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

|  |  |  |  |
| --- | --- | --- | --- |
| Data | Versão | Autor (es) | Descrição |
| 26/03/2022 | 1.0 | Lucas Dias Matheus Felipe Vinicius Almeida | Entrega da Fase 1. |

AGRADECIMENTOS

[A inclusão desta seção é opcional.]

RESUMO

[ O resumo deve ser escrito em apenas um parágrafo. Não ultrapasse 250 palavras. Não utilize siglas. No final, apresente de 3 a 6 palavras-chave, separadas entre si por pontos. O espaçamento é simples. ]

**Palavras-chave:** Palavra-chave1. Palavra-chave2. Palavra-chave3.

LISTA DE FIGURAS

[Figura 1: Foto do animal preso dentro do carro por 1 hora 16](#_Toc98864526)

[Figura 2: Trabalhando com Cache 20](#_Toc98864527)

[Figura 3: Sistema em camadas. 22](#_Toc98864528)

[Figura 4: Grupo de processos de gerenciamento de projetos. 27](#_Toc98864529)

[Figura 5 - O Modelo Incremental 29](#_Toc98864530)

[Figura 6 - Processo SCRUM 30](#_Toc98864531)

LISTA DE QUADROS

[QUADRO 1 – Estimativa de esforços 18](#_Toc65846274)

LISTA DE ABREVIATURAS E SIGLAS

ABNT Associação Brasileira de Normas Técnicas

API *Application Programming Interface*

ART Artigo

CASE *Computer-Aided Software Engineerring*

CPF Cadastro de Pessoa Física

DER Diagrama Entidade e Relacionamento

FAI FAI - Centro de Ensino Superior em Gestão, Tecnologia e Educação

GO Goiânia

HTML *Hyper Text Markup Language*

HTTP *HyperText Transfer Protocol*

IDE *Integrated Development Environment*

MER Modelo Entidade Relacionamento

MVC *Model-View-Controller*

MVP *Mininum Viable Product*

Nº Numero

ONG Organizações não governamentais

OSCIP Organização da Sociedade Civil de Interesse Público

PCU Pontos de Casos de Uso

PDF *Portable Document Format*

PMBoK *Project Management Body of Knowledge*

PMI *Project Management Institute*

RAM *Random Access Memory* (Memória de Acesso Aleatório)

REST *Representation State Transfer*

RF Requisito Funcional

RNF Requisito Não Funcional

SGBD Sistema Gerenciador de Banco de Dados

SPA Sociedade Protetora dos Animais

UML *Unified Modeling Language*

URL *Uniform Resource Location*

SUMÁRIO

[1 INTRODUÇÃO 14](#_Toc98865234)

[2 REVISÃO BIBLIOGRÁFICA 15](#_Toc98865235)

[2.1 O abandono de animais 15](#_Toc98865236)

[2.2 ONGS 17](#_Toc98865237)

[2.3 Arquitetura REST 19](#_Toc98865238)

[2.3.1 Arquitetura Cliente-Servidor 20](#_Toc98865239)

[2.3.2 Stateless (Sem Estado) 20](#_Toc98865240)

[2.3.3 Cache 21](#_Toc98865241)

[2.3.4 Sistema em camadas 22](#_Toc98865242)

[2.4 WEB COMPONENTS 23](#_Toc98865243)

[2.5 SPRING MVC 23](#_Toc98865244)

[3 OBJETIVO DO PROJETO 25](#_Toc98865245)

[3.1 FORMULAÇÃO DO PROBLEMA 25](#_Toc98865246)

[3.2 OBJETIVOS 25](#_Toc98865247)

[3.3 JUSTIFICATIVA 26](#_Toc98865248)

[3.4 PÚBLICO DE INTERESSE 26](#_Toc98865249)

[3.5 NÍVEIS DE DECISÃO E GRUPOS FUNCIONAIS ATENDIDOS 26](#_Toc98865250)

[4 MÉTODOS GERENCIAIS 28](#_Toc98865251)

[4.1 PLANO DE ELABORAÇÃO E GERENCIAMENTO DO PROJETO 29](#_Toc98865252)

[4.2 MODELO DE CICLO DE VIDA 29](#_Toc98865253)

[4.2.1 Modelo Incremental 29](#_Toc98865254)

[4.2.2 Modelo Ágil: SCRUM 31](#_Toc98865255)

[4.3 RECURSOS NECESSÁRIOS 32](#_Toc98865256)

[4.3.1 Recursos Humanos 32](#_Toc98865257)

[4.3.2 Recursos de Software 32](#_Toc98865258)

[4.3.3 Recursos de Hardware 33](#_Toc98865259)

[4.4 RELATÓRIO DE DESEMPENHO 33](#_Toc98865260)

[4.5 ESTIMATIVAS DE TAMANHO E ESFORÇO 33](#_Toc98865261)

[4.6 GERENCIAMENTO DE CONFIGURAÇÃO 34](#_Toc98865262)

[5 ESPECIFICAÇÃO E ANÁLISE DOS REQUISITOS 35](#_Toc98865263)

[5.1 REQUISITOS DO SISTEMA DE SOFTWARE 35](#_Toc98865264)

[5.1.1 Requisitos Funcionais 35](#_Toc98865265)

[5.1.2 Requisitos não Funcionais 35](#_Toc98865266)

[5.1.2.1 Requisitos de Produto 35](#_Toc98865267)

[5.1.2.2 Requisitos Organizacionais 36](#_Toc98865268)

[5.1.2.3 Requisitos Externos 36](#_Toc98865269)

[5.1.3 Principais Regras de Negócio 36](#_Toc98865270)

[5.2 ANÁLISE DOS REQUISITOS 37](#_Toc98865271)

[5.2.1 Visão Funcional 37](#_Toc98865272)

[5.2.2 Modelo Conceitual dos Dados 37](#_Toc98865273)

[5.2.3 Modelo Inicial da Interface de Usuário 37](#_Toc98865274)

[6 ARQUITETURA E PROJETO DO SISTEMA DE SOFTWARE 38](#_Toc98865275)

[6.1 VISÃO ESTRUTURAL 38](#_Toc98865276)

[6.1.1 Diagrama de Pacotes 38](#_Toc98865277)

[6.1.2 Diagramas de Classes 38](#_Toc98865278)

[6.1.3 Diagramas de Objetos 38](#_Toc98865279)

[6.2 VISÃO COMPORTAMENTAL 38](#_Toc98865280)

[6.2.1 Projeto das Interações entre Objetos 39](#_Toc98865281)

[6.2.2 Diagramas de Atividades 39](#_Toc98865282)

[6.3 VISÃO DOS DADOS 39](#_Toc98865283)

[6.3.1 Modelo Lógico 39](#_Toc98865284)

[6.3.2 Dicionário de Dados do Modelo Lógico 40](#_Toc98865285)

[6.4 PROJETO DA INTERAÇÃO HUMANO-COMPUTADOR 40](#_Toc98865286)

[6.4.1 Perfil de Usuário 40](#_Toc98865287)

[6 CONCLUSÃO 41](#_Toc98865288)

[REFERÊNCIAS 42](#_Toc98865289)

[OBRAS CONSULTADAS 47](#_Toc98865290)

[APÊNDICE A – PLANO DE ELABORAÇÃO E GERENCIAMENTO DO PROJETO 48](#_Toc98865291)

[APÊNDICE B – RELATÓRIO DE DESEMPENHO 49](#_Toc98865292)

[APÊNDICE C – VISÃO FUNCIONAL 50](#_Toc98865293)

[APÊNDICE D – VISÃO DOS DADOS 51](#_Toc98865294)

[APÊNDICE E – MODELO INICIAL DA INTERFACE DE USUÁRIO 52](#_Toc98865295)

[APÊNDICE F – VISÕES ESTRUTURAL E COMPORTAMENTAL 53](#_Toc98865296)

[APÊNDICE G – ENTREVISTAS COM USUÁRIOS 54](#_Toc98865297)

[APÊNDICE H – ESTIMATIVA DE TAMANHO E ESFORÇO 55](#_Toc98865298)

[ANEXO A - NOME DO PRIMEIRO ANEXO 56](#_Toc98865299)

# 1 INTRODUÇÃO

Nas paisagens urbanas, além da presença dos característicos edifícios e viadutos, cachorros e gatos não domiciliados são tão presentes que já se tornaram parte integrante deste meio, principalmente nas metrópoles brasileiras. Mesmo sem números oficiais, a Organização Mundial da Saúde estima que milhões de animais, sendo a maioria cães e gatos, estão em estado de abandono, maus tratos ou desamparados.

Para a subsistência de muitos animais em situação de rua, é necessário o auxílio da sociedade protetora dos animais, abrigos e Organizações não governamentais (ONGs). Contudo, devido à alta demanda constante, a superlotação, a falta de recursos para manter e o controle de saúde da população animal, se torna um problema que necessita a atenção e ferramentas capazes de contribuir com tais atividades e acelerar o processo de doação de um animal, aliviando assim a carga de trabalho para as demais instituições.

Perante o exposto, este projeto, nomeado Deu Pet, tem como objetivo apoiar a sociedade protetora dos animais, por meio do desenvolvimento de um sistema de *software* que será disponibilizado via web e mobile. Por meio dele, será facilitada a divulgação dos animais que habitam os abrigos e estão disponíveis para adoção. Também será possível acompanhar e se inscrever em campanhas gerais de saúde dos animais.

Este documento se divide inicialmente em 7 capítulos, sendo que o primeiro capítulo apresentado é a introdução.

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

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

O capítulo 4 apresenta o plano e a gerência do projeto e aborda as gerências do escopo, do tempo, da integração, da quantidade e os riscos.

O capítulo 5 apresenta os requisitos funcionais e não funcionais da aplicação Deu Pet, também exibe o modelo de casos de uso, o modelo conceitual dos dados, o modelo da interface de usuário e as estimativas de esforços.

O capitulo 6 apresenta a arquitetura do sistema de software, a visão estrutural, comportamental, dados e interações.

O capitulo 7 apresenta a conclusão do projeto, 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 fosse realizado. Demonstra-se a relação entre o homem e o animal domesticado a fim de entender as demandas dos interessados como também 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 pontos supracitados, também apresenta o embasamento teórico a respeito das tecnologias aplicadas, técnicas, linguagens de programação e a padronização utilizada no projeto.

## 2.1 O abandono de animais

Os animais domésticos assumem um papel importante na vida do homem desde há milhares de anos. Na realidade, desde a origem da civilização humana que o Homem depende da interação que estabelece com outras espécies animais, tendo essa relação sida inicialmente de predação, e posteriormente de domesticação (Walsh, 2009).

Uma das práticas muito recorrentes na atualidade consiste no abandono irresponsável dos animais. Em especial os domésticos como cães e gatos, um problema que afeta de maneira cada vez mais os grandes centros (DOS SANTOS TRINDADE, et al., 2013).

Com o gradativo aumento da população de cães e gatos presentes pela cidade, problemas pertinentes à saúde pública, trânsito, transtorno na coleta de lixo, além de ataques de animais ferozes em praças e logradouros de grande movimento, é necessário maneiras de abrigar esses animais ou a diminuição da natalidade entre eles.

Abandonar ou maltratar animais é crime previsto pela Lei Federal nº. 9.605, de 12.02.1998, em seu art. 32. Que prevê: “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 14.064/20, sancionada em setembro, aumentou a pena de detenção que era de até um ano para até cinco anos para quem cometer este 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) levanta 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 é a porta de entrada para o grande número de maus tratos 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”. (Ricardo Silva, et al., p. 1, 2018).

Crime de maus-tratos aos animais é submeter ao Abandono; 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 retirar 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 Organizações não governamentais (ONGs) e de protetores independentes que normalmente abrigam mais animais do que tem condições financeiras de manter (Souza e 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. Uma das causas mais levantadas nos últimos tempos é a questão dos fogos de artifício para os nossos “*pets*”, que são extremamente nocivos para os mesmos.

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, Valente (2019)

Na atualidade, 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 bichos em situação de rua em todo o país. Levantamento da ONG AMPARA Animal, 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).

Conforme conversa com o representante da Sociedade Protetora dos Animais de Santa Rita do Sapucaí, Rafael:

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 de animais, conscientização populacional que torna-se um incentivo para que mais pessoas participem de forma voluntária a causa animal.

## 2.3 Arquitetura REST

*REST* (*Representation State Transfer*) é um estilo de arquitetura proposto por Roy Fielding em sua tese “*Architectural Styles and the Design of Network-based Software Architectures*” (FIELDING, 2000) que tem como objetivo a criação de uma arquitetura que possa relacionar de forma condizente a *WEB* com os protocolos HTTP através de estilos arquiteturais.

Inúmeros serviços utilizados em nosso cotidiano como, acesso a um portal *WEB*, compras em lojas de *e-commerce*, sites de *stream* entre outro exemplo, demandam dos servidores requisições de inúmeras aplicações com as mais diversas arquiteturas e tecnologia.

Visando a comunicação entre as partes de forma a unificar a maneira como os dados são requisitados, foram criados os *Web Services*, que são componentes de *software* que, por meio de padrões previamente estabelecidos, fornecem serviços específicos e promovem trocas de informações entre sistemas, independentemente das arquiteturas, tecnologias ou linguagens de programação utilizadas na construção das aplicações envolvidas. (DEITEL H.; DEITEL P., 2010).

A arquitetura *REST*, além de possuir suporte nas principais IDEs presentes no mercado, torna esta arquitetura extremamente utilizada, proporcionando uma gama de soluções em *Web* Services atendendo as mais diversas necessidades. *REST* possibilita que seus serviços sejam acessados através de requisições feitas por *URL* (*Uniform Resource Location* – Localização Uniforme de Recursos). Esta requisição é direcionada através da aplicação conforme definido pela regra de negócio do projeto, provendo orientações dinâmicas quanto à forma de construção e endereçamento das requisições (RIBEIRO, et al., 2016).

Para este projeto, são tratados quatro pontos importantes da arquitetura *REST* (FRANCISCO, 2016)

### 2.3.1 Arquitetura Cliente-Servidor

Fielding (2000) define a arquitetura cliente-servidor como a resposta de uma requisição onde o servidor irá responder com os dados retornados em um formato fixo, ou seja, o cliente ao solicitar um dado ao servidor, sendo que o cliente não terá conhecimento da natureza destes dados, tornando as requisições mais seguras e impedindo que acessos maliciosos ou indevidos sejam realizados ao servidor.

Este tipo de acesso aos dados, cria uma forma padronizada de solicitar e tratar os mesmos, simplificando a manutenção e portabilidade destes dados. Desta forma, cria-se um cenário onde uma mesma base de dados possa ser compartilhada entre as aplicações, seja ela *WEB* ou mobile.

### 2.3.2 Stateless (Sem Estado)

Servidores *Stateless*, definem que as requisições feitas ao servidor, não mantenha seus estados, criando uma independência de requisições onde uma solicitação não irá necessitar do estado anterior. Portanto, cada requisição cliente-servidor deve conter todas as informações necessárias para que a solicitação possa ocorrer e consequentemente, 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 as informações sem a dependência de outra e a resposta não terá nem uma interferência por ser independente.

Em uma situação onde o 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 o seu *token* o qual posteriormente é utilizado pelo cliente para validar posteriores solicitações realizadas pelo cliente.

Em contrapartida, ao se trabalhar com requisições *stateless* é que as solicitações acabam causando 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.

### 2.3.3 Cache

O *cache* em uma arquitetura *REST* tem como papel realizar o balanceamento das cargas de requisições para otimizar o acesso aos dados, desta forma evita-se que o servidor responda solicitações repetidas e gere um fluxo de dados desnecessário evitando o congestionamento da rede. O cache pode ser usado também no servidor que salva recursos frequentemente acessados em memória ou disco, respondendo o 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 não podem ser ‘reaproveitadas’ pelo *cache*. (KUWANO, 2018)



Figura 2: Trabalhando com 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 realizado uma consulta no *cache* e se verificado que os dados são iguais, 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, se tem a necessidade de realizar checagens constantes para verificar a integridade dos dados, já que o servidor pode ter seu estado alterado.

### 2.3.4 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 a 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)

## 2.4 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 bem 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, isso acaba se tornando algo mais complexo. Os *web componnents* buscam uma maneira prática de solucionar estes problemas.

## 2.5 SPRING MVC

O *framework Spring*, atualmente é um dos *frameworks* Java mais conhecido e utilizado. Esse *framework* possibilita a implementação de um grande número de funções, tais como injeção de dependência, persistência de dados e uma implementação para o padrão MVC (*Model-View-Controller*) voltada para a criação de aplicações *WEB* (DEVMEDIA, 2014).

A metodologia MVC traz uma visão diferente para elaboração de projetos a qual 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, lógico e rápido.

Além disso, o *framework* Spring possui uma vasta quantidade de bibliotecas flexíveis que possibilitam a criação de qualquer tipo de solução. Através das funcionalidades de Inversão de Controles e de Injeção de Dependências, se proporciona inúmeras possibilidades e recursos para projetos orientados a microsserviços, *reactives*, *WEB* baseado em nuvem entre outros (SPRING.IO).

# 3 OBJETIVO DO PROJETO

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

## 3.1 FORMULAÇÃO DO PROBLEMA

[Durante o levantamento de ideias para a escolha do projeto, foram identificadas as situações onde o projeto poderia ser útil e aplicável. Descreva qual foi o problema encontrado e que se deseja resolver com este projeto. Pode-se descrevê-lo seguindo a estrutura: qual é o problema; onde ele acontece; como ele acontece; quando acontece; quem é afetado por ele; quais são as consequências desse problema.]

## 3.2 OBJETIVOS

O objetivo geral da aplicação Deu Pet é desenvolver um sistema de software, para plataforma *Web* e *Mobile*, que possibilita às ONGs, abrigos e associações protetoras de animais divulgarem os animais resgatados para adoção e também divulgação de campanhas gratuitas para saúde de animais.

Por meio do aplicativo *Web*, será intermediado pelas instituições protetoras o processo de adoção e também divulgação de campanhas de saúde animal. Pelo aplicativo *Mobile* será possível que pessoas interessadas na adoção de animais, encontrem os mesmos que estão disponíveis para adoção em sua cidade; e também que pessoas interessadas nas campanhas gratuitas de saúde para animais se cadastrem para participar da lista de seleção.

O sistema de *software* deverá permitir aos seus usuários:

1. Acesso *online* para cadastro de animais resgatados pelas instituições protetoras.
2. Acesso *online* para a divulgação de campanhas de saúde para animais pelas instituições protetoras
3. Acesso *online* para o público que busca por adoção de animais.
4. Acesso *online* para o público que busca por campanhas de saúde para o seu animal.

## 3.3 JUSTIFICATIVA

Com base na entrevista realizada à sociedade protetora dos animais de Santa Rita do Sapucaí, é possível averiguar que entre as principais dificuldades, então no topo a falta de adoção dos animais e a divulgação de campanhas de castração gratuitas na cidade.

Visto isso, um sistema que facilite a divulgação dos animais abrigados e campanhas de castração, para um público filtrado que está interessado justamente na adoção ou participação das campanhas, ajudará essas instituições a aumentar o fluxo de adoção e diminuição de natalidade dos animais, abrindo assim possibilidade de resgate de novos animais em situação de rua.

## 3.4 PÚBLICO DE INTERESSE

A aplicação *Web* Deu Pet tem como público de interesse, as ONGs, abrigos e associações protetoras dos animais. A aplicação *Mobile* Deu Pet tem como público de interesse, pessoas que desejam adotar um animal ou participar das campanhas de saúde gratuitas para seus animais.

## 3.5 NÍVEIS DE DECISÃO E GRUPOS FUNCIONAIS ATENDIDOS

O sistema de *software* Deu Pet trabalhará em nível operacional.

1. O usuário interessado na adoção irá visualizar os dados dos animais, escolhendo aquele que mais lhe agradar. Entretanto, deverá se submeter a uma entrevista com a instituição, para finalizar o processo de adoção.
2. O usuário interessado na participação das campanhas de saúde disponíveis, irá sinalizar interesse determinada campanha. Entretanto, será colocado 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.

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

# 4 MÉTODOS GERENCIAIS

Com o intuito de reunir profissionais de gerenciamento de projetos, que desejavam uma troca de experiências e criar boas práticas entre si, foi criado no final da década de 1960 o *Project Management Institute* ( PMI). Deste contexto, os profissionais com um contato mais próximo graças ao PMI, surgiu o *Project Management Body of Knowledge* (PMBoK), um guia que contém diversas boas práticas sobre como proceder na execução de todo ciclo de vida de um projeto.

Fundamentado no PMBoK, neste capítulo será exposto o plano do projeto, fracionado em: 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.

No 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). É composto por cinco grupos de processos de gerenciamento de projeto. São eles: Iniciação, planejamento, execução, monitoramento e encerramento; representados na figura 4, contendo todos os processos necessários para atender os requisitos de projeto segundo o PMI.



Figura 4: Grupo de processos de gerenciamento de projetos.

Fonte: (PMI, 2020)

## 4.1 PLANO DE ELABORAÇÃO E GERENCIAMENTO DO PROJETO

[Esta seção se refere ao preenchimento do *template* disponibilizado pela disciplina de Gerência de Projetos. Deve ser entregue dentro da pasta do Apêndice A. ]

## 4.2 MODELO DE CICLO DE VIDA

Para criação de um *Software*, se faz 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 produção do Software. A organização previamente ao início da produção torna o projeto menos custoso e eleva o nível de qualidade do produto final.

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, são tratados dois tipos de processos de *softwares* sendo o modelo incremental destinado a gerenciar as etapas de entregas principais e o modelo ágil para a realização das atividades diárias e entregas parciais do projeto.

### 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 incremental concilia elementos dos fluxos e processos tanto lineares quanto paralelos (**Figura 5**).



Figura 5 - O Modelo Incremental

Fonte: (PRESSMAN, 2016)

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

O primeiro incremento, tem como objetivo a elaboração e entrega parcial do primeiro artefato, a documentação. A data definida é 23/04/2022 e contempla as sessões de introdução; revisão bibliográfica; objetivos do projeto contendo cinco sessões: formulação do problema, objetivos, justificativa, público de interesse, níveis de decisão e grupos funcionais atendidos; métodos gerenciais e suas quatro primeiras sessões: 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 bibliográficas.

A entrega do segundo incremento em 04/06/2022 contendo: Correção da documentação parcial da fase 1; sessão de resumo; métodos gerenciais com a complementação do plano de elaboração e gerenciamento do projeto, relatório de desempenho da fase 2, descrever a estimativa de tamanho de e esforço e de gerenciamento de configuração; arquitetura e projeto do sistema com as sessões: visão estrutural, visão comportamental, visão de dados e perfil de usuário; capítulo de conclusão; complemento das referências e por fim, a entrega de ao menos 30% dos casos de uso essenciais codificados.

Entrega do terceiro incremento em:

Entrega do quarto incremento em:

### 4.2.2 Modelo Ágil: SCRUM

O método *SCRUM* tem como foco a metodologia ágil para gestão de projetos. Visa otimizar o processo produtivo, consequentemente proporcionando o aumento da produtividade e assertividade garantindo entregas com qualidade.

Este projeto utiliza do *SCRUM* para realização das atividades definidas nas entregas principais do método incremental. O modelo ágil possibilita com que o processo produtivo se torne constante com incrementos realizados diariamente.

Para cada etapa definida no modelo incremental, é realizado o levantamento dos objetivos desta entrega como também o prazo para entrega do incremento. Estes prazos são denominados de *Sprint* que possuem tempo médio de duração inferior aos estabelecidos no modelo incremental (em torno de uma a duas semanas). Então se distribui as atividades entre os membros e após definido qual membro será responsável por qual atividade, é estipulado os prazos para realização das mesmas e diariamente, reuniões chamadas de *Daily Scrum* são realizadas para acompanhamento do processo de produção.

Ao término de cada *Sprint*, é realizada uma retrospectiva do que se foi realizado onde é avaliado e validado as entregas realizadas. Concluído esta etapa, tem se início a uma nova *Sprint*.

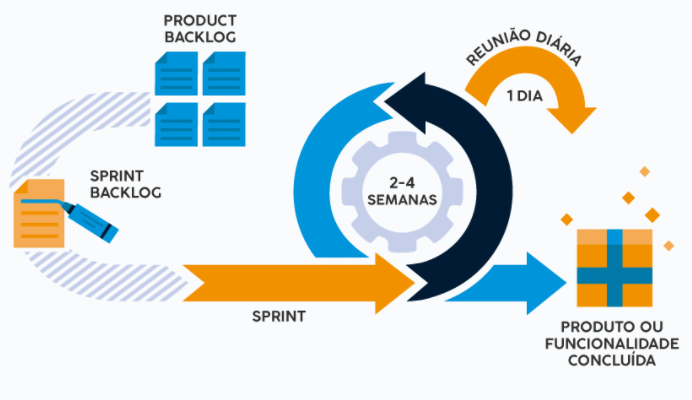


Figura 6 - Processo SCRUM

Fonte: (TECNICON, 2019)

## 4.3 RECURSOS NECESSÁRIOS

Para o desenvolvimento deste projeto, são necessários recursos humanos, de *software* e de *hardware*.

### 4.3.1 Recursos Humanos

Para a realização desse projeto, são necessários três graduandos em Sistemas de Informação, com conhecimentos em *front-end, back-end* e PostgreSQL. São 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 e realização de teste.

### 4.3.2 Recursos de Software

Os recursos de *software* necessários para este projeto:

1. Sistema Operacional Windows 10 de 64 bits;
2. IntelliJ IDEA 2021.3.3
3. PostgreSQL 14;
4. PgAdmin 4 v4.3;
5. Dia 0.97.2
6. Microsoft Excel 2013;
7. Microsoft Word 2016;
8. WBS Tool 0.9 beta;
9. Google drive;
10. Visual Studio Code 1.32.3;
11. Ideas Modeler 11.96;
12. Postman 7.0.6;
13. Git 2.21.0;
14. Git Hub;
15. Java 8;
16. Bootstrap 4.3.1;
17. Angular 8.3.4;
18. Sublime text 3
19. Discord;
20. Google Forms;
21. Adobe XD;
22. Adobe Photoshop;
23. HTML 5;
24. Javascript;

### 4.3.3 Recursos de Hardware

Os recursos de *hardware* necessários para este projeto:

1. 2 notebooks com processador i5 de quarta geração (ou superior), memória ram de 4gb (ou superior).
2. 1 computador desktop com processador i5 de quarta geração (ou superior), memória ram de 4gb (ou superior).
3. 3 mouses ópticos;
4. 1 teclado;
5. 1 monitor;
6. 3 celulares com android;

## 4.4 RELATÓRIO DE DESEMPENHO

O desempenho é o processo de coleta e distribuição de informações sobre o desempenho do projeto, inclusive os relatórios de andamento, medições do progresso e previsões. Abrange a coleta e análise periódica do andamento real do projeto x o andamento esperado.

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

## 4.5 ESTIMATIVAS DE TAMANHO E ESFORÇO

[Deve ser utilizada a técnica por Pontos de Casos de Uso (PCU), a partir da Fase 2, sendo que os resultados devem ser atualizados nas fases seguintes. O Quadro 2 deve apontar os resultados obtidos, em Homem/Hora (Hh). As planilhas que deram origem aos resultados devem ficar no Apêndice H. ]

|  |  |  |  |
| --- | --- | --- | --- |
| Métrica | Fase 2 (Hh) | Fase 3 (Hh)  (reestimativa) | Fase 4 (Hh)  (reestimativa) |
| PCU - Karner |  |  |  |
| PCU - Schneider e Winters |  |  |  |

QUADRO 1 – Estimativa de esforços

[Deve ser inserida uma explicação dos valores apontados no Quadro 1. ]

## 4.6 GERENCIAMENTO DE CONFIGURAÇÃO

[Descreva os papéis e responsabilidades, como são feitos a identificação e o versionamento de cada artefato produzido, a organização do repositório, as ferramentas de software utilizadas para controle das versões. Coloque figura(s) para mostrar o(s) repositório(s) criado(s).]

# 5 ESPECIFICAÇÃO E ANÁLISE DOS REQUISITOS

[Este capítulo deve descrever os requisitos para o desenvolvimento do software. Insira uma pequena apresentação sobre ele.]

## 5.1 REQUISITOS DO SISTEMA DE SOFTWARE

[Insira uma apresentação sobre o assunto tratado nesta seção.]

### 5.1.1 Requisitos Funcionais

[Exemplo para apresentação dos requisitos funcionais:

RF01 – nome do requisito

Descrição:

Prioridade:

[Classifique a prioridade do requisito como essencial, importante ou desejável.]

### 5.1.2 Requisitos não Funcionais

[Os requisitos não funcionais definem as propriedades do sistema, bem como suas restrições. A classificação adotada está baseada na obra Engenharia de Software de Ian Sommerville.]

#### 5.1.2.1 Requisitos de Produto

a) Requisitos de Eficiência

RNF01 – nome do requisito

Descrição:

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

c) Requisitos de Portabilidade

d) Requisitos de Usabilidade

#### 5.1.2.2 Requisitos Organizacionais

a) Requisitos de Entrega

b) Requisitos de Modelagem e Codificação

c) Requisitos de Padrões

#### 5.1.2.3 Requisitos Externos

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

b) Requisitos Éticos

c) Requisitos de Privacidade e Proteção dos Dados

### 5.1.3 Principais Regras de Negócio

[Descreva as principais regras de negócio que estão associadas aos requisitos funcionais. Depois relacione essas regras aos requisitos funcionais descritos anteriormente.]

## 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 devem ser colocados 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 ou do Diagrama de Classes de Negócio. O diagrama deve ser colocado no Apêndice D.]

### 5.2.3 Modelo Inicial da Interface de Usuário

[Colocar os modelos da interface de usuário no Apêndice E.]

# 6 ARQUITETURA E PROJETO DO SISTEMA DE SOFTWARE

[Insira uma apresentação sobre o assunto tratado neste capítulo. A palavra projeto aqui se refere à etapa de construção do sistema de software.]

## 6.1 VISÃO ESTRUTURAL

[Insira uma apresentação sobre o assunto tratado nesta seção.]

[Todos os diagramas devem ser entregue dentro da pasta do Apêndice F.]

### 6.1.1 Diagrama de Pacotes

[Defina os pacotes (agrupamentos lógicos) idealizados para o sistema. Os pacotes geralmente são identificados segundo os padrões de projeto utilizados.]

### 6.1.2 Diagramas de Classes

[De acordo com os pacotes identificados na Seção 6.1.1, construa o diagrama de classes para cada pacote].

### 6.1.3 Diagramas de Objetos

[Recomenda-se construir um diagrama de objetos para pelo menos os objetos das classes de negócio.]

## 6.2 VISÃO COMPORTAMENTAL

[Insira uma apresentação sobre o assunto tratado nesta seção. Todos os diagramas devem ser entregue dentro da pasta do Apêndice F.]

### 6.2.1 Projeto das Interações entre Objetos

6.2.1.1 Diagramas de Sequência

[Para cada cenário dos casos de uso, deverá ser construído um diagrama de sequência ou um diagrama de comunicação.]

6.2.1.2 Diagrama de Visão Geral da Interação

[Mostre as interações que interligam as sequências apresentadas em 6.2.1.1.]

### 6.2.2 Diagramas de Atividades

[Os diagramas de atividades poderão ser feitos para descrever o fluxo entre um ou mais casos de usos ou para detalhar casos de uso específicos, de importância para o projeto.]

[Todos os diagramas devem ser entregue dentro da pasta do Apêndice F.]

## 6.3 VISÃO DOS DADOS

[Insira uma apresentação sobre o assunto tratado nesta seção.]

### 6.3.1 Modelo Lógico

[A partir do modelo conceitual do Capítulo 5, faça o modelo lógico dos dados que devem ser persistidos. O modelo lógico é conhecido também como modelo operacional.]

[Devem ser entregues dentro da pasta do Apêndice D].

### 6.3.2 Dicionário de Dados do Modelo Lógico

[Descreva os elementos de dados presentes no modelo operacional (ou lógico).]

[Devem ser entregues dentro da pasta do Apêndice D.]

## 6.4 PROJETO DA INTERAÇÃO HUMANO-COMPUTADOR

[Apresente o estudo dos perfis de usuários esperados, mostrar os padrões ergonômicos e as heurísticas de usabilidade para a implementação da interface de usuário.]

### 6.4.1 Perfil de Usuário

[Identifique pelo menos três perfis de usuários para o futuro sistema. Em seguida, entreviste 3 possíveis usuários e apresente dados sobre eles, como idade, experiência com outros sistemas de software, experiência no domínio do problema, principais tarefas no sistema, frequência de uso no futuro sistema e preferências de gerais de uso. Devem ser entregues dentro da pasta do Apêndice G.]

# 7 CONCLUSÃO

[Faça um relato das principais atividades realizadas ao longo das fases. Inclua os aspectos mais relevantes observados em cada fase. Apresente quais objetivos do projeto foram alcançados e quais requisitos foram atendidos. Apresente dados que comprovam o alcance desses. Discuta os possíveis pontos de melhoria do projeto e o que poderá ser feito no futuro para continuar o trabalho. Apresente onde o trabalho foi divulgado. Finalize com as lições aprendidas. ]

# REFERÊNCIAS

ADOTAFÁCIL. **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

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

BRASIL. LEI Nº 9.605. **Dispõe sobre as sanções penais e administrativas derivadas de condutas e atividades lesivas ao meio ambiente, e dá outras providências**. Brasília, DF, 12 de fevereiro de 1998. Acesso em: 16 Mar. 2022.

BRASIL. LEI Nº 14.064. **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.** Brasília, DR, 29 de setembro de 2020. Acesso em: 16 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.

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, Paul; DEITEL, Harvey. **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, EMÍLIO. **Desmistificando REST com Java**. 1a Edição, 2016. Disponível em: <http://cafe.algaworks.com/livreto-desmistificando-rest-com-java/>. Acesso em: 17 Mar. 2022.

DOS SANTOS TRINDADE, L.; FELIPE RIBEIRO SANTOS, L.; ELISA GEHRKE, G.; PINTO GOMES, G.; GRADUANDO, G.; TORRES PIMENTEL, F. **O abandono irresponsavel de animais domésticos**. Anais do Salão Internacional de Ensino, Pesquisa e Extensão, v. 2, n. 1, 2 fev. 2013. Acesso em: 16 Mar. 2022.

FERNANDA GARCIA, MARIA. **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-do-brasil/#:~: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.

INSTITUTO PET BRASIL**. País tem 3,9 milhões de animais em condição de vulnerabilidade**., 26 de agosto de 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.

KUWANO, JOÃO PAULO. **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.

MAJOLO, SABRINA; CHIELA, EDIANA;LUSA, TATIANE.. **ONG’s E A IMPORTÂNCIA DE SUAS AÇÕES NO MEIO SOCIAL: SUPERPOPULAÇÃO, ABANDONO E A LUTA PELO BEMESTAR ANIMAL**. Disponível em: <https://eventos.uceff.edu.br/eventosfai\_dados/artigos/cibea2018/855.pdf>. Acesso em: 21 Mar. 2022.

MOUTINHO, F. F. B.; do 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. 2015, 16, 574-588. Acesso em: 17 Mar. 2022.

MOUTINHO, F. F. B.; SERRA, CATHIA, 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/nFpjLbdSnzvz4TSPSbDPy6b/?format=pdf&lang=pt. Acesso em: 21 Mar. 2022

MOZILLA. **Web Components,** c2018. Disponível em: <https://developer.mozilla.org/en-US/docs/Web/Web\_Components>. Acesso em: 17 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 Sanitá, (1994). 30 (2), 223-227. Acesso em: 17 Mar. 2022.

PATRONEK, G.J. & GLICKMAN, LARRY & BECK, ALAN & MCCABE, G.P. & ECKER, C. (1996). **Risk Factors For Relinquishment Of Dogs To An Animal Shelter. Journal of the American Veterinary Medical Association.** 209. 572-81. Acesso em: 17 Mar. 2022.

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

PMI. **Um guia do conhecimento em gerenciamento de projetos. Guia PMBOK** 6a. ed. - EUA: Project Management Institute, 2017. Acesso em: 17 Mar. 2022.

RIBEIRO, MARIA FATIMA; FRANCISCO, Rafael Emanoel. **WEB SERVICES REST CONCEITOS, ANÁLISE E IMPLEMENTAÇÃO**. Revista E.T.C. Educação, tecnologia e cultura, Bahia, ano 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, CAROLINA. **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, VITOR. **Cachorro fica preso mais de uma hora dentro de carro estacionado em rua de Goiânia; vídeo**. 06 de Jan. de 2020. G1. 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., CASTRO, E. E. R. R. DE, SILVA, F. DA, SILVA, R. C. DA, SOUZA, T. F. L. DE, & PEREIRA, S. G. (2018). **LEVANTAMENTO DAS NOTÍCIAS SOBRE MAUS TRATOS À 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, 4(Suppl1), 79–79. Disponível em: <http://psicodebate.dpgpsifpm.com.br/index.php/periodico/article/view/425>. Acesso em: 17 Mar. 2022.

SOUZA, ALEXANDRA CARIBÉ DE ARAÚJO; REIS, SÉRVIO TÚLIO JACINTO. Origem e histórico dos animais resgatados e tutelados por ONGs de proteção aos animais, protetores independentes e 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.

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

STEPHANI LINARD, S.L. **Abandono de animais cresce 61,6% na pandemia e sobrecarrega abrigos.** Agência de notícias do curso de jornalismo da UFCA. 14 set. 2021. Disponível em: <https://sites.ufca.edu.br/agenciacariri/abandono-de-animais-cresce-616-na-pandemia-e-sobrecarrega-abrigos/>. Acesso em: 21 Mar. 2022.

TECNICON. **Metodologia Scrum para a gestão de processos ágeis na indústria**. 2019. Disponível em: <https://www.tecnicon.com.br/blog/411-Metodologia\_Scrum\_para\_a\_gestao\_de\_processos\_ageis\_na\_industria>. Acesso em: 22 Mar. 2022.

VELASCO, ENRIC RUHI. **Web authorization and authentication for single page applications (SPAs)**. 2018. 98 p. Tese (Licenciatura em engenharia de telemática) - Escola Tècnica 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=y. Acesso em: 21 Mar. 2022.

WALSH, F. (2009). **Human-animal bonds I- the relational significance of companion animals.Family Process**, 48, 463-480. Acesso em: 17 Mar. 2022.

# OBRAS CONSULTADAS

# DEVMEDIA. 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.

LOPES, A.S, PEREIRA, D.F, MENDES, T.S. **HELP A PET- SISTEMA DE SOFTWARE PARA AUXÍLIO À CAUSA DA PROTEÇÃO ANIMAL**. 2019. Acesso em: 17 Mar. de 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.

# ANEXO A - NOME DO PRIMEIRO ANEXO