções ficarão disponíveis apenas no aplicativo *Web*.

Prioridade: essencial.

RF 32 – Excluir dados da campanha

Este requisito refere-se a permitir a exclusão de campanhas previamente cadastradas no sistema (RF 27). Deverá ficar acessível por usuários responsáveis pelas instituições por meio da edição da campanha (RF 31). Para exclusão, o usuário deverá confirmar se deseja realmente realizar a exclusão.

Prioridade: essencial.

RF 33 – Divulgar campanhas

Este requisito refere-se à exibição para os usuários do tipo adotante/interessados em campanhas, toda vez que realizarem o acesso ao sistema, uma mensagem apresentada via um “cartão” contendo as campanhas mais recentes. Esse cartão será exibido logo após o acesso validado e poderá conter uma imagem, descrição e data. Deverá ser exibida também uma opção de acesso para que o usuário possa acessar os detalhes da campanha em exibição e também uma opção para não mostrar esta campanha novamente. Ambas serão acessíveis via o aplicativo *mobile*.

Prioridade: importante.

f) Módulo de informação geral

RF 34 – Exibir informações para denúncias

Este requisito visa oferecer as informações de contato com as autoridades responsáveis para que se possa realizar denúncias de maus-tratos ou abandono de animais. Os dados exibidos irão conter os contatos de serviço comum como polícia, bombeiros, prefeitura e também serão listadas as informações de contato de todas as instituições de proteção ao animal as quais pertençam ao município do usuário adotante/interessado. Ficará acessível para usuários adotantes/interessados via o aplicativo *mobile*.

Prioridade: importante.

RF 35 – Exibir informações para sugestões ou críticas

Este requisito refere-se a apresentar os dados para contato com os administrados do sistema Deu Pet, como e-mail e telefone para que possam ser enviadas sugestões ou críticas. Esses dados ficarão acessíveis para usuários adotantes/interessados via o aplicativo *mobile*.

Prioridade: importante.

5.1.2 Requisitos não Funcionais

Os requisitos não funcionais definem as propriedades do sistema, bem como suas restrições. A classificação adotada está baseada em Sommerville (2013).

a) Requisitos de Eficiência

RNF 01 - Recuperação do servidor

Em caso de falha, o sistema de software deverá ter condições de acesso em até 24h após a interrupção do acesso ao servidor que executa o sistema.

Prioridade: importante.

b) Requisitos de Segurança (confiabilidade, disponibilidade, autenticidade e integridade)

RNF 02 - Disponibilidade das informações

Este requisito refere-se ao nível de controle de acesso e autenticação, que regula que tipo de informações os usuários poderão visualizar, dada a confidencialidade de algumas informações que apenas pessoas autorizadas poderão ter acesso.

Prioridade: essencial.

RNF 03 - Regras para formação de senhas

Para cadastro de uma senha, o sistema deverá aceitar apenas aquelas que sigam a regra a seguir: possuir no mínimo 8 caracteres e no máximo 20 caracteres, com letras maiúsculas e minúsculas, número e caracteres especiais.

Prioridade: essencial.

c) Requisitos de Portabilidade

RNF 04 - Portabilidade de plataformas

Este requisito refere-se às plataformas onde o sistema estará disponível para acesso dos seus usuários e a versão mínima para utilização. As plataformas móveis Android versão 8.0 e iOS versão 15 e para navegadores *Web*, como: Google Chrome, versão 101.0.4951.54, *Mozilla Firefox*, versão 91.0 e Microsoft Edge, versão 101.0.1210.47.

Prioridade: essencial.

d) Requisitos de Usabilidade

RNF 05 - Aplicação de interfaces intuitivas, com usabilidade e acessibilidade

Este requisito refere-se a prover usabilidade para as interfaces de usuário dos aplicativos *Web* e *mobile*, criadas para interação com o usuário. As interfaces deverão ter funcionalidades bem distribuídas pelas telas de forma que promovam a usabilidade e a acessibilidade do sistema.

Prioridade: essencial.

5.1.2.2 Requisitos Organizacionais

a) Requisitos de Entrega

RNF 06 - Entregas dos incrementos

Este requisito refere-se em entregar quatro incrementos do sistema, conforme previsto no Capítulo 4. A versão beta do sistema deverá ficar pronta para apresentação na Feira de Tecnologia e Empreendedorismo da FAI (Faitec), que é realizada no último trimestre do ano.

Prioridade: essencial.

b) Requisitos de Modelagem e Codificação

RNF 07 - Linguagens de programação

O código de *back-end* será desenvolvido em Java 8, utilizando *Spring Model View Controller* (MVC), versão 4.14.1. O *front-end* do aplicativo *Web* será desenvolvido utilizando *HyperText Markup Language* (HTML 5), *Cascading Style Sheets* (CSS 3), Javascript e Thymeleaf como *template engine*. A linguagem Dart versão 2.17 será utilizada para a codificação do aplicativo *mobile*.

Prioridade: essencial.

RNF 08 - Ambiente de desenvolvimento

A aplicação deverá ser codificada utilizando as *Integrated Development Environment* (IDE) *Visual Code*, para a programação do aplicativo *mobile*, a IDE do SpringBoot para a programação do *front-end* para a *Web*, com a linguagem Java. O aplicativo *mobile* será programado com o *framework* Flutter 2.10.3.

Prioridade: essencial.

RNF 09 - Ferramentas de controle de versão

Para controle de versionamento de código deverá ser utilizado o Git. O GitHub será o repositório hospedeiro do código.

Prioridade: essencial.

RNF 10 - Ferramentas Computer-Aided Software Engineering (Case)

Para modelagem dos diagrama da UML deverá ser utilizado o Ideas Modeler, versão 11.96, e o Dia para criação do diagrama de entidade e relacionamento (DER).

Prioridade: essencial.

c) Requisitos de Padrões

RNF 11 - Padrão de documentação

O documento técnico deve seguir as normas para trabalhos científicos da ABNT e Diretrizes para Elaboração de Trabalhos Acadêmicos da FAI (2018).

Prioridade: essencial.

RNF 12 - Padrões de codificação

O padrão de código fonte seguirá as *Java Code Conventions* para a linguagem Java, definidas pela Oracle e encontradas no *website* desse desenvolvedor.

Prioridade: essencial.

RNF 13 - Padrão *Model View Controller* (MVC)

O padrão arquitetural será o MVC, contendo as camadas de controle (*controller*), de interação (*view*) e camada de manipulação de dados (*model*).

Prioridade: essencial.

5.1.2.3 Requisitos Externos

a) Requisitos de Interoperabilidade (hardware, software, comunicações)

RNF 14 – Hardware para hospedagem do servidor

Para hospedagem do sistema, o servidor deverá conter as seguintes configurações mínimas: processador Intel Core i7, de 9ª geração ou superior, 8 Gbytes de memória RAM e armazenamento de 1 Tbyte.

Prioridade: essencial.

RNF 15 – Sistema operacional do servidor

O servidor deverá ter disponível o sistema operacional Linux Mint 20.3 ou superior.

Prioridade: essencial.

RNF 16 – Sistema Gerenciador de Banco de Dados (SGBD)

O servidor deverá ter disponível o SGBD PostgreSQL, versão 14.

Prioridade: essencial.

RNF 17 – Servidor de e-mail

Para envio de mensagens de redefinição de senha deverá ser utilizado o protocolo *Simple Mail Transfer Protocol* (SMTP), a ser provido pelo Outlook.

Prioridade: essencial.

b) Requisitos Éticos

RNF 18 - Confidencialidade dos dados do usuário

Os dados particulares de cada usuário do sistema não poderão ser expostos ao público.

Prioridade: essencial.

c) Requisitos de Privacidade e Proteção dos Dados

RNF 19 - Segurança no tráfego de dados sigilosos

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

Prioridade: essencial.

RNF 20 - Privacidade e acesso aos dados

Os dados privados dos usuários deverão ser mantidos em sigilo, visando maior privacidade dentro da aplicação. Somente os administradores do SGBD poderão ter acesso e permissão para alterar o conteúdo ou a estrutura do banco de dados.

Prioridade: essencial.

RNF 21 - Autenticação e controle de acesso

Somente usuários autenticados no sistema terão acesso aos módulos privados. Deverão ser fornecidos seu e-mail e senha para autenticação e acesso ao sistema.

Prioridade: essencial.

RNF 22 – Atendimento à Lei Geral de Proteção de Dados (LGPD)

Toda política envolvendo os dados será baseada na Lei nº 13.709/2018 (BRASIL, 2018), que será seguida de forma rígida para segurança dos usuários titulares dos dados tratados pelo sistema.

Prioridade: essencial.

5.1.3 Principais Regras de Negócio

A seguir são descritas as principais regras de negócio que envolvem o contexto do sistema.

RN 01 - Os animais cadastrados estarão vinculados a uma única instituição.

RN 02 - Só será possível consultar os animais disponíveis para adoção após autenticação pelo sistema.

RN 03 - As campanhas de bem-estar animal ficarão vinculadas à instituição que a cadastrou.

RN 04 - Só será possível consultar e se inscrever nas campanhas estando com acesso autorizado no aplicativo.

RN 05 - Para realizar todas as ações pelo sistema, os usuários deverão ter conexão ativa com a Internet.

5.2 ANÁLISE DOS REQUISITOS

5.2.1 Visão Funcional

O modelo de casos de uso é constituído pelo diagrama de casos de uso e pela descrição dos fluxos de eventos.

O diagrama e descrição do projeto Deu Pet pode ser encontrado no Apêndice C.

5.2.2 Modelo Conceitual dos Dados

Esta seção descreve o modelo de dados de um sistema com alto nível de abstração no qual as relações são construídas através da associação de um ou mais atributos das entidades. Trata-se do Modelo Conceitual representado por meio do Diagrama de Entidade-Relacionamento.

O diagrama de entidade-relacionamento do projeto Deu Pet pode ser encontrado no Apêndice D.

5.2.3 Modelo Inicial da Interface de Usuário

Um modelo de interface de usuário é uma representação de como os usuários finais interagem com um programa de computador ou outro dispositivo e também como o sistema responde.

O modelo de interface de usuário do projeto Deu Pet pode ser encontrado no Apêndice E.

No capítulo a seguir, consta o detalhamento da arquitetura do sistema Deu Pet, a partir dos modelos de análise elaborados.

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, os subsistemas, 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 um sistema distribuído. Para representar as visões aplicam-se os conceitos do paradigma orientado a objetos e dos diagramas da UML.

6.1 VISÃO ESTRUTURAL

A visão estrutural do projeto tem como objetivo a representação sistêmica das estruturas das aplicações, demonstrando seu funcionamento, fluxo de ações, detalha as classes, pacotes e objetos.

Para este projeto, são detalhados os seguintes diagramas: diagrama de pacotes e diagramas de classes.

6.1.1 Diagrama de Pacotes

O diagrama de pacotes tem como função detalhar os agrupamentos lógicos do sistema e detalhar as relações de dependências entre eles. Os elementos podem ser outros pacotes, classes, objetos entre outros.

Para o projeto Deu Pet, estão detalhados os pacotes dos *frameworks* utilizados, integração com a base de dados, componentes de aplicação *back-end* e as tecnologias externas integradas ao projeto.

Ambos os aplicativos *Web* e *mobile* utilizam de *frameworks* externos e possuem os seguintes pacotes:

- a) *Views*, contendo toda a interface de usuário do sistema.
- b) *Controllers*, contendo as classes de controle responsáveis por realizar as conexões entre os pacotes de *front-end*.

- c) *Services*, com as classes responsáveis pela comunicação do *front-end* com a aplicação API Rest.

A aplicação *back-end* do sistema é representado por API Rest que contém os pacotes *Model*, *Resources* e *Data Access Object* (DAO).

6.1.2 Diagramas de Classes

Um sistema necessita de suas classes para definir os atributos e operações dos objetos e para uma boa estruturação da arquitetura em camadas. Para uma boa compreensão, um diagrama de classes demonstra quais classes de objetos estarão presentes no sistema, bem como seus atributos e operações.

O diagrama de classes do sistema Deu Pet referentes aos pacotes *model* do pacote *dp-model*, o qual contém as classes das entidades do sistema, encontra-se no Apêndice F.

6.2 VISÃO COMPORTAMENTAL

A visão comportamental do projeto exemplifica os comportamentos do sistema. As interações entre os objetos e os tipos de associações entre eles é demonstrada nesta seção.

Para este projeto, as interações entre os objetos são demonstradas: diagrama de sequência e diagrama de atividades.

6.2.1 Projeto das Interações entre Objetos

6.2.1.1 Diagramas de Sequência

Um diagrama de sequência traz a representação das interações entre os objetos e os atores, além das próprias interações dos objetos dentro do sistema. Essas representações são demonstradas por caso de uso tipificando os as ações e resultados dos casos.

Os diagramas de sequência dos casos de uso Cadastrar Animal, Cadastrar Instituição, Editar Animal, Excluir Animal e Listar Animais se encontram no Apêndice F.

6.2.2 Diagramas de Atividades

Um diagrama de atividade é essencialmente um gráfico de fluxo, mostrando o fluxo de controle de uma atividade para outra e serão empregados para fazer a modelagem de aspectos dinâmicos do sistema.

No Apêndice F desse projeto consta o diagrama de atividades do caso de uso de “Cadastro de Animais”.

6.3 VISÃO DOS DADOS

Nesta seção encontra-se o modelo operacional do banco de dados, para demonstrar a base de dados da aplicação, assim como os relacionamentos que existem entre as tabelas e também o dicionário de dados de todas as tabelas.

6.3.1 Modelo Lógico

O modelo lógico leva em conta a modelagem conceitual de dados, implementando recursos como adequação de padrão, nomenclatura, definindo as chaves primárias e estrangeiras, normalização e integridade referencial.

O modelo lógico do projeto Deu Pet pode ser encontrado no Apêndice D.

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 pelo sistema, incluindo as especificações de cada campo presente nas respectivas tabelas.

O dicionário do modelo lógico do projeto Deu Pet pode ser encontrado no Apêndice D.

6.4 PROJETO DA INTERAÇÃO HUMANO-COMPUTADOR

Segundo Pressman “segundo-se um conjunto de princípios de projeto de interfaces, o projeto identifica um layout de tela que forma a base para um protótipo de interface do usuário”.

O projeto de interação humano-computador tem como objetivo coletar as informações para a criação da interface de usuário que intermediará a comunicação entre o ser humano e computador.

Então o projeto de interfaces atribui uma medida qualitativa da facilidade e eficiência com a qual um ser humano consegue explorar as funções e recursos oferecidos pelo sistema.

Assim, com base no perfil dos usuários, o sistema deve oferecer uma interface que seja compreensível e agradável, criando uma boa experiência de uso.

6.4.1 Perfil de Usuário

Quando se observam as principais características do sistema onde os componentes do software Deu Pet estarão disponíveis para acesso, é possível analisar e identificar os perfis das pessoas que serão usuárias do sistema, conforme seguem.

- a) Interessado em campanhas de bem-estar animal
Características: são pessoas que possuem alguma familiaridade na utilização de aplicativos móveis, por utilizarem redes sociais, etc.;
- b) Interessado em adoção
Características: são pessoas que possuem alguma familiaridade na utilização de aplicativos móveis e possuem interesse em adotar animais;
- c) Responsável pela instituição
Características: são pessoas que possuem familiaridade com aplicativos *Web* e de gerenciamento e que saibam analisar resultados. O sistema Deu Pet trará uma ferramenta simples de ser utilizada, trazendo funções bem definidas para esse perfil.

No Apêndice G, constam 3 entrevistas distintas com possíveis usuários do sistema.

6.4.2 Projeto da Interface de Usuário

Uma boa interface de usuário é essencial para a usabilidade e conciliar o um bom design com suas funcionalidades é um ponto chave para a construção de sistemas interativos. Segundo Souza e Silva (2009), a interface é a primeira impressão para os usuários, por onde os mesmos se comunicarão com o *software*, apesar de ainda serem criadas interfaces que não condizem com suas funcionalidades, mesmo com o grande esforço e energia empenhados na elaboração.

Visando uma qualidade de interação entre usuário e interfaces, a estruturação de cores, fontes, plano de fundo e ícones seguirão os padrões detalhados a seguir.

6.4.2.1 Cores

As cores que são aplicadas em um projeto têm papel fundamental para direcionar os usuários para certas ações, possibilita criar um ambiente que seja amigável e leve visualmente, além de marcar a identidade visual da interface do sistema.

Seguindo os princípios da teoria das cores proposta por Goethe (1993), para que haja equilíbrio e harmonia em uma composição, deve haver uma proporção entre as cores e isso depende da luminosidade de cada uma delas. Quanto mais luminosa a cor, menos ela será utilizada na composição.

Para a logo do projeto Deu Pet são utilizadas três cores de destaque. A cor rosa clara é a principal, pois segundo Goethe (1993), representa a doçura, amor e felicidade que são os sentimentos marcantes quando se trata da relação com os animais de estimação. Para o texto do logotipo utiliza-se a cor marrom, que representa a praticidade, solidez e confiabilidade no objetivo de representar as qualidades do projeto Deu Pet. Por fim, o branco vem como cor de fundo para equilibrar o contraste entre as cores e representa segundo Goethe (1993) a honestidade, fé e benevolência, pontos estes que envolvem toda a relação de adoção de um animal.



FIGURA 12 - Logotipo para os componentes do Deu Pet.

A aplicação das cores nos textos segue o padrão adotado no logotipo, contudo, presam-se pelos níveis ideias de contraste de cada sessão. Neste ponto, duas cores básicas são utilizadas escolhidas para os textos, a cor marrom para sessões de contraste onde o plano de fundo é de cores claras e a cor branca para os textos inseridos em planos mais escuros.

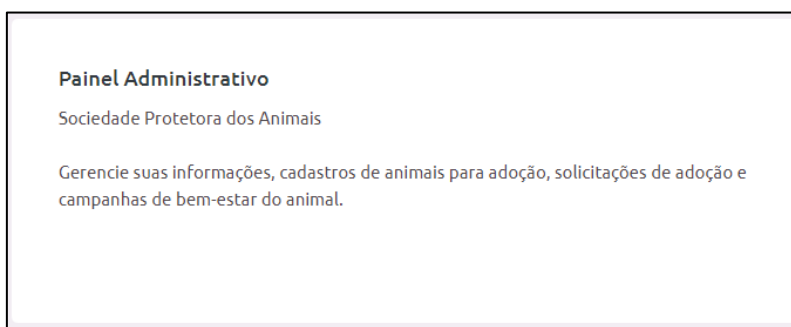


FIGURA 13 - Exemplo de cores aplicadas no texto.

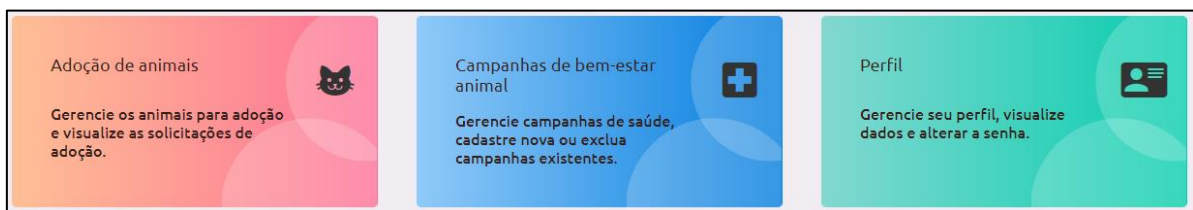


FIGURA 14 - Exemplo de textos com alto contraste.

No exemplo da Figura 14, devido à sobreposição do texto em cores fortes, uma análise de contraste foi aplicada com o *software Colour Contrast Analyser*, desenvolvido pela The Paciello Group, a fim de se obter o melhor resultado de contraste sem fugir dos padrões adotados nos textos. O bloco destinado aos dados do perfil pode ser observado na análise

demonstrada pela Figura 15. As cores definidas foram aprovadas para todas as situações, incluindo a exibição de textos como se refere a imagem na Figura 14.



FIGURA 15 - Análise do contraste entre as cores de fundo e de texto.

6.4.2.2 Fontes

A aplicação das fontes do texto visa manter um tamanho que seja minimamente adequado para leitura e com o estilo da fonte padronizado em todo o sistema. O tipo Ubuntu-Regular contém tamanhos da fonte variando para situações como títulos e textos, a fim de se obter um layout mais responsivo, atentando-se aos contrastes das cores utilizadas.

6.4.2.3 Plano de Fundo

Para o plano de fundo, é necessário escolher uma cor que possa destacar os elementos principais do sistema sem que isso desvie a atenção do usuário, sendo limpa e com cores sólidas e claras. Segundo Jackson, Macdonald e Freeman (1994), a cor cinza é a mais indicada para o fundo das interfaces, pois por ser acromática, minimiza o contraste entre a cor mais escura e a mais clara da cena, diminuindo o cansaço visual ao se passar de uma para outra.

Seguindo o conceito proposto, a interface de usuário utiliza da cor cinza como plano de fundo para proporcionar o destaque ideal dos principais elementos como exibido na Figura 16.

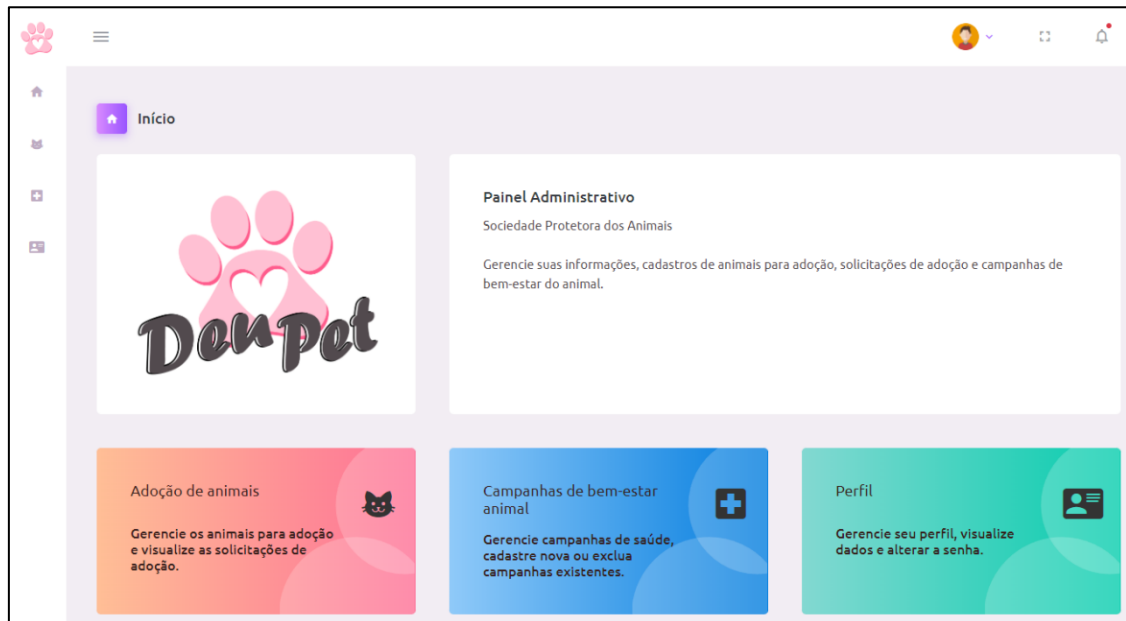


FIGURA 16 - Página inicial do aplicativo *Web*.

Seguindo os padrões de cores do aplicativo *Web*, o aplicativo *mobile* também possui cores limpas e intuitivas representando suas ações, como para o botão que indica interesse em adotar um animal. Esse botão possui um ícone com maior destaque utilizando uma cor para atrair o usuário, enquanto o botão de recusa do animal utiliza-se um tom mais neutro para evitar a atenção do usuário para esta ação como demonstrado na Figura 17.

FIGURA 17 - Tela inicial do aplicativo *mobile*

6.4.2.4 Ícones

No sistema Deu Pet, a escolha dos ícones se baseou na representatividade de suas funções, buscando sempre relacionar a imagem visual ao que de fato a função representa. Assim se faz possível a fácil interpretação e usabilidade por parte dos usuários.

FIGURA 18 - Menu principal do aplicativo *Web*



FIGURA 19 - Menu principal do aplicativo *mobile*

6.4.3 Heurísticas de Usabilidade

Projetar as interfaces de Usuário (IU) significa encontrar formas de desenvolver interações que permitam ao usuário ter uma melhor experiência. A interface do usuário não pode ser confusa, exigente ou causar estresse aos visitantes. Em vez disso, as jornadas do usuário devem ser fluidas que sua navegação se torne intuitiva e sem esforço. Para atingir esses objetivos, aplicam-se as dez heurísticas de usabilidade propostas por Nielsen e Lorenger (2007).

Segundo Nielsen e Lorenger (2007), as heurísticas são princípios gerais, o que significa que não determinam regras de usabilidade específicas, em vez disso, elas ajudam a criar produtos digitais mais acessíveis, fáceis de usar e intuitivos.

As dez heurísticas de usabilidade e suas aplicações na interface de usuário dos aplicativos são demonstradas a seguir.

6.4.3.1 Visibilidade do estado do sistema

O primeiro princípio é manter os usuários informados sobre suas ações e o que está acontecendo em uma determinada interação. Quando os usuários são informados sobre a situação real do sistema, eles aprendem os resultados de suas interações anteriores, para que possam determinar melhor quais serão os próximos passos.

Dessa forma, é importante fornecer um *feedback* instantâneo que sirva para informar a situação da interação, além de orientar e conduzir o usuário para os próximos passos.

Nos aplicativos *Web* e *Mobile* do sistema Deu Pet, um alerta de sucesso é exibido para situação de cadastros realizados de forma bem-sucedida, por exemplo, como demonstrado na Figura 20.

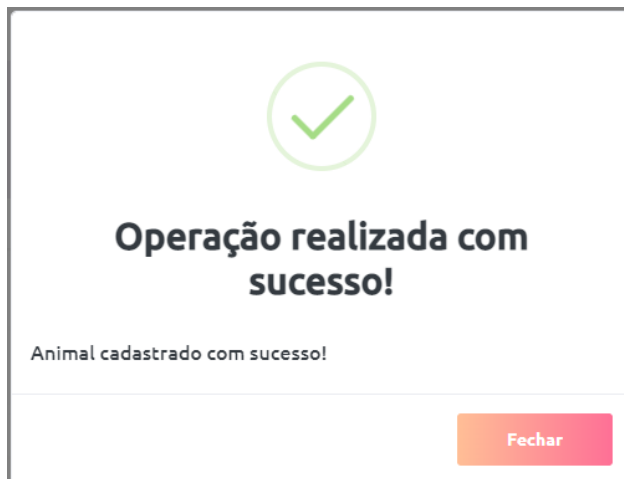


FIGURA 20 - Diálogo de confirmação de cadastro de animal.

6.4.3.2 Correspondência entre o Sistema e o Mundo Real

Este princípio afirma que um sistema deve sempre falar a língua do usuário e seguir as convenções do mundo real. Isso significa evitar jargões da área ou outras expressões que possam ser familiares para quem está construindo a interface, mas não para o usuário. Portanto, as palavras, frases e conceitos devem ser familiares ao seu público de interesse.

Além disso, para estabelecer uma conexão com o mundo real, os componentes devem aparecer em uma ordem lógica que faça sentido para os usuários de acordo com suas experiências de vida.

6.4.3.3 Controle e Liberdade do Usuário

Um bom projeto de interface do usuário nunca deve impor uma ação ao usuário ou tomar decisões por ele. Em vez disso, o sistema deve apenas sugerir quais caminhos os usuários poderão seguir. As interações criadas devem dar aos usuários a liberdade de decidir e tomar as ações que acharem adequadas, respeitando as regras e exceções do sistema.

Como exibido na Figura 17, o usuário possui total liberdade de escolha para realizar suas ações ao escolher um animal para adoção, podendo se interessar por esse animal como também o recusar.

6.4.3.4 Consistência e Padronização

Essa heurística consiste em manter a consistência em todo o sistema para evitar confundir o usuário. Assim, quando os usuários interagem com uma interface, eles devem conseguir entender sobre o significado das palavras, ícones ou símbolos utilizados.

Portanto, uma interface deve seguir as convenções do sistema, mantendo padrões de interação em diferentes contextos. Deve-se criar um design consistente que use a mesma linguagem e trate as coisas semelhantes da mesma maneira.

Como demonstrado nas Figuras 16, 17, 18 e 19, os aplicativos *Web* e *Mobile* seguem um padrão nos ícones onde cada um representa unicamente uma ação dentro de todo o sistema.

6.4.3.5 Prevenção de Erros

Nielsen (2007) propõe nesta heurística que um bom projeto de interface deve tentar evitar a ocorrência de problemas ou erros para o usuário. Em situações, por exemplo, de exclusão de um arquivo ou cadastro, deve-se supor que os usuários poderão executar tal ação acidentalmente ou que podem até imaginar um resultado diferente desse. Dessa forma, para evitar que o usuário se frustre ou cometa uma ação por engano, é fundamental criar uma mensagem de alerta para confirmar a decisão antes de prosseguir.

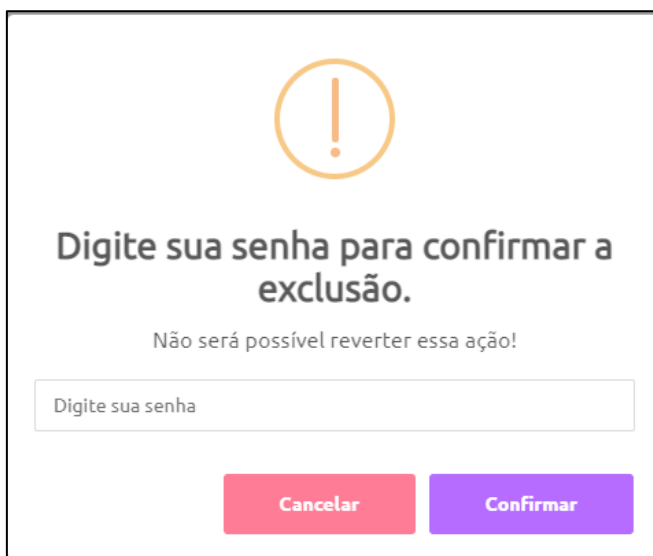
Na Figura 21, a página de autenticação apresenta uma checagem de dados de acesso, informando o usuário para tentar novamente caso encontre uma inconsistência nos dados inseridos.



The login screen for DenPet features a pink paw print logo with a heart in the center, above the text "DenPet". Below the logo, the text "Olá! Vamos começar." is followed by "Faça o acesso para continuar." There are two input fields: "E-mail" with a placeholder "Insira o e-mail" and "Senha" with a placeholder "Insira sua senha". A pink error message "E-mail ou senha inválidos. Tente novamente!" is displayed below the fields. A link "Esqueceu sua senha?" is located below the error message. A purple "Acessar" button is at the bottom.

FIGURA 21 - Tela de autenticação.

Já a Figura 22 mostra um caso de exclusão de conta, onde o usuário deverá, obrigatoriamente, informar a sua senha de acesso para que seja confirmada a exclusão.



The dialog box for account deletion features a large orange exclamation mark icon at the top. Below the icon, the text "Digite sua senha para confirmar a exclusão." is displayed, followed by "Não será possível reverter essa ação!". There is a text input field with the placeholder "Digite sua senha". At the bottom, there are two buttons: a pink "Cancelar" button and a purple "Confirmar" button.

FIGURA 22 - Diálogo de confirmação de exclusão de conta.

6.4.3.6 Reconhecimento em Vez de Memorização

Reduzir a capacidade cognitiva na utilização dos sistemas é uma das propostas de Nielsen (2007) para alcançar a usabilidade. É preciso exibir uma opção, informar ações disponíveis e assimilar imagens a funções, de tal forma que o usuário reconheça algo intuitivamente.

O usuário não deve ter que se lembrar de todas as ações ou funções do sistema. Por isso, é importante disponibilizar sempre pequenos lembretes de informações que possam auxiliá-los quanto à navegação.

Como item de ajuda ao usuário, balões com dicas e informações são exibidos para as ações mais importantes do aplicativo *Web*, como exibido na Figura 23.



FIGURA 23 - Orientação ao usuário para alterar dados.

6.4.3.7 Flexibilidade e Eficiência de Uso

Cada usuário é único e possui suas próprias necessidades e habilidades diferentes. Da mesma forma, cada tarefa é única e requer controladores diferentes. Nielsen (2007) diz que uma boa interface de usuário deve oferecer funcionalidade adequada tanto para usuários inexperientes quanto para os experientes.

Para usuários menos experientes é importante que o sistema seja organizado com facilidade na navegação e exibir apenas elementos e comandos relevantes da interface do usuário. Já os usuários com maior experiência, podem apreciar opções avançadas, atalhos ou até mesmo extensibilidade e personalização da interface do aplicativo. Os usuários avançados precisam economizar tempo e executar tarefas com rapidez, mas também com precisão e confiabilidade.

A Figura 16 demonstra que funções principais, como para o cadastro de animais para adoção, possuem blocos de acesso rápido destacados na parte principal da página para que usuários mais leigos encontrem com mais facilidade tais funções.

6.4.3.8 Projeto estético e minimalista

De acordo com esta heurística, não se deve considerar a estética acima das funcionalidades. Portanto, é necessário criar interações que contenham apenas informações essenciais e evitar elementos visuais que possam sobrecarregar a memória ou distrair os usuários.

Como exemplo, a página do aplicativo *Web* e o aplicativo *mobile* apresentados nas Figuras 16 e 17, possuem um visual limpo, minimalista, sem textos desnecessários e com informações objetivas e diretas.

6.4.3.9 Apoio ao Reconhecimento, Diagnóstico e Recuperação de Erros

Os projetos devem ajudar o usuário a identificar e encontrar soluções para eventuais problemas e erros. Para isso, as mensagens de erro devem ser exibidas em linguagem simples, sem código e clara. Além disso, informar ao usuário qual era o problema e sugerir uma solução torna o sistema mais amigável e confiável.

O aplicativo *Web* exibe páginas para os erros mais comuns, como o erro 404, para páginas não encontradas ou então erro 500 para problemas com o servidor.

6.4.3.10 Ajuda e documentação

Esta heurística de usabilidade diz respeito à documentação que ajudará os usuários a entender como realizar suas tarefas. Embora todas as heurísticas listadas devam ajudar os usuários a evitar erros e facilitar a navegação sem assistência, ainda é essencial fornecer assistência adicional a qualquer momento.

6.4.4 Projeto da Acessibilidade

Acessibilidade na *Web* é a prática de tornar os *websites* utilizáveis para todos os visitantes, incluindo aqueles com deficiências ou limitações. Para isso, é preciso atender a princípios de

acessibilidade que permitam às pessoas que experimentam dificuldades ou limitações tenham a mesma experiência ou uma experiência semelhante àquelas que não têm.

No aplicativo *Web*, existe a inclusão do atributo *alt* a fim de proporcionar textos alternativos para as principais imagens, facilitando o uso de ferramentas de transição do conteúdo para deficientes visuais. Outro atributo importante utilizado é o *title* que representa o título da página, sendo esse um dos primeiros elementos procurado pelos aplicativos leitores de textos dentro de uma página. Todos os atributos são estruturados seguindo as convenções da HTML.

Outro ponto importante de acessibilidade é a utilização de um *design* minimalista que fornece ao usuário o conteúdo necessário, sendo direto nas ações para evitar confusões ou dúvidas e com cores que levem ao melhor contraste entre texto e fundo para que a visualização se torne a mais nítida possível, como exemplo da Figura 15.

6.5 PROJETO DO SISTEMA DISTRIBUÍDO

Um sistema distribuído é aquele em que os componentes localizados em um computador estão conectados uns aos outros em uma rede, comunicam-se e coordenam suas ações apenas passando mensagens. Essa definição leva as seguintes características de sistemas distribuídos: dispositivos sendo acessados por vários usuários ao mesmo tempo, falta de relógio global e falhas de componentes individuais (COULOURIS; DOLLIMORE; KINDBERG, 2007).

6.5.1 Procedimentos para Tratamento dos Desafios

Esta seção descreve os principais requisitos necessários para um sistema distribuído apresentar interoperabilidade funcional. Esses requisitos são: heterogeneidade, escalabilidade, abertura, segurança, manuseio de falhas, concorrência e transparência.

6.5.1.1 Heterogeneidade

Com a Internet, é possível aos usuários acessarem serviços e executarem aplicativos por meio de um conjunto heterogêneo de computadores e redes (COULOURIS; DOLLIMORE; KINDBERG, 2007).

A heterogeneidade se aplica aos seguintes aspectos:

- a) redes: via protocolos de Internet é possível ser realizada a comunicação;
- b) *hardwares* de computador: diferentes padrões de placas de rede fazem que aconteça diferentes implementações. Exemplo: placa de rede Ethernet têm uma implementação diferente daqueles que possuem placas de outros padrões, isto acontece também com arquiteturas de microprocessadores (Intel, ARM, por exemplo);
- c) sistemas operacionais: diferentes sistemas operacionais trazem consigo diferentes formas de programação;
- d) linguagens de programação: cada linguagem de programação possui particularidades, seja no tratamento de vetores, registros e variáveis. É necessário tratar essas diferenças para que não haja erros de dados e aconteça uma comunicação efetiva;
- e) implementação de diferentes desenvolvedores: é necessário que diferentes desenvolvedores utilizem de padrões e convenções comuns durante a programação para que haja comunicação entre os diferentes sistemas.

O sistema Deu Pet é uma solução distribuída. O componente de *back-end* é implementado utilizando a linguagem de programação Java. O *front-end* da aplicação *Web* é constituído por meio da Thymeleaf. O *front-end* do aplicativo *mobile* é constituído por meio do *framework* Flutter.

Também é utilizada neste sistema a arquitetura 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 segura de dados entre as aplicações.

6.5.1.2 Escalabilidade

Um sistema é classificado como escalável se, perante um significativo número de recursos e usuários, ele permanece eficiente (COULOURIS; DOLLIMORE; KINDBERG, 2007).

Para possuir escalabilidade, o sistema Deu Pet utiliza boas práticas de programação, com algoritmos otimizados. Em relação ao banco de dados, existem rotinas de manutenção e criação de índices. Outra forma possível para aprimorar a escalabilidade é a utilização de serviços *Cloud*, como IaaS.

6.5.1.3 Abertura

De acordo com Tanenbaum (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. Um exemplo são os aplicativos que utilizam as APIs do Google Maps, conseguindo utilizar dados fornecidos pela API, sem precisar ter contato com os desenvolvedores da Google.

O sistema Deu Pet, mesmo se utilizando de uma documentação padronizada, convenções de código e padrões de projeto, não tem em seu escopo a abertura de sua API para terceiros.

A Figura 24 demonstra a utilização do Swagger para documentação do código.

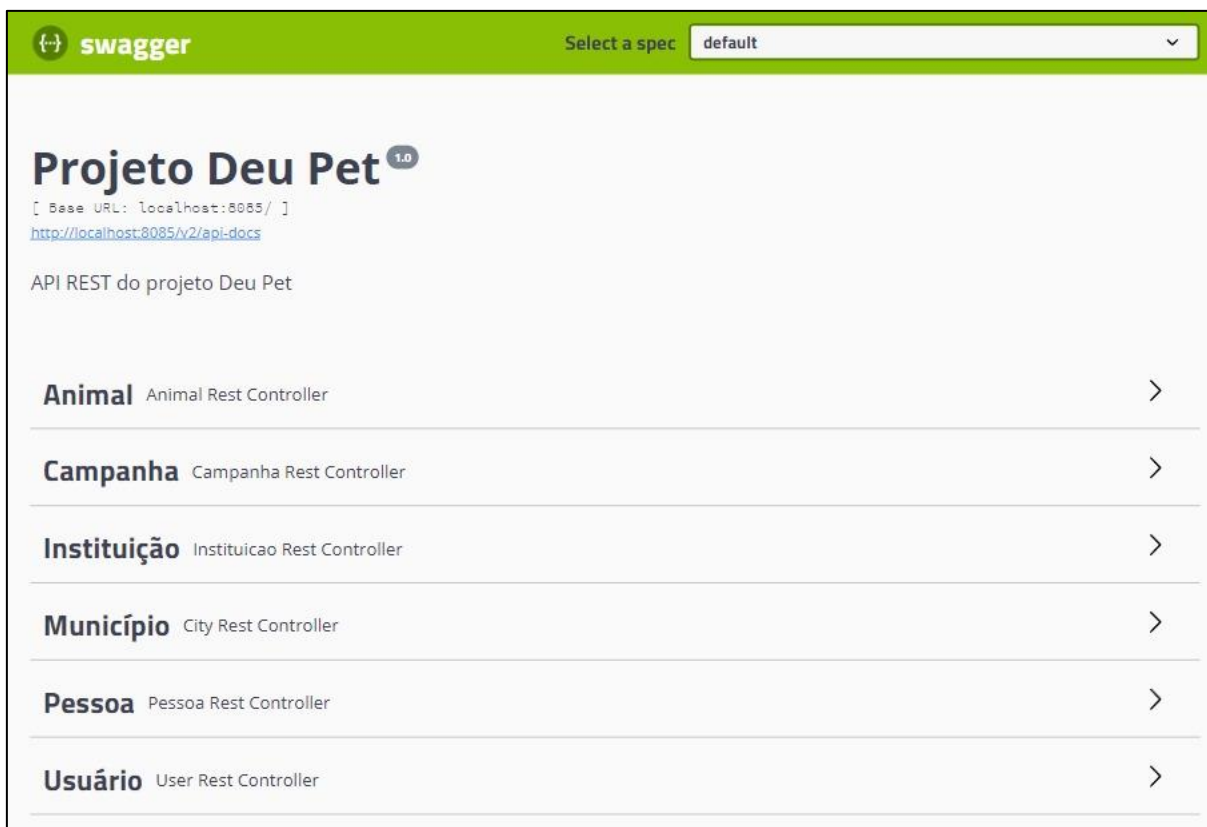


FIGURA 24 – Documentação com Swagger.

6.5.1.4 Segurança

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 fundamental importância. As seguranças 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) (COULOURIS; DOLLIMORE; KINDBERG, 2007).

Além dos três citados, existe a autenticidade, que completa a confidencialidade, visa estabelecer a validade da transmissão, da mensagem e do seu remetente. O objetivo é que o destinatário possa comprovar a origem e autoria de um determinado documento.

No sistema Deu Pet, 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 25 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.

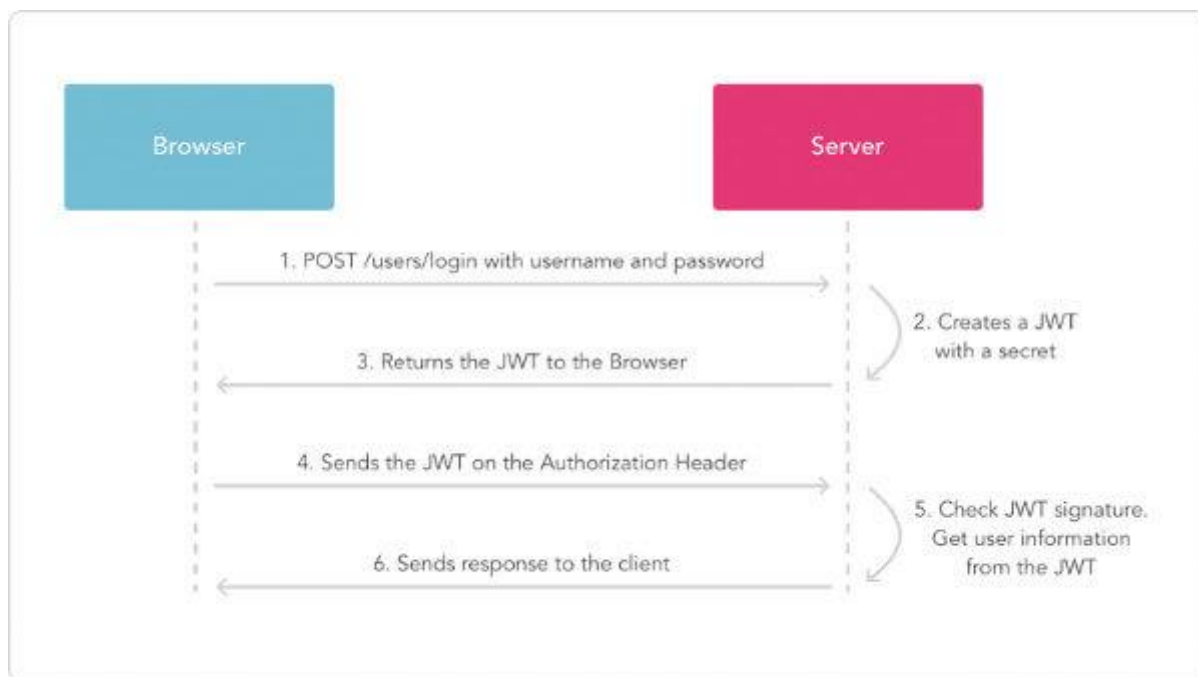


FIGURA 25 - Sequência de autenticação com JWT.

FONTE: Adriano (2017).

O sistema Deu Pet também se utiliza do *Spring Security*, que é uma estrutura de autenticação, autorização e controle de acesso poderosa e altamente personalizável. 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 *Spring Security* oferece a facilidade com que pode ser estendido para atender aos requisitos personalizados.

Para utilização do Deu Pet é necessário realizar acesso ao sistema por meio de e-mail e senha previamente cadastrados. Nenhum dado sensível como CPF do usuário é usado para realização de validações no sistema.

6.6.1.5 Manuseio de Falhas

Coulouris, Dollimore e Kindberg (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, 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.

Com base nisso, a aplicação Deu Pet ainda não tratou de forma total o desafio do manuseio de falhas. A princípio, é tratado em nível de aplicação. Em falha na API, o sistema irá retornar uma mensagem simples de falha ao usuário final para conhecimento.

6.5.1.6 Concorrência

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 (COULOURIS; DOLLIMORE; KINDBERG, 2007).

Os aplicativos do Deu Pet utilizam a mesma API, desenvolvida de modo concorrente, para que todas as requisições sejam atendidas. Os processos são tratados pelo sistema operacional, seguem a ordem de prioridade e tempo de uso dos recursos conforme as características do hardware e sistema operacional.

6.5.1.7 Transparência

O desafio da transparência tem o objetivo de tornar aspectos da distribuição invisíveis para o desenvolvedor e usuários, 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 (COULOURIS; DOLLIMORE; KINDBERG, 2007).

Um sistema com o objetivo de atender às necessidades dos clientes pode realizar diversas operações relacionadas ao banco de dados, utilizar as APIs sem a necessidade de entendimento da distribuição, suas regras técnicas e de implementação, apenas fornecendo os dados necessários. O Deu Pet utiliza métodos que podem ser executados concorrentemente e

recursos compartilhados sem a interferência entre si, realizados via funções do *Thymeleaf* e pelo SGBD *PostgreSQL*, que fornece interfaces genéricas para esses aspectos bem como implementações específicas de armazenamento no SGBD.

6.5.2 Tecnologias e Arquiteturas de Distribuição

A sistema Deu 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 e com passagens de métodos Rest.

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.

O diagrama de distribuição do sistema Deu Pet está representado na Figura 26. Ela explica como funciona o sistema Deu Pet da perspectiva de um sistema distribuído. Nela está presente os processos clientes, que são constituídos pelos processos:

- a) P1 - processo SGBD PostgreSQL: é o processo responsável por fazer o gerenciamento de acessos ao banco de dados;
- b) P2 - processo Outlook: é o processo responsável por receber e enviar e-mails;
- c) P3 - processo back-end: conecta as APIs com o banco de dados, gerencia as conexões dos usuários e alimenta os aplicativos *Web* e *mobile*.
- d) P4 - processo *front-end*: relacionado com a interface gráfica do projeto;
- e) P5 - processo do navegador cliente: fará o processamento do aplicativo *Web*;
- f) P6 - processo Deu Pet *mobile*: fará o processamento do aplicativo *mobile*.

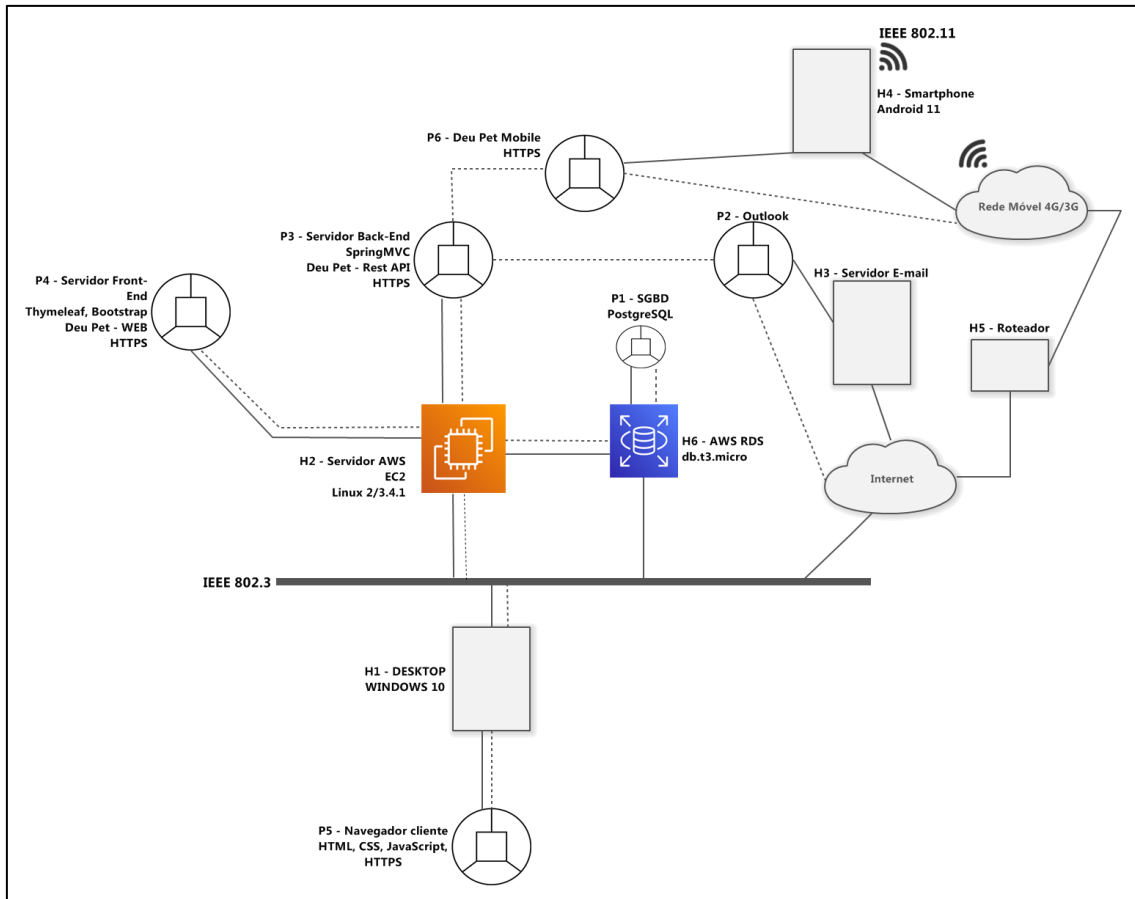


FIGURA 26 – Um possível diagrama de distribuição.

7 IMPLEMENTAÇÃO DO SISTEMA DE SOFTWARE

A implementação transforma o projeto (*design*) e suas modelagens em diversos componentes necessários para a efetiva execução e operação do sistema de software. Este capítulo apresenta a descrição técnica dessa fase de desenvolvimento do sistema de software, objeto deste projeto.

7.1 COMPONENTES DO SISTEMA DE SOFTWARE

A Figura 27 representa os componentes que compõe o sistema Deu Pet, com explicitação do esquema do aplicativo *Web*.

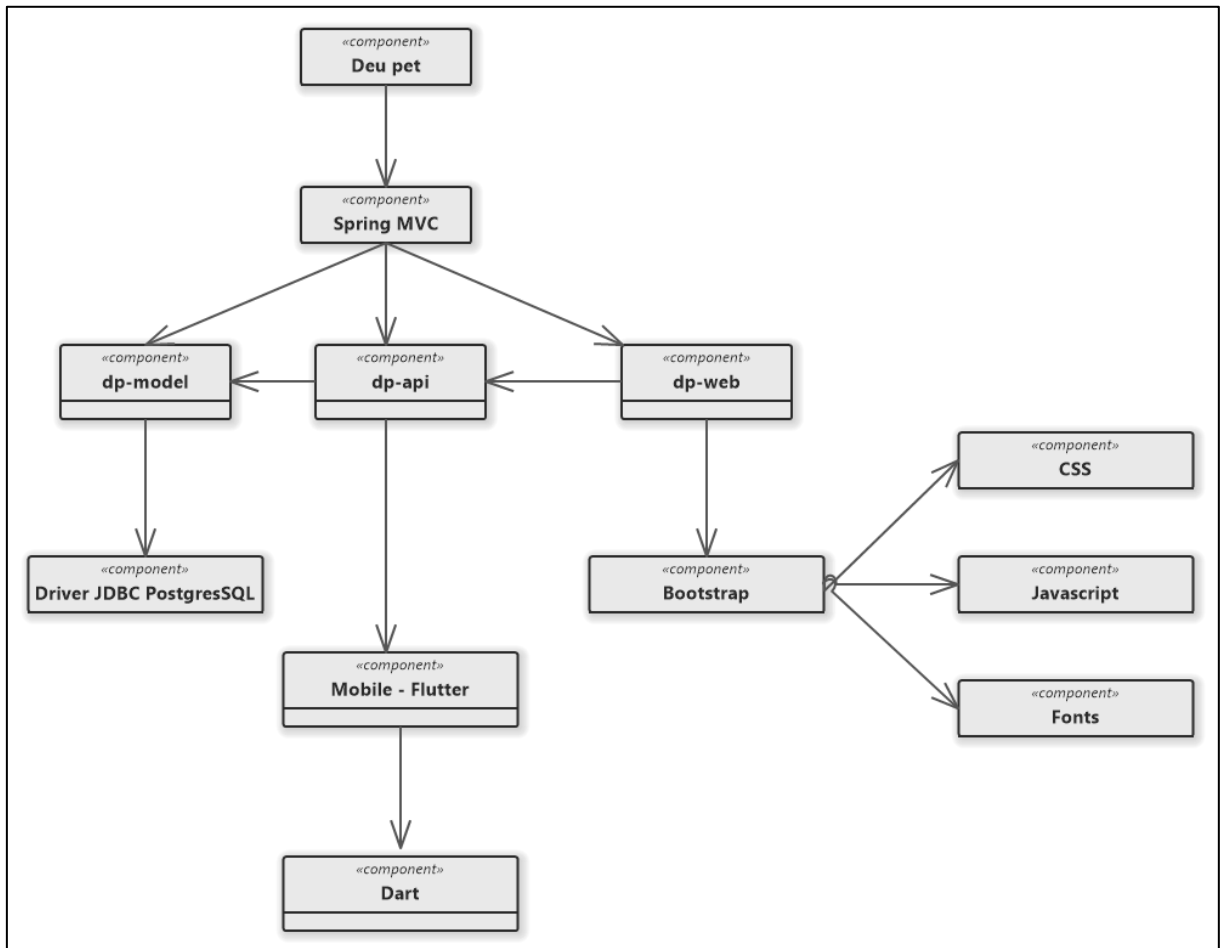


FIGURA 27 - Diagrama de componentes do aplicativo *Web*

7.2 TECNOLOGIAS DE IMPLEMENTAÇÃO

7.2.1 Linguagens de Programação e *Frameworks* Adotados

O Deu Pet utiliza o paradigma da orientação a objetos, utilizando classes e objetos para o funcionamento.

Para o aplicativo *Web*, utiliza-se a linguagem Java, com o *framework* SpringBoot, para seguir a arquitetura proposta pelo mesmo: SpringMVC. Para o *front-end* utiliza-se o Bootstrap como *framework* para desenvolvimento de componentes de interface com o usuário e JavaScript para manipulação do DOM.

Para o aplicativo *mobile*, utiliza-se o *Flutter* como *framework*, que traz a possibilidade de realizar o desenvolvimento com foco em multiplataforma em dispositivos móveis. O *framework* utiliza a linguagem de programação Dart e foi criado pela Google em 2015.

Para o banco de dados, utiliza-se o banco de dados relacional PostgreSQL. A SQL é a linguagem para criação de *scripts* de criação e manipulação desse banco.

7.2.2 *Design Patterns* Aplicados

A arquitetura sugerida pelo *framework* principal da codificação do sistema é padrão MVC que divide o código em três camadas diferentes, onde a camada *Model* é responsável pela concentração das regras de negócio da aplicação e gerência de comportamentos dos dados baseado nessas regras. A camada *View* é responsável por trazer as interfaces para que o usuário interaja. Consequentemente, a camada *Controller* é responsável por gerenciar as requisições vindas da *View*, processar e gerenciar para onde destinar essas requisições, servindo justamente como um controlador da aplicação.

7.2.3 Convenções e Guias para Codificação

Utilizar convenções e guias de código já consolidados são considerados boas práticas na programação. Pode auxiliar, por exemplo, na evolução e manutenção de um sistema. Seguir as

boas práticas ajuda 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 utilizadas na programação consistem na adoção das *Java Code Conventions*, da *Oracle*; *GitHub Conventional Commits*, que é uma metodologia com intuito de auxiliar a estruturar melhor as mensagens do *commit* e manter o histórico de mudanças organizado e legível para todos e *Code formatting* disponível nos documentos do *Flutter* compilando alguns padrões importantes.

7.2.4 Estrutura Física do Banco de Dados

A estrutura física do banco de dados 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 D deste documento.

7.3 ANÁLISE DE COMPLEXIDADE ALGORÍTMICA

Visando desenvolver algoritmos eficientes, o desenvolvedor além do conhecimento de técnicas de programação, deve ser capaz de analisar o seu comportamento com poucos ou com muitos dados de entrada. A avaliação do comportamento de um algoritmo se faz através da análise de sua complexidade. Através da análise de complexidade é possível verificar se o algoritmo está correto, avaliar sua eficiência e comparar com outros algoritmos.

Segundo Ponti (2010), a análise geralmente é baseada em uma descrição de pseudocódigo (ao invés de código fonte em determinada linguagem). Caracteriza a complexidade de tempo como uma função do tamanho da entrada, representada por “n”.

A divisão de complexidade é feita da seguinte forma:

- a) Pior caso (O);
- b) Caso Médio (Θ - Theta);
- c) Melhor caso (Ω - Ômega).

No Quadro 3 é possível verificar 3 métodos de classes distintas, informando sua respectiva complexidade.

Métodos	Complexidade	Comando de repetição
Listar todas as instituições (<i>readAll</i>)	O(N)	Figura 28
Listar todas as campanhas(<i>readAll</i>)	O(N)	Figura 29
Carregar todas as imagens de animais (<i>loadAnimalImages</i>)	O(N)	Figura 30

QUADRO 3 - Complexidade algorítmica

```
while (resultSet.next()) {

    final Instituicao instituicao = new Instituicao();

    instituicao.setId(resultSet.getLong("usuario_id"));
    instituicao.setNome(resultSet.getString("nome"));
    instituicao.setSenha(resultSet.getString("senha"));
    instituicao.setEmail(resultSet.getString("email"));
    instituicao.setCelularTelefone(resultSet.getString("celular_telefone"));
    instituicao.setSituacao(resultSet.getBoolean("situacao"));
    instituicao.setDataCadastro(resultSet.getTimestamp("data"));
    instituicao.setTipo(resultSet.getString("tipo"));
    instituicao.setCpfCnpj(resultSet.getString("cpf_cnpj"));
    instituicao.setLogradouro(resultSet.getString("logradouro"));
    instituicao.setNumero(resultSet.getString("numero"));
    instituicao.setCep(resultSet.getString("cep"));
    instituicao.setMunicipioId(resultSet.getLong("municipio_id"));

    instituicoes.add(instituicao);

}
```

FIGURA 28 - Laço de repetição para recuperar todas as instituições

```
while (resultSet.next()) {  
    final Campanha campaign = new Campanha();  
    campaign.setTitulo(resultSet.getString("titulo"));  
    campaign.setDataInicio(resultSet.getDate("data_inicio"));  
    campaign.setDataFim(resultSet.getDate("data_fim"));  
    campaign.setId(resultSet.getLong("id"));  
  
    campanha.add(campaign);  
}
```

FIGURA 29 - Laço de repetição para recuperar todas as campanhas

```
while (resultSet.next()) {  
  
    final ArquivoAnimal animalImage = new ArquivoAnimal();  
  
    animalImage.setId(resultSet.getLong("id"));  
    animalImage.setAnimalID(resultSet.getLong("animal_id"));  
    animalImage.setPath(resultSet.getString("caminho"));  
    animalImage.setPrimary(resultSet.getBoolean("primaria"));  
  
    animalsImagesList.add(animalImage);  
}
```

FIGURA 30 - Laço de repetição para recuperar todas as imagens dos animais

8 PLANO DE TESTES

Teste de Software é um processo necessário para o desenvolvimento de um software e tem como principal objetivo revelar falhas para que sejam corrigidas até que o produto final atinja a qualidade desejada e acordada.

Neste capítulo apresenta-se um plano de testes do projeto Deu Pet.

8.1 FINALIDADE

Nesta sessão são apresentadas as ações elaboradas para o plano de testes do projeto Deu Pet as quais visam verificar o correto funcionamento de partes das funcionalidades oferecidas pelo sistema.

8.2 ESCOPO

O plano de testes deste documento limita-se a 7 requisitos funcionais definidos para o aplicativo Deu Pet *Web*, demonstrando todo fluxo de execução de cada um desses requisitos, desde a criação, edição e exclusão dos dados.

8.2.1 Referências aos Documentos Relevantes,

O plano de testes do projeto Deu Pet tem como material base para elaboração a Seção 5.1 deste documento a qual referência os requisitos essenciais e não essenciais do sistema.

Tipo do material	Referência
Requisitos Funcionais e não funcionais.	Seção 5.1 do atual documento.

QUADRO 4 - Documentos relevantes para testes

8.2.2 Ambiente para a Realização dos Testes

Equipamento	Marca/modelo/configuração	Finalidade
Desktop	Processador AMD Ryzen 5 1500x, 24GB de RAM, SSD de 512GB e HD de 2TB.	Servidor <i>localhost</i> , execução e teste.

QUADRO 5 - Equipamentos para a realização dos testes

Software/versão	Fabricante	Finalidade
IntelliJ IDEA 2021.3.3	JetBrains	Ambiente de desenvolvimento integrado.
Windows 10	Microsoft	Sistema operacional.
PostgreSQL 14	The PostgreSQL Global Development Group	Sistema gerenciador de banco de dados objeto relacional.
Google Chrome 106.0.5	Google	Navegador <i>Web</i> para acessar o sistema.

QUADRO 6 - Softwares para a realização dos testes

8.3 ESPECIFICAÇÃO DOS CASOS DE TESTES

A descrição dos casos de testes e a demonstração dos resultados obtidos na realização de uma bateria constam no Apêndice J.

8.4 RESULTADOS DOS TESTES

O plano de testes elaborado e executado para o projeto Deu Pet visou testar a efetividade dos principais requisitos funcionais do sistema relacionados às operações as quais serão executadas constantemente e necessitam de uma atenção maior quanto ao seu funcionamento.

Com a conclusão dos testes realizados, foi considerado como satisfatório o funcionamento do sistema Deu Pet *Web* como também o projeto da API do sistema de acordo com o esperado ao realizar o levantamento dos requisitos.

9 PLANO PARA IMPLANTAÇÃO

Um plano de implantação visa disponibilizar o sistema de software aos seus usuários, proporcionando acesso às funcionalidades autorizadas de forma íntegra, consistente e segura.

Neste capítulo são apresentados a metodologia de implantação, a delegação das responsabilidades sob cada fase do processo, os treinamentos previstos, a documentação de apoio e o panorama de implantação em um cenário específico.

9.1 METODOLOGIA

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

9.1.1 Descrição da Metodologia

Para a realização do processo de implantação e a fim de garantir o funcionamento correto do sistema, são necessários os seguintes passos:

1. Designar os responsáveis pela implantação;
2. Definição do servidor onde serão hospedadas as APIs e o aplicativo *Web*;
3. Levantamento de requisitos de hardware e software necessários para executar as aplicações servidoras;
4. Liberação dos aplicativos.
5. Aplicação do treinamento presencial do aplicativo *Web*.
6. Gravação do treinamento para eventuais dúvidas.
7. Disponibilidade do conteúdo explicativo sobre a utilização dos aplicativos.

Ainda, para o aplicativo *mobile* é necessário apenas a instalação do APK. Eventuais dúvidas relacionadas com a instalação, serão solucionadas por meio do contato com a equipe de suporte do aplicativo pelo *Instagram* *deupet_oficial*.

9.1.2 Matriz de Responsabilidade

Atividades	Responsável
Planejamento	Matheus Felipe
Definição da equipe de implantação	Matheus Felipe
Levantamento de recursos necessários de hardware	Lucas Dias
Levantamento de recursos necessários de software	Lucas Dias
Definição de estratégias para conversão e migração de dados (caso exista sistema legado)	Vinicius Almeida
Programação dos treinamentos	Matheus Felipe
Preparação dos testes de aceitação	Vinicius Almeida
Execução	Equipe
Configuração da infraestrutura de Tecnologia da Informação	Lucas Dias
Instalação e ou configuração do ambiente	Matheus Felipe
Conversão e migração de base de dados (caso exista sistema legado)	Vinicius Almeida
Treinamentos	Equipe
Realização de testes de aceitação	Equipe
Avaliação	Equipe
Acompanhamento pós-implantação	Equipe

QUADRO 7 - Papéis e responsabilidades na implantação

9.2 TREINAMENTO PREVISTO

O treinamento tem como objetivo capacitar os usuários para utilizarem o sistema de forma plena, desfrutando de todos os módulos dispostos e tirando proveito de todos eles inteiramente.

Os treinamentos previstos no Quadro 8 capacitam o usuário a desempenhar todas as funções da aplicação de modo correto e previsto, para que possa sanar todas as necessidades ao qual o sistema foi moldado para resolver.

Treinamento	Conteúdo	Grupo de usuários
Demonstração das funcionalidades completas do sistema para os usuários responsáveis pelas instituições.	Demonstrar os módulos completos de cadastro de animais, campanhas e solicitações que contemplam: cadastro, acompanhamento ações sob os mesmos.	Usuários responsáveis pelas instituições.
Demonstração no cadastro do aplicativo para população.	Demonstrar o fácil funcionamento do aplicativo e efetividade dos campos, tal como todos os módulos de forma simples	Usuários adotantes ou interessados em campanhas de vacinação.

QUADRO 8 - Treinamentos previstos.

9.3 CRONOGRAMA DE IMPLANTAÇÃO

Nesta seção são definidos os períodos necessários para que todo processo de implantação ocorra.

Tarefas	Duração	Período
Definição de servidor onde será hospedado	8 horas	-
Levantamento de requisitos de hardware e software	8 horas	-
Implantação (<i>deploy</i>) da aplicação	30 horas	-
Aplicação de treinamento presencial do sistema <i>Web</i>	6 horas	-
Gravação do treinamento para eventuais dúvidas	6 horas	-
Disponibilidade do conteúdo explicativo	3 horas	-
Tempo estimado total:	61 horas	

QUADRO 9 - Cronograma de atividades da implantação

9.4 RECURSOS DE APOIO À IMPLANTAÇÃO

Os recursos oferecidos para apoiar o processo de implantação e posterior uso do sistema são listados no Quadro 10.

Documento	Referência
Manual do usuário	Vide Apêndice K.
Demonstração operacional do sistema	http://deupet.us-east-1.elasticbeanstalk.com/

QUADRO 10 - Recursos de apoio à implantação

9.5 VISÃO DA IMPLANTAÇÃO

No Apêndice K encontram-se o diagrama para melhor visualização das necessidades da implantação em um cenário específico. O exemplo traz a implantação do sistema em uma instancia da AWS, utilizando os serviços EC2 e RDS.

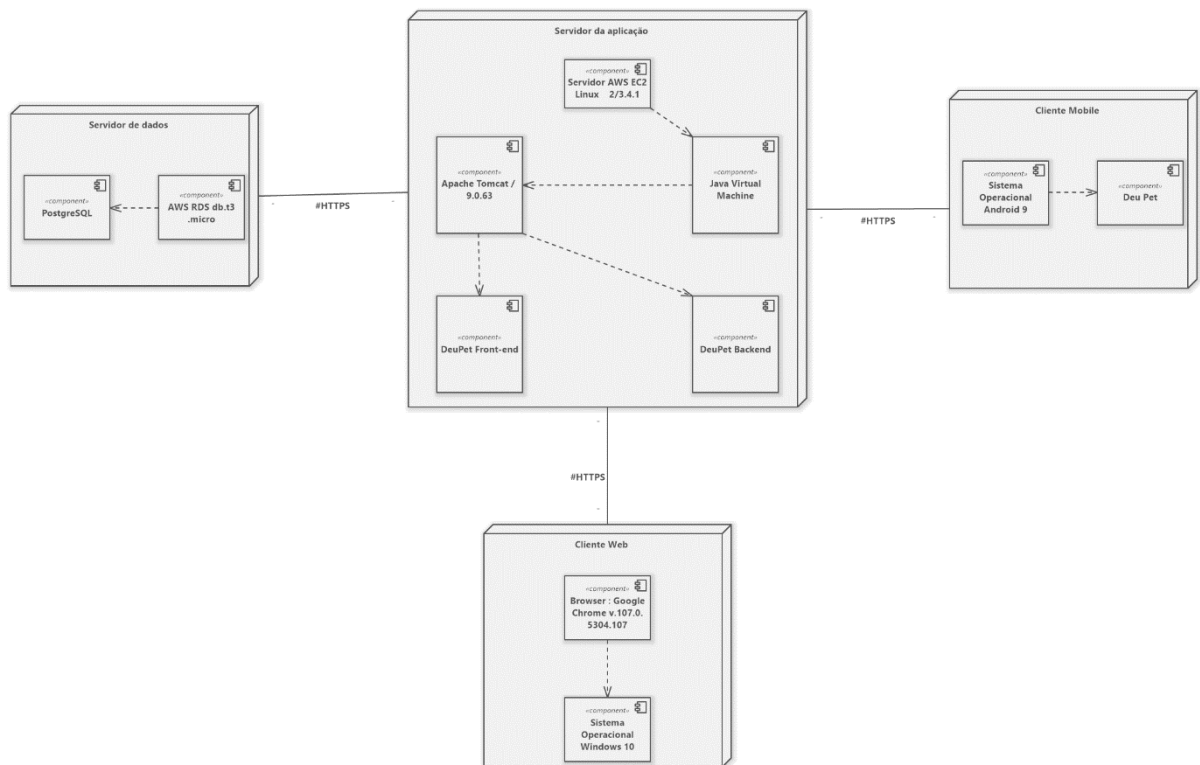


FIGURA 31- Diagrama de implantação

10 CONCLUSÃO

Os desafios encontrados e que motivaram a realização deste projeto abarcaram, dentre outros, a quantidade de animais não domiciliados abandonados e a superlotação de abrigos destinados ao resgate. Não obstante, constatou-se também a importância da adesão da população às campanhas para o bem-estar animal, bem como dos animais que estão nos abrigos para adoção, além de um modo sistemático de divulgação das informações de acordo com a necessidade de cada tipo de pessoa envolvida no processo, sejam interessados em adoção, em campanhas ou responsáveis pelas instituições.

Todas as pessoas entrevistadas durante a elicitação dos requisitos deste projeto afirmaram que utilizam as redes sociais como meio de comunicação para divulgar e receber notícias acerca de campanhas relacionadas aos *pets* e também para informar sobre animais não domiciliados e mordedores. Também confirmaram apoio à proposta de um sistema de software que fosse capaz de oferecer suporte para todas essas demandas.

Vale ressaltar a importância do trabalho das ONGs registradas e anônimas que apoiam a causa animal e todas as suas vertentes, que realizam um excelente trabalho e com certeza o sistema Deu Pet poderá beneficiar demais não só essas pessoas, como toda sociedade em si.

Baseando-se nas necessidades levantadas pela Sociedade Protetora dos Animais e na visão populacional dos temas, foram definidos os objetivos do projeto e os caminhos a serem tomados com relação as possíveis melhorias e evoluções ao longo do tempo. As pesquisas realizadas, questões absorvidas nas entrevistas e reuniões de definição do projeto Deu Pet permitiram elaborar um escopo e definir as prioridades, tratando-se de requisitos funcionais e não funcionais do sistema de software. Em seguida, foram realizadas atividades de modelagem, codificação, testes e implantação do sistema.

Em outubro de 2022, o sistema foi apresentado para as instituições e à população de Santa Rita do Sapucaí, durante a exposição da 33ª. Faitec montada no Ginásio Municipal. Neste evento, uma rápida pesquisa foi realizada pelos autores deste projeto e, cerca de 97% das pessoas entrevistadas não sabiam como funcionam o processo de adoção e 40% delas não tinham ciência do trabalho realizado pelas ONGs de proteção animal. Os 3% restantes que tinham ciência de como funciona o processo de adoção de animais, são pessoas diretamente envolvidos com a causa, sejam eles os próprios colaboradores da SPA de Santa Rita do Sapucaí e pessoas autônomas que ajudam diretamente o movimento. Esses dados foram

colhidos durante os 3 dias de apresentação do sistema na Faitec e servem de amostra para dar importância e valor ao projeto aqui realizado. Com isso, ficou evidente a defasagem da informação, que, embora seja difundida pelas redes sociais, não tem uma eficácia tanto para chegar a população, quanto impactá-la.

Baseado no cenário atual preocupante, o projeto Deu Pet foi desenvolvido, com o foco na população, que terá, nas palmas de suas mãos, um aplicativo que traz informação, facilidade e praticidade, de tal forma que a falta de informação não seja a justificativa da maioria das pessoas para não realizar a adoção dos animais.

Além disso, todo trabalho desempenhado pelas ONGs com o Deu Pet *Web* será conectado ao aplicativo Deu Pet *mobile*, onde a população irá obter acesso às informações dos animais disponíveis para adoção e às campanhas de bem-estar animal disponíveis no momento.

As Figuras 32 e 33 mostram os aplicativos Deu Pet *mobile* e Deu Pet *Web* em funcionamento.



FIGURA 32 - Deu Pet *mobile*

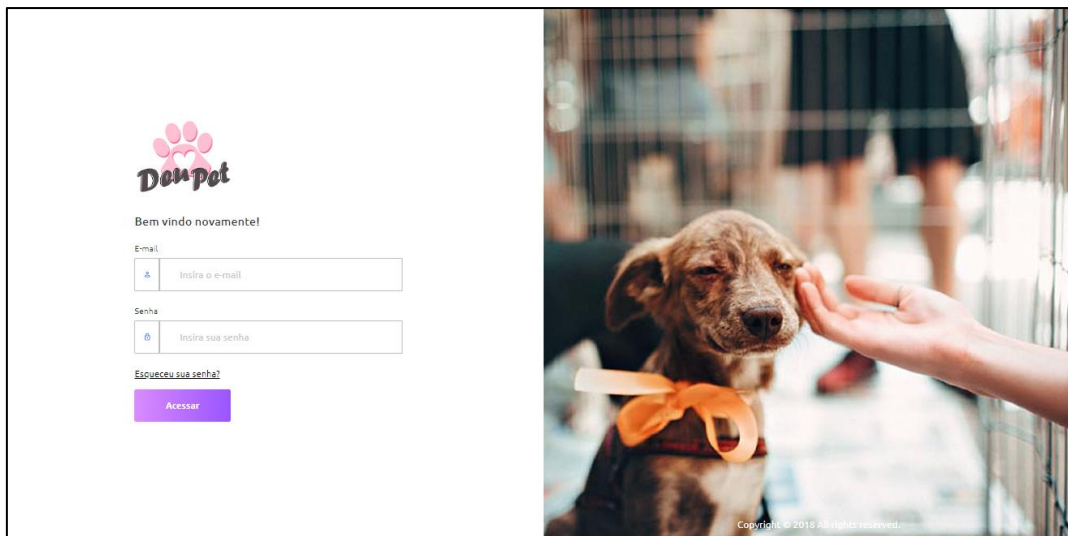


FIGURA 33- Tela de acesso Deu Pet Web

O sistema encontra-se implantado para demonstração em uma conta da AWS, disponível no seguinte URL: <http://deupet.us-east-1.elasticbeanstalk.com/>.

Cabe ainda ressaltar, que o presente projeto foi agraciado com o Prêmio Municipal de Inovação do ano de 2022, concedido pela Prefeitura Municipal de Santa Rita do Sapucaí - MG. A Figura 34 mostra a premiação dos autores.



FIGURA 34 - 1º Lugar Prêmio Municipal de Inovação 2022

Tendo esse cenário a favor, a expectativa é que o número de adoções aumente, diminuindo a superlotação dos abrigos e abrindo espaço para que novos animais sejam resgatados da situação de vulnerabilidade, dando total sentido à implantação e continuação do projeto.

REFERÊNCIAS

ADRIANO, Thiago. 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: 15 Nov. 2022.

ADOTA FÁCIL. **Adota Fácil**. Google Play, 2019. Disponível em: <https://play.google.com/store/apps/details?id=com.adotafacil.daniel.adotafacil&hl=pt_BR&gl=US>. Acesso em: 23 Mar. 2022.

ADOTA PET GO. **Adota Pet GO**. Google Play, 2019. Disponível em: <https://play.google.com/store/apps/details?id=com.labup.adotapetv2&hl=pt_BR&gl=US>. Acesso em: 23 Mar. 2022

AMBIENTAL CERNE. **ODS da ONU: 17 Objetivos de Desenvolvimento Sustentável**. 2020. Disponível em: <https://cerneambientalsc.com.br/2020/09/ods-da-onu-17-objetivos-de-desenvolvimento-sustentavel/>>. Acesso em: 1 set. 2022.

AMIGO NÃO SE COMPRA. **Amigo não se Compra**. Rio de Janeiro, 2019. Disponível em: <<https://www.amigonaosecompra.com.br/>>. Acesso em: 23 Mar. 2022

BRASIL. Lei Nº 9.605 de 12 de fevereiro de 2012. Dispõe sobre as sanções penais e administrativas derivadas de condutas e atividades lesivas ao meio ambiente, e dá outras providências. **Diário Oficial [da] União**, Brasília, DF, 17 de fevereiro de 1998. Acesso em: 16 Mar. 2022.

BRASIL. Lei Nº 14.064 de 29 de setembro de 2020. Altera a Lei nº 9.605, de 12 de fevereiro de 1998, para aumentar as penas cominadas ao crime de maus-tratos aos animais quando se tratar de cão ou gato. **Diário Oficial [da] União**, Brasília, DF, 30 de setembro de 2020. Acesso em: 16 Mar. 2022.

BRASIL. Lei nº 13.709, de 14 de agosto de 2018. Lei Geral de Proteção de Dados Pessoais. Brasília, DF, **Diário Oficial [da] União**, 15 ago. 2018. Disponível em: <http://www.planalto.gov.br/ccivil_03/_ato2015-2018/2018/lei/l13709.htm>

CONVENTIONAL COMMITS. **Conventional Commits 1.0.0.** , 2020?. Disponível em: <<https://www.conventionalcommits.org/pt-br/v1.0.0/>>. Acesso em: 16 nov. 2022.

DEEPAK, K. **Best Practices for Building RESTful Web Services**. Infosys Limited, 2015. Disponível em: <<https://www.infosys.com/digital/insights/Documents/restfulweb-services.pdf>>. Acesso em: 17 Mar. 2022.

DEITEL, P.; DEITEL, H.. **Java: como programar**. 8. ed. São Paulo: Pearson, 2010. Acesso em: 17 Mar. 2022.

DEVMEDIA. **Java Spring MVC: criando Aplicações Web em Java**. 2014. Disponível em: <<https://www.devmedia.com.br/java-spring-mvc-criando-aplicacoes-web-em-java/31521>>. Acesso em: 21 Mar. 2022.

DIAS, E. **Desmistificando REST com Java**. 1. ed. 2016. Disponível em: <<http://cafe.algaworks.com/livreto-desmistificando-rest-com-java/>>. Acesso em: 17 Mar. 2022.

ESTRATÉGIA ODS. **O que são os Objetivos de Desenvolvimento Sustentável**, 2016. Disponível em: <https://www.estrategiaods.org.br/conheca-os-ods>. Acesso em: 27 out. 2022.

GOETHE, J.W. **Doutrina das Cores**. Trad. Marco Giannotti. São Paulo: Nova Alexandria, 1993.

INSTITUTO PET BRASIL. **País tem 3,9 milhões de animais em condição de vulnerabilidade**. 2019. Disponível em: <<http://institutopetbrasil.com/imprensa/pais-tem-39-milhoes-de-animais-em-condicao-de-vulnerabilidade/#::~:~:text=De%20acordo%20com%20os%20dados,evoluem%20para%20o%20abandono%20completo>>. Acesso em: 16 Mar. 2022.

JACKSON, R.; MACDONALD, L.; FREEMAN, K. **Computer Generated Color: a practical guide to presentation and display**. New York: John Wiley & Sons, 1994.

NAÇÕES UNIDAS BRASIL (Brasil). **Os Objetivos de Desenvolvimento Sustentável no Brasil: Vida terrestre**. 2020?. Disponível em: <https://brasil.un.org/pt-br/sdgs/15>. Acesso em: 15 nov. 2022.

NIELSEN, J.; LORANGER, H. **Usabilidade na Web**: projetando websites com qualidade. Ed. Campus: Rio de Janeiro, 2007.

KUWANO, J. P. **Desenvolvimento De Um Website De Acompanhamento E Adoção De Animais De Rua**. 2018. Trabalho de conclusão de curso (Tecnólogo em Análise e Desenvolvimento de Sistemas) - Universidade Tecnológica Federal Do Paraná, [S. l.], 2020. Disponível em: <http://repositorio.utfpr.edu.br:8080/jspui/bitstream/1/16823/1/PG_COADS_2018_2_03.pdf> . Acesso em: 21 Mar. 2022.

LOPES, A.S.; PEREIRA, D.F.; MENDES, T.S. **Help a Pet- Sistema de Software para Auxílio à Causa da Proteção Animal**. Santa Rita do Sapucaí: FAI – Centro de Ensino Superior em Gestão, Tecnologia e Educação. 2019. Acesso em: 17 Mar. de 2022.

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. **Ciênc. Anim. Bras.**, n. 16, p. 574-588, 2015. Acesso em: 17 Mar. 2022.

MOUTINHO, F. F. B.; SERRA, C. M. B.; VALENTE, L. C. M.. Situação Pós-adoção dos animais adotados junto a uma ONG de Proteção Animal no Estado do Rio de Janeiro. **Ciência Animal Brasileira**, [S.l.], p. 8-12, 2019. Disponível em: <<https://www.scielo.br/j/cab/a/nFpjLbdSnz4TSPSbDPy6b/?format=pdf&lang=pt>>. Acesso em: 21 Mar. 2022

MOZILLA. **Web Components**. 2018. Disponível em: <https://developer.mozilla.org/en-US/docs/Web/Web_Components>. Acesso em: 17 Mar. 2022.

ORACLE. **Java Code Conventions**. 1997. Disponível em: <<https://www.oracle.com/technetwork/java/codeconventions-150003.pdf>>. Acesso em: 16 nov. 2022.

PATRONEK, G. J.; GLICKMAN, L.; BECK, A.; MCCABE, G.P. ; ECKER, C. Risk Factors For Relinquishment of Dogs to an Animal Shelter. **Journal of the American Veterinary Medical Association**. 1996. 209. 572-81Acesso em: 17 Mar. 2022.

PISCO DE LUZ. **Desenvolvimento Sustentável**. 2022. Disponível em: <<https://www.piscodeluz.org/desenvolvimentosustentavel?gclid=CjwKCAjwsMGYBhAEEiw>>

AGUXJaRv2Ba55jxVGOck2DraeiteIv0ZukKvCp7A9h42hgbEBQsZSmxtpghoCuaAQAvD_BwE>. Acesso em: 01 Set.. 2022

PONTI, Moacir J. **Análise de Algoritmos (parte 3)**, 2010. Disponível em: <http://wiki.icmc.usp.br/images/3/38/ICC2_03.AnalisedeAlgoritmos_parte3.pdf>. Acesso em: 16 nov. 2022.

PRESSMAN, R.S.P.; MAXIM, B.R.M. **Engenharia de Software**. Tradução de João Eduardo Nóbrega Tortello. Porto Alegre: AMGH Editora, 2016. Acesso em: 17 Mar. 2022.

PROJECT MANAGEMENT INSTITUTE. **Um Guia do Conhecimento em Gerenciamento de Projetos. Guia PMBOK**. 6. ed. - EUA: Project Management Institute, 2017. Acesso em: 17 Mar. 2022.

RIBEIRO, M. F.; FRANCISCO, R. E. Web Services Rest Conceitos, Análise e Implementação. **Revista E.T.C. Educação, Tecnologia e Cultura**, Bahia, 2016, v. 1, n. 14, p. 4-7, 30 jun. 2016. Disponível em: <<http://www.publicacoes.ifba.edu.br/index.php/etc/article/view/25>>. Acesso em: 21 Mar. 2022.

SALLES, C. **Saiba quais Atitudes Podem Ser Consideradas Maus Tratos aos Animais**. Jusbrasil, 2015. Disponível em: <<https://carollinasalle.jusbrasil.com.br/noticias/155756645/saiba-quais-atitudes-podem-ser-consideradas-maus-tratos-aos-animais>>. Acesso em: 17 Mar. 2022.

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: 17 Mar. 2022.

SANTANA, V. Cachorro Fica Preso mais de uma Hora Dentro de Carro Estacionado em Rua de Goiânia. **G1**, 06 de Jan. de 2020. Disponível em: <<https://g1.globo.com/go/goias/noticia/2020/02/06/cachorro-fica-preso-mais-de-uma-hora-dentro-de-carro-estacionado-em-rua-de-goiania-video.ghtml>>. Acesso em: 17 Mar. 2022.

SILVA, D. R. *et al.* Levantamento das Notícias sobre Maus Tratos a Animais em um Site Público de Notícias na Região De Patos De Minas – MG, no Período de 2016 a Outubro de 2018. **Psicologia e Saúde em Debate**, 2018, 4 (Suppl1), 79–79. Disponível em: <<http://psicodebate.dpgpsifpm.com.br/index.php/periodico/article/view/425>>. Acesso em: 17 Mar. 2022.

SOMMERVILLE, I. **Engenharia de Software**. 9. ed. São Paulo: Pearson Education do Brasil, 2013

SOUZA, A. C. A.; REIS, S. T. J. Origem e histórico dos animais resgatados e tutelados por ONGs de proteção aos animais, protetores independentes. **Revista Científica de Medicina Veterinária**. Associação de protetores de animais da cidade de Salvador/BA., [S. l.], v. 2, n. 48, p. 5-9, 2018. Disponível em: <<https://medvep.com.br/wp-content/uploads/2020/06/Origem-e-hist%C3%B3rico-dos-animais-resgatados-e-tutelados-por-ONGs-de-prote%C3%A7%C3%A3o-aos-animais-protetores-independentes-e-Associa%C3%A7%C3%A3o-de-protetores-de-animais-da-cidade-de-Salvador-BA.pdf>>. Acesso em: 21 Mar. 2022.

SOUZA, R. F. **Entrevista Sociedade Protetora Dos Animais**. 15 mar. 2022. Acesso em: 26 Mar. 2022.

SOUZA, R.; SILVA, R. A interface com o usuário no projeto de software. **Nucleus**, 6, 2009.

SPRING.IO. **Why Spring?** 2022. Disponível em: <<https://spring.io/why-spring>>. Acesso em: 21 Mar. 2022.

TANENBAUM, A. S. **Distributed Systems: Principles and Paradigms**. 3. ed. Vrije Universiteit: [s. n.], 2017.

TRINDADE, L. S. *et al.* O Abandono Irresponsável De Animais Domésticos. **Anais do Salão Internacional de Ensino, Pesquisa e Extensão**, v. 2, n. 1, 2 fev. 2013. Acesso em: 16 Mar. 2022.

URIC. **Objetivos de Desenvolvimento Sustentável**. Disponível em: <<https://unric.org/pt/Objetivos-de-Desenvolvimento-Sustentavel/>>. Acesso em: 1 set. 2022.

VELASCO, E. R. **Web Authorization and Authentication for Single Page Applications (SPAs)**. 2018. 98 p. Tese (Licenciatura em Engenharia de Telemática) - Escola Técnica d'Enginyeria de Telecomunicació de Barcelona, Barcelona, 2018. Disponível em: <<https://upcommons.upc.edu/bitstream/handle/2117/117772/Memoria-Enric-Ruhi-Velasco.pdf?sequence=1&isAllowed=>>>. Acesso em: 21 Mar. 2022.

WALSH, F. Human-Animal Bonds I- The Relational Significance of Companion Animals. **Family Process**, 48, 463-480, 2009. Acesso em: 17 Mar. 2022.

OBRAS CONSULTADAS

DEV MEDIA. **Introdução aos Processos de Software e o Modelo Incremental e Evolucionário**. 2013. Disponível em: <<https://www.devmedia.com.br/introducao-aos-processos-de-software-e-o-modelo-incremental-e-evolucionario/29839>>. Acesso em: 23 mar. 2022.

CAVALCANTI, Alberes Veloso Alves. **Animais Abandonados: uma perspectiva de ONGS** quanto ao problema público da proteção animal no município de João Pessoa/PB. 2021. Trabalho de conclusão de curso (Bacharelado em Administração) - Instituto Federal de Educação, Ciência e Tecnologia da Paraíba (IFPB), 2021. Disponível em: <<https://repositorio.ifpb.edu.br/bitstream/177683/1484/1/Alberes%20Veloso%20Alves%20Cavalcanti%20-%20Animais%20abandonados%20uma%20perspectiva%20das%20ONGs.pdf>>. Acesso em: 21 Mar. 2022.

CIÊNCIA ANIMAL BRASILEIRA. **Uma ONG de Proteção Animal no Estado do Rio de Janeiro**, [S. l.], p. 8-12, 2019. Disponível em: <<https://www.scielo.br/j/cab/a/nFpjLbdSnzvz4TSPSbDPy6b/?format=pdf&lang=pt>>. Acesso em: 21 Mar. 2022.

GARCIA, M. F. **Frio Cruel: 30 milhões de animais vivem nas ruas do Brasil**. Observatório do terceiro setor. 24 de agosto de 2020. Disponível em: <<https://observatorio3setor.org.br/noticias/frio-cruel-30-milhoes-de-animais-vivem-nas-ruas-dobrasil/#:~:text=A%20Organiza%C3%A7%C3%A3o%20Mundial%20da%20Sa%C3%BAde,s%C3%A3o%204%20milh%C3%B5es%20de%20animais.>>. Acesso em: 16 Mar. 2022.

MAJOLO, S.; CHIELA, E.; LUSA, T. **ONGs e a Importância de suas Ações no Meio Social: auperpopulação, Abandono e a luta pelo bem-estar animal**. Disponível em: <https://eventos.uceff.edu.br/eventosfai_dados/artigos/cibea2018/855.pdf>. Acesso em: 21 Mar. 2022.

NATOLI, E. Urban feral cats (*Felis catus* L.): perspectives for a demographic control respecting the psycho-biological welfare of the species. **Annali dell'Istituto Superiore di Sanità**, 1994, 30 (2), 223-227. Acesso em: 17 Mar. 2022.

APÊNDICE A – PLANO DE ELABORAÇÃO E GERENCIAMENTO DO PROJETO

O plano de elaboração e gerenciamento do projeto está disponível na pasta “ApêndiceA” que acompanha este documento.

APÊNDICE B – RELATÓRIO DE DESEMPENHO

Os relatórios de desempenho estão disponíveis na pasta “ApêndiceB” que acompanha este documento.

APÊNDICE C – VISÃO FUNCIONAL

O modelo de casos de uso está disponível na pasta “ApêndiceC” que acompanha este documento.

APÊNDICE D – VISÃO DOS DADOS

O modelo conceitual dos dados (representado por meio do modelo entidade-relacionamento), o modelo lógico dos dados e o dicionário dos dados do modelo lógico estão disponíveis na pasta “ApêndiceD” que acompanha este documento.

APÊNDICE E – MODELO INICIAL DA INTERFACE DE USUÁRIO

Os modelos de interface de usuário de baixa e média fidelidades estão disponíveis na pasta “ApêndiceE” que acompanha este documento.

APÊNDICE F – VISÕES ESTRUTURAL E COMPORTAMENTAL

Os diagramas elaborados por meio da UML estão disponíveis na pasta “ApêndiceF” que acompanha este documento.

APÊNDICE G – ENTREVISTAS COM USUÁRIOS

As entrevistas com os potenciais usuários do sistema encontram-se disponíveis na pasta “ApêndiceG” que acompanha este documento.

APÊNDICE H – ESTIMATIVA DE TAMANHO E ESFORÇO

As estimativas de tamanho e esforço de desenvolvimento do sistema, realizadas por meio da técnica de Pontos de Casos de Uso, estão disponíveis na pasta “ApêndiceH” que acompanha este documento.

APÊNDICE I – MODELO DE NEGÓCIO

O modelo de negócio em Canvas está disponível na pasta “ApêndiceI” que acompanha este documento.

APÊNDICE J – CASOS DE TESTES

Os casos de testes e histórico de realização estão disponíveis na pasta “ApêndiceJ” que acompanha este documento.

APÊNDICE K – RECURSOS DE APOIO À IMPLANTAÇÃO

Os documentos de apoio ao usuário estão disponíveis na pasta “ApêndiceK” que acompanha este documento.

APÊNDICE L – ENTREVISTA COM A SOCIEDADE PROTETORA DOS ANIMAIS

A entrevista realizada ao presidente Rafael Ferrari de Souza da Sociedade Protetora dos Animais de Santa Rita do Sapucaí no dia 15 de março de 2022 em Santa Rita do Sapucaí – MG, está disponível na pasta “ApêndiceL” que acompanha este documento.