

FACULDADE DE TECNOLOGIA DE SÃO PAULO

LETÍCIA RODRIGUES COSTA

EDGAR SOUZA ARAÚJO

GABRIEL LUIZ DA CRUZ IGNÁCIO

RAPHAELA TUISSI RAMOS

<ADOCÃES: SOFTWARE PARA DIVULGAÇÃO DE ANIMAIS EM SITUAÇÃO
DE ADOÇÃO>

SÃO PAULO

2024

FACULDADE DE TECNOLOGIA DE SÃO PAULO

LETÍCIA RODRIGUES COSTA

EDGAR SOUZA ARAÚJO

GABRIEL LUIZ DA CRUZ IGNÁCIO

RAPHAELA TUISSI RAMOS

< ADOCÃES: SOFTWARE PARA DIVULGAÇÃO DE ANIMAIS EM SITUAÇÃO
DE ADOÇÃO >

Trabalho submetido como exigência parcial
Para a obtenção do Grau de Tecnólogo em
Análise e Desenvolvimento de Sistemas
Orientador: <Professor Banin>

SÃO PAULO
2024

RESUMO

Este trabalho apresenta o desenvolvimento de um site com o objetivo de melhorar a divulgação de animais em situação de rua, utilizando um conjunto de tecnologias composto por PHP, MySQL, HTML, CSS e Javascript. A metodologia adotada baseou-se nos conceitos de engenharia e arquitetura de software, visando garantir um sistema acessível, usável e de fácil manutenção. Inicialmente, foram levantados os requisitos funcionais e não funcionais do sistema, destacando a necessidade de cadastro, busca e detalhamento dos animais, além de um sistema administrativo para gerenciamento dos mesmos. A arquitetura do sistema foi estruturada em camadas, com a separação clara entre a apresentação (frontend), a lógica de negócio (backend) e a persistência de dados (banco de dados). Durante o desenvolvimento, utilizou-se o ambiente XAMPP para configuração local do servidor Apache, banco de dados MySQL e execução de scripts PHP. O frontend foi implementado com HTML, CSS e JavaScript para estrutura, estilização e interatividade das páginas, enquanto o backend, desenvolvido em PHP, processava as solicitações dos usuários e interagia com o banco de dados. Além disso, seguiu-se uma metodologia ágil, com iterações curtas e feedback constante, assegurando a qualidade e aderência aos requisitos estabelecidos. O resultado foi um sistema funcional, que facilita a divulgação e adoção de animais em situação de rua, com uma interface amigável e de fácil uso. A plataforma permite que usuários cadastrem, busquem e obtenham informações detalhadas sobre os animais, também possibilita o contato com os administradores do site para expressar interesse em adoções, além de criar um ambiente administrativo para as ONGs cadastrarem seus animais.

Palavras-Chave: ONGs; Adoção de animais; Aplicação web; Divulgação de animais, usabilidade, PHP.

ABSTRACT

This work presents the development of a website aimed at improving the dissemination of information about stray animals, using a set of technologies composed of XAMPP, PHP, MySQL, HTML, CSS, and JavaScript. The methodology adopted was based on software engineering and architecture concepts, aiming to ensure an accessible, usable, and easy-to-maintain system. Initially, the functional and non-functional requirements of the system were gathered, highlighting the need for registration, search, and detailing of animals, as well as an administrative system for their management. The system architecture was structured in layers, with a clear separation between presentation (frontend), business logic (backend), and data persistence (database). During development, the XAMPP environment was used for local configuration of the Apache server, MySQL database, and execution of PHP scripts. The frontend was implemented with HTML, CSS, and JavaScript for structuring, styling, and interactivity of the pages, while the backend, developed in PHP, processed user requests and interacted with the database. In addition, an agile methodology was followed, with short iterations and constant feedback, ensuring quality and adherence to the established requirements. The result was a functional system that facilitates the dissemination and adoption of stray animals, with a friendly and easy-to-use interface. The platform allows users to register, search, and obtain detailed information about the animals, also enabling contact with site administrators to express interest in adoptions, and creating an administrative environment for NGOs to register their animals.

Keywords: NGOs; Animal adoption; Web application; Animal dissemination; Usability; PHP.

Sumário

INTRODUÇÃO	6
ENGENHARIA DE SOFTWARE	7
2.1 Requisitos Funcionais e Não Funcionais	8
2.2 Diagrama de Caso de Uso.....	11
2.2.1 Detalhes de Cada UC	12
2.3 Diagrama de classes.....	13
2.4 Dicionário de dados	14
2.5 Diagrama de atividades	16
2.6 Diagrama de sequências	17
2.7 Estruturação da Modelagem de Dados.....	20
2.8 Arquitetura de software MVC	21
METODOLOGIAS	23
3.1 Princípios da Metodologia Ágil.....	23
3.2 Implementação da Metodologia Kanban	24
3.3 Ferramentas e Tecnologias	29
3.4 Estrutura XAMPP	29
3.4.1 Configuração do Apache	30
3.4.2 Utilização do PHP e phpMyAdmin	30
3.5 Estruturação do Banco de Dados	30
EXECUÇÃO DO PROJETO	31
4.1 Descrição do projeto	31
4.2 Principais Dificuldades Encontradas	31
4.3 Processo de Prototipagem	31
4.3.1 Menu.....	32
4.3.2 Página Inicial	33
4.3.3 Tela Adotar	37
4.3.4 Tela Ongs Parceiras	38
4.3.5 Perfil Adotantes.....	39
4.3.6 Tela de Login	40
4.3.7 Tela de Administração	40
4.3.8 Tela de Cadastro Adotante e ONG	43
4.4 Fase de Teste	46
CONSIDERAÇÕES FINAIS	48
REFERÊNCIAS.....	49

INTRODUÇÃO

"Pesquisas da União Internacional Protetora dos Animais (UIPA) indicam um aumento expressivo de 400% nas adoções de animais durante os primeiros meses da pandemia. Além disso, conforme revelado no estudo "Radar Pet 2021," aproximadamente 30% dos animais de estimação pertencentes aos tutores entrevistados foram adquiridos durante o período de isolamento social. No Brasil, a principal via pela qual os animais de estimação ingressam nas famílias é através da adoção ou como presentes, conforme indicam os dados reunidos pela Comissão de Animais de Companhia (COMAC). A pesquisa revela que 84% dos gatos e 54% dos cães são acolhidos como membros de suas famílias.

No entanto, paradoxalmente ao aumento nas adoções, o número de animais abandonados também tem aumentado gradualmente. Nos últimos anos, o Brasil tem enfrentado desafios significativos relacionados ao crescimento da população de animais em situação de vulnerabilidade. Em 2018, o número de animais nessas condições era de 3,9 milhões. Surpreendentemente, com o início da pandemia em 2020, esse número cresceu para 8,8 milhões, representando um alarmante crescimento de 126%. É fundamental notar que a maioria dos animais abandonados e vítimas de maus-tratos encontra-se sob os cuidados de organizações não governamentais (ONGs). Essas organizações desempenham um papel crucial na proteção e recuperação de animais em situação de vulnerabilidade. Atualmente, cerca de 60% desses animais são assistidos pelas ONGs, enquanto os 40% restantes são resultado de abandonos. Surpreendentemente, entre os mais de 184 mil animais sob os cuidados dessas organizações, 96% são cães. (IPB, 2022).

Esse cenário destaca a importância de encontrar soluções inovadoras para enfrentar o problema de sobrecarga em relação à capacidade de abrigo das ONGs. É nesse contexto que este TCC busca desenvolver uma plataforma de adoções, que tem o potencial de oferecer inúmeros benefícios, como simplificar a busca para adoção de cães e gatos, uma vez que, estão concentrados em um único site, permite que os adotantes filtrem suas preferências aumentando as chances de adotar, além de agilizar o dinamismo, fornecendo informações essenciais sobre os animais e o processo de adoção. Espera-se expandir o

alcance das instituições e aumentar as adoções, paralelamente reduzindo o número de animais nas ruas, o que contribuirá para a redução de doenças e maus-tratos.

ENGENHARIA DE SOFTWARE

O capítulo a seguir refere-se ao processo de Engenharia de Software da aplicação proposta, esta etapa consiste em quatro camadas; foco na qualidade, processo, métodos e ferramentas. Tais níveis são definidos por Pressman (2016, p.15), onde o compromisso com a qualidade é o alicerce que sustenta todos os aspectos subsequentes, uma vez que é essencial assegurar a excelência em todos os processos, mesmo que isso envolva a revisão de etapas anteriores. O processo, por sua vez, une as diversas camadas tecnológicas, permitindo o desenvolvimento eficiente e pontual do software. Ele estabelece uma metodologia a ser seguida para garantir o sucesso na entrega de tecnologia de software, servindo também como base para o gerenciamento de projetos.

Esse processo proporciona o contexto para a aplicação de técnicas, criação de documentos e garantia de qualidade, além de gerenciar mudanças de maneira adequada. No âmbito dos métodos, são realizadas várias tarefas, que englobam comunicação, análise de requisitos, modelagem de projetos, desenvolvimento de programas, realização de testes e prestação de suporte. E por último, as ferramentas, apresentam suporte automatizado ou sem automatizado para os processos e métodos. Pensando nesse contexto, para garantir a qualidade do *software* e simplificar o desenvolvimento do mesmo, será percorrido as definições, técnicas, modelagem de informações, estrutura, criação, manutenção e documentação.

Inicialmente, realizamos o levantamento de requisitos apresentando as funcionalidades necessárias ao sistema de forma geral, não nos aprofundando nas esmiúças. Em seguida incorporamos a linguagem de modelagem UML (Linguagem de Modelagem Unificada), conforme definida por Grady Booch (2010) “A UML é uma linguagem padronizada para modelagem de sistemas orientados a objetos, sendo utilizada para visualizar, especificar, construir e documentar artefatos de um sistema de software. ” Que nos levou a criação dos diagramas de casos de uso, classes e atividades, essenciais para a modelagem

do software. É importante ressaltar que alguns dos diagramas, como o de sequência, foram elaborados após a conclusão do desenvolvimento, evidenciando uma abordagem dinâmica e adaptável. Em relação ao banco de dados, utilizamos o diagrama de Entidade e Relacionamento (ER), uma técnica modelagem de dados que, segundo Carlos Alberto (2009, p.36), é a abordagem de modelagem de dados mais difundida e amplamente utilizada, na qual se baseia no modelo entidade-relacionamento (ER). Essa abordagem envolve a representação gráfica dos dados por meio de um diagrama entidade-relacionamento (DER). A técnica ER foi criada por Peter Chen em 1976 e é considerada um padrão de fato para a modelagem conceitual de dados, mesmo técnicas mais modernas, como a modelagem orientada a objetos usando a Linguagem de Modelagem Unificada (UML), baseiam-se nos conceitos fundamentais dessa modelagem. (CARLOS ALBERTO, 2009).

No âmbito da arquitetura de software, adotamos o padrão MVC (Model-View-Controller), uma escolha estratégica pois o padrão MVC é fortemente usado no desenvolvimento web mediante à clara separação de responsabilidades, facilitando a manutenção e reusabilidade do código. Essa estrutura modular também promove adaptações ágeis a mudanças nos requisitos e facilidade na fase de testes. Seguimos a mesma lógica ao escolher pelas metodologias ágeis na metodologia de Engenharia de Software.

2.1 Requisitos Funcionais e Não Funcionais

O levantamento de requisitos é uma tarefa extremamente importante e primordial ao desenvolver um novo *software*, seu papel é crucial no ciclo de desenvolvimento, representando as condições essenciais para a concepção e funcionamento eficiente de um sistema, assim como Roque Maitino Neto (2016, p. 59) afirma:

"Requisitos são as condições necessárias para que um determinado evento aconteça. Tome como exemplo uma aula presencial de Engenharia de Software. Para que ela aconteça, é necessário professor, alunos, lousa, giz, carteiras. Todos esses itens formam o conjunto de requisitos da aula. No desenvolvimento de software acontece o mesmo. Fazem parte dos requisitos de um software suas funções, suas características, restrições e todas as demais condições para que ele exista e cumpra seu objetivo."

É um processo que o engenheiro de software adquire um entendimento profundo das expectativas do usuário, define com clareza o propósito do software e avalia se é possível realizar a implementação com sucesso.

Portanto, a seguir, será definido os requisitos funcionais, que são as funcionalidades que o sistema deve executar, englobando as respostas esperadas em relação a diferentes entradas e a partir disso o comportamento esperado em circunstâncias específicas.

"Requisitos funcionais são as especificações que descrevem as funções específicas e os comportamentos do sistema, delineando o que o sistema deve realizar em termos de operações, serviços ou atividades." (PRESSMAN, 2014). A tabela 1 mostra os requisitos funcionais deste projeto.

Tabela 1 – Requisitos Funcionais

Identificação	Requisitos Funcionais
RF001	Cadastro de adotantes realizado através de uma pessoa que não tenha cadastro no sistema.
RF002	Cadastro de usuários administrativos realizado através da administração da própria ONG que ainda não possua cadastro no sistema.
RF003	Prover para qualquer navegante da web a visualização de todos os animais cadastrados na página principal
RF004	Filtragem dos dados por: cor, porte, sexo, idade, ONG e região. Esse requisito está diretamente relacionado com o requisito RF003.
RF005	Usuários administrativos serão responsáveis pela adição, edição ou remoção dos cadastros de animais e informações de adoção.
RF006	Prover todas as informações do animal escolhido e da ONG que ele está situado ao usuário com perfil adotante.

Fonte: Elaborado pelo autor (2024)

"Os requisitos não funcionais abrangem aspectos críticos para o sucesso do sistema, como escalabilidade, segurança e eficiência, que transcendem as

funcionalidades básicas." (SOMMERVILLE, 2011). A Tabela 2 apresenta os requisitos não funcionais deste projeto.

Tabela 2 – Requisitos Não Funcionais

Identificação	Requisitos Não Funcionais	Categoria
RNF001	Cada senha de usuário é pessoal e intransferível. É de extrema importância que os usuários não repassem suas senhas ou palavras-passe a outros usuários, uma vez que diferentes tipos de permissões de acesso são atribuídos a cada categoria de usuário.	Segurança
RNF002	O sistema deve ser responsivo, para ser usado em celulares, tablets e computadores, adaptando-se automaticamente para oferecer uma boa experiência em qualquer dispositivo.	Acessibilidade
RNF003	A plataforma deve ser ágil ao carregar fotos e informações dos animais, evitar acessos demorados e ter um tempo de resposta rápido para atrair mais usuários e agilizar o trabalho das ONGs	Desempenho
RNF004	O sistema web de forma geral deve ser intuitivo, com instruções claras e de fácil compreensão, uma vez que, nem todos os usuários sabem mexer na internet e o objetivo é ter o máximo de usuários possíveis.	Usabilidade
RNF005	O banco de dados que suportará a aplicação deve ser robusto e escalável para armazenar uma grande quantidade de informações relacionadas aos animais e usuários, garantindo desempenho consistente com o aumento no volume de dados.	Desempenho

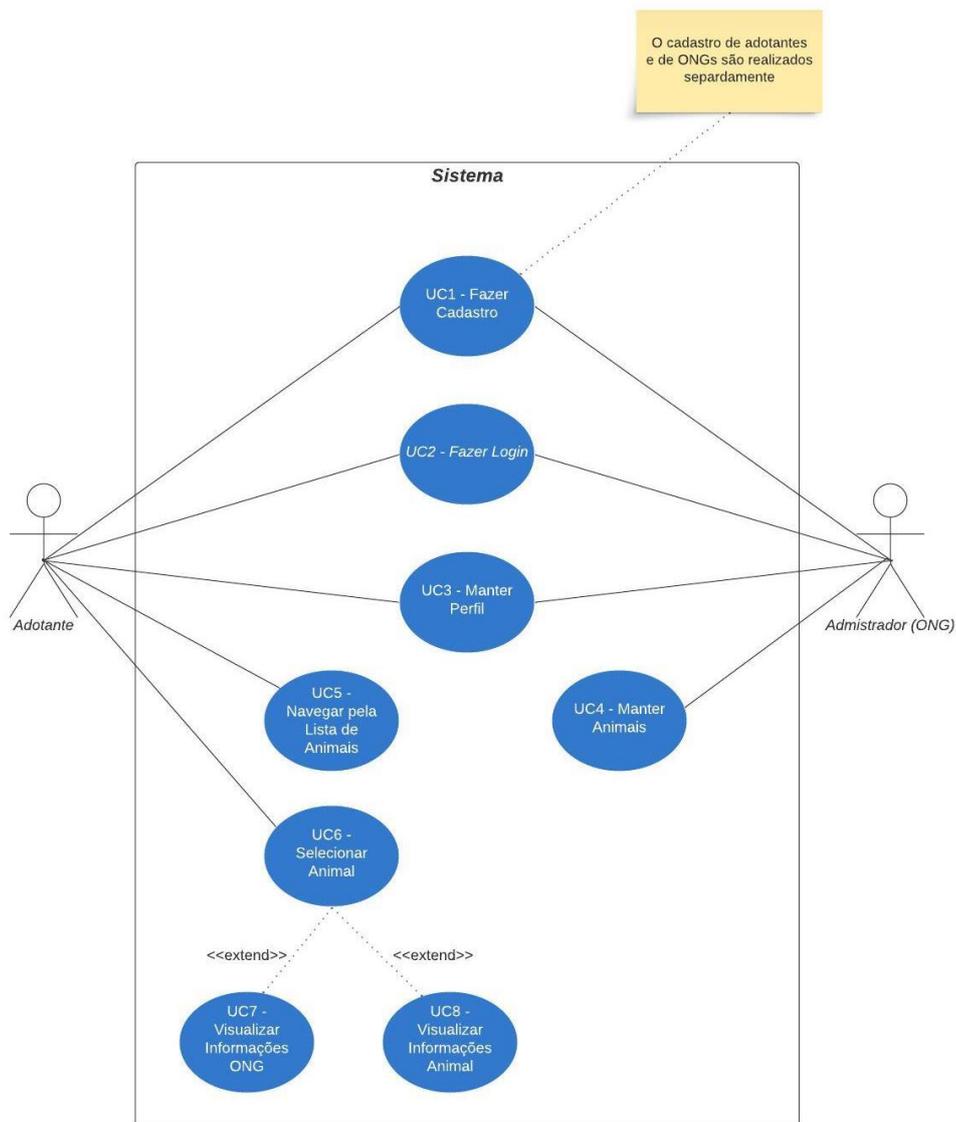
Fonte: Elaborado pelo autor (2024)

2.2 Diagrama de Caso de Uso

Segundo Ivar Jacobson (1992, p. 157), um caso de uso é descrito como um "documento narrativo" que detalha a sequência de eventos em que um ator (usuário ou sistema externo) interage com um sistema para realizar um processo específico.

A figura 1 apresenta o diagrama de casos de uso, resultante da análise dos requisitos funcionais realizados anteriormente. Este diagrama proporciona uma visualização clara e objetiva das interações entre os usuários e o sistema. O mesmo foi desenvolvido através da ferramenta de criação de diagramas Lucidchart.

Figura 1 – Caso de Uso



Fonte: Elaborado pelo autor (2024)

2.2.1 Detalhes de Cada UC

- Atores: Adotante e Administrador (ONG)

Adotante: Usuário no sistema que faz o cadastro para buscar o animal de sua preferência e visualizar as informações detalhadas do animal escolhido e da ONG no qual se encontra.

Administrador (ONG): Os usuários administradores são aqueles que se registram como ONGs, com permissão para adicionar, editar e remover os animais da listagem, além de manter as informações atualizadas.

UC1 – Fazer Cadastro

Neste caso de uso, os usuários têm duas opções de cadastro, de adotante ou de ONG, com campos de preenchimento distintos.

UC2 – Fazer Login

O login de adotantes e ONGs é necessário para acessar o sistema

UC3 – Manter Perfil

Ambos os usuários adotantes ou ONGs conseguem visualizar seu próprio perfil e editar ou remover os dados uma vez adicionados. Somente os donos das contas tem permissão de manter seus perfis, não é possível usuários, sejam adotantes ou ONGs, modificar outros perfis.

UC4 – Manter Animais

Este caso de uso possibilita apenas as ONGs cadastrar, editar ou remover os animais pertencentes à sua base de dados.

UC5 – Navegar pela Lista de Animais

O usuário adotante consegue navegar pela lista de animais disponíveis para a adoção e pesquisar especificamente o que procura utilizando os filtros.

UC6 – Selecionar Animal

Quando o usuário adotante encontra o animal de sua escolha, ele pode selecioná-lo.

UC7 – Visualizar Informações ONG

Ao selecionar o animal em questão, é exibido as informações da ONG na qual ele se encontra, como localização e meios de contato.

UC8 – Visualizar Informações Animal

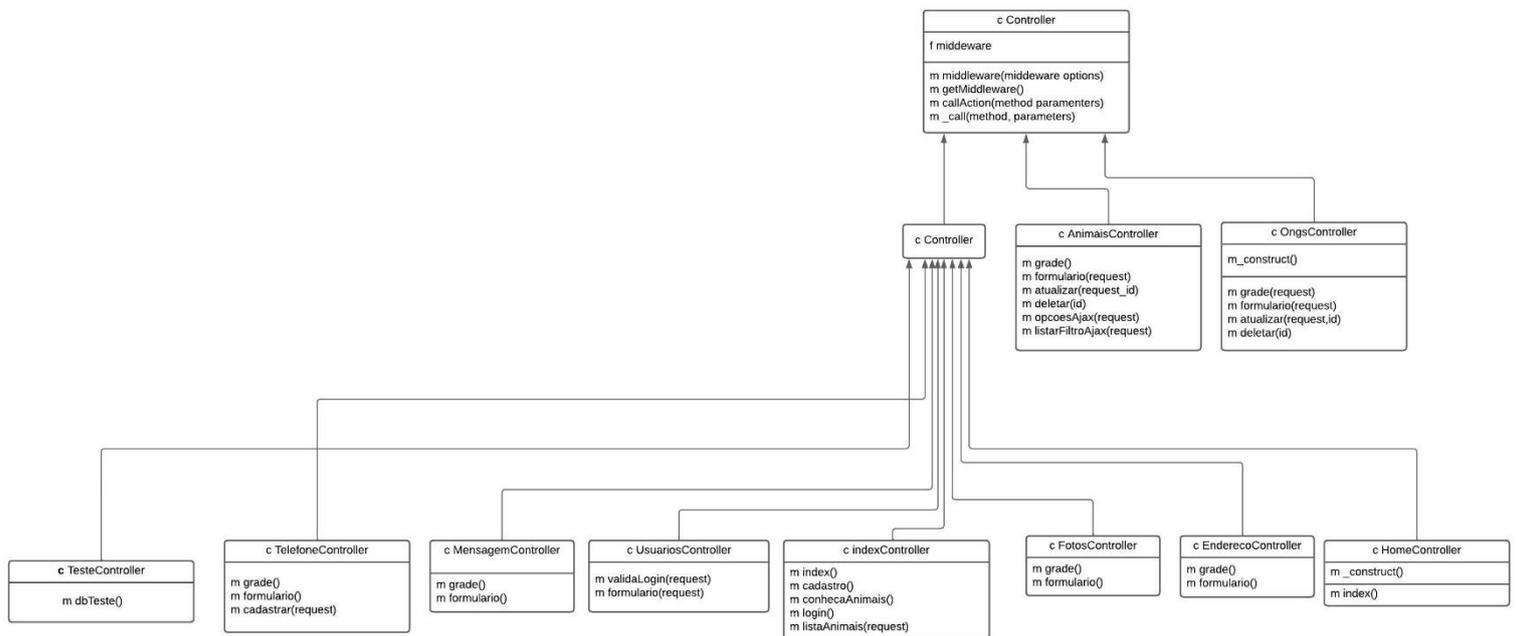
Ao selecionar o animal em questão, é exibido informações detalhadas sobre ele, como a cor, o porte, observações e etc.

2.3 Diagrama de classes

O diagrama de classes demonstra de que maneira as classes estão interligadas, mostrando como se relacionam, se complementam e trocam informações entre si. Essa representação gráfica é essencial para o entendimento estrutural do sistema durante o processo de modelagem.

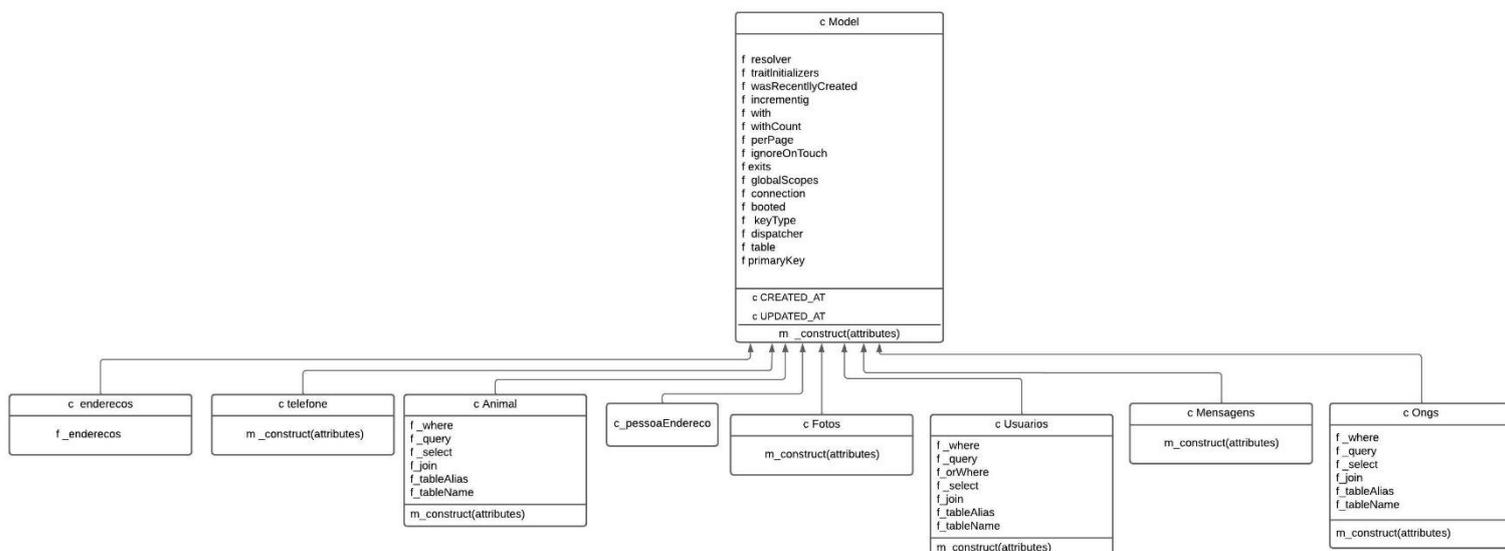
Segundo Lazaro, o principal objetivo desse diagrama é “[...] apresentar uma visão estática de como as classes estão organizadas, preocupando-se em como definir a estrutura lógica delas [...]”. (LAZARO, 2009, p. 114).

Figura 2 – Diagrama de Classes (Controller)



Fonte: Elaborado pelo autor (2024)

Figura 3 – Diagrama de Classe (Gerenciamento)



Fonte: Elaborado pelo autor (2024)

2.4 Dicionário de dados

O dicionário de dados é um documento de metadados que se encontra em um banco de dados relacional. Sua função é descrever e mapear a estrutura de dados de um sistema, incluindo informações detalhadas sobre tabelas, campos e relações (INMON, 2005).

Tabela: ADOTANTES						
	Campo	Descrição	Tipo	Tamanho	DEC	
PK	idAdotantes	Código Adotante	do INT	10	NOT NULL AUTO_INCREMENT	
	nomeAdotante	Nome Adotante	do VARCHAR	45	NOT NULL	
	CPF	CPF Adotante	do VARCHAR	14	NOT NULL	
	Email	Email adotante	do VARCHAR	100	NOT NULL	
	Senha	Senha Adotante	do VARCHAR	20	NOT NULL	

Tabela: ONGs					
	Campo	Descrição	Tipo	Tamanho	DEC
PK	idONG	Código da ONG	INT	10	NOT NULL AUTO_INCREMENT
	nomeOng	Nome da ONG	VARCHAR	120	NOT NULL
	CNPJ	CNPJ da ONG	VARCHAR	18	NOT NULL
	Site	Site da ONG	VARCHAR	100	NOT NULL
	Senha	Senha da Ong	VARCHAR	20	NOT NULL

Tabela: Telefone					
	Campo	Descrição	Tipo	Tamanho	DEC
PK	idTelefone	Código do Telefone	INT	10	NOT NULL AUTO_INCREMENT
	número	Número de Telefone	VARCHAR	12	NOT NULL
FK	idAdotantes	Código do Adotante	INT	10	NULL
FK	idONG	Código da ONG	INT	10	NULL

Tabela: Endereco					
	Campo	Descrição	Tipo	Tamanho	DEC
PK	idEndereco	Código do Endereco	INT	10	NOT NULL AUTO_INCREMENT
	bairro	Nome do Bairro	VARCHAR	50	NOT NULL
	numero	Número da residência	VARCHAR	6	NOT NULL
	rua	Nome da Rua	VARCHAR	120	NOT NULL
	cidade	Nome da Cidade	VARCHAR	40	NOT NULL
	estado	Nome do Estado	CHAR	2	NOT NULL
	cep	Numero do CEP	VARCHAR	9	NOT NULL
FK	idAdotantes	Código do Adotante	INT	10	NULL
FK	idONG	Código da ONG	INT	10	NULL

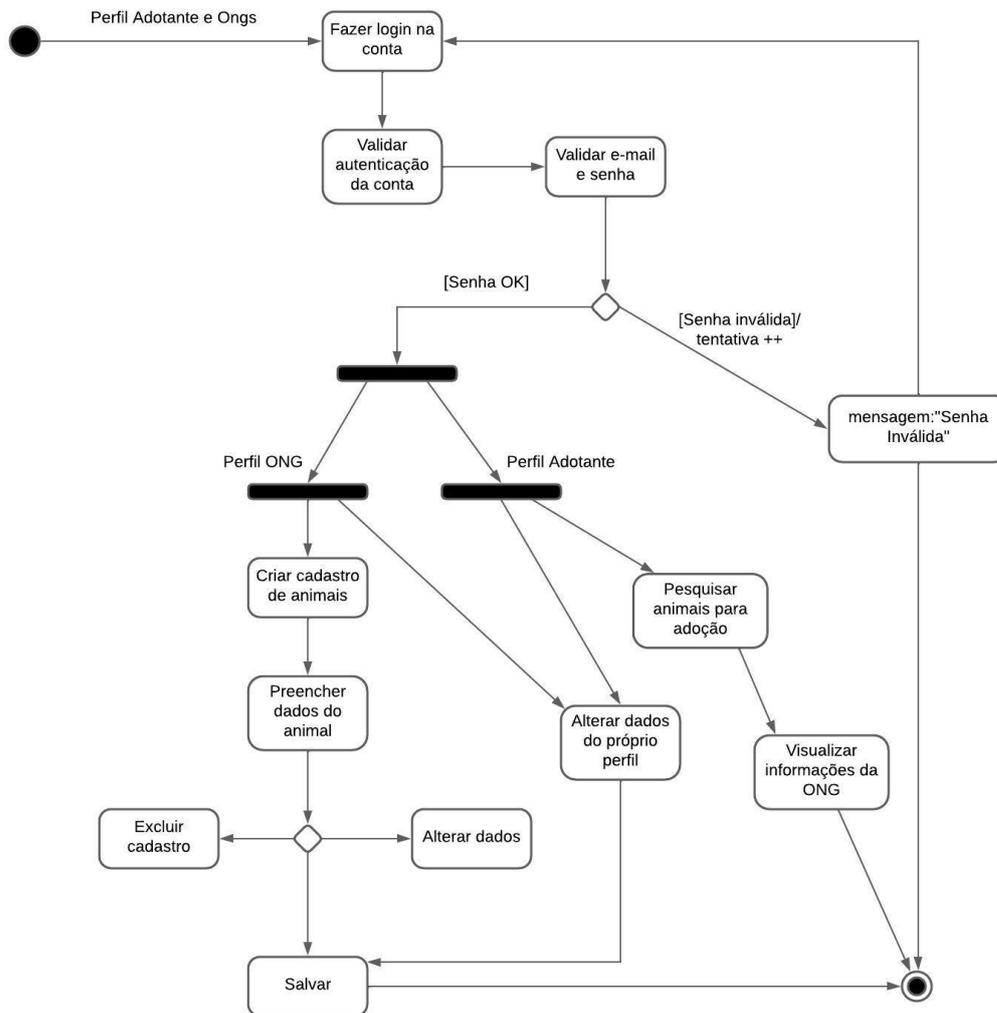
Tabela: Animais					
	Campo	Descrição	Tipo	Tamanho	DEC
PK	idAnimais	Código do Animal	INT	10	NOT NULL AUTO_INCREMENT
FK	ONGs_idONG	Código da ONG	INT	10	NOT NULL
	nomeAnimal	Nome do Animal	VARCHAR	55	NOT NULL
	raca	Nome da Raca	VARCHAR	30	NOT NULL
	dataNascimento	Data de nascimento do Animal	DATE	--	NOT NULL
	genero	Gênero do Animal	CHAR	1	NOT NULL
	porte	Porte do Animal	CHAR	1	NOT NULL
	cor	Cor do Animal	VARCHAR	25	NOT NULL
	castrado	Animal castrado sim ou nao	CHAR	1	NOT NULL
	vacinas	Vacinas do Animal	TEXT	--	NOT NULL
	observacao	Observação do Animal	TEXT	--	NOT NULL
	tipo	Tipo de Animal	CHAR	1	NOT NULL

Tabela: URLS					
	Campo	Descrição	Tipo	Tamanho	DEC
PK	idURL	Código da URL	INT	10	NOT NULL AUTO_INCREMENT
	url	Link da URL	TEXT	--	NOT NULL
	nome	Nome da URL	CHAR	25	NULL
FK	idAdotantes	Código do Adotante	INT	10	NULL
FK	idONG	Código da ONG	INT	10	NULL
FK	idAnimais	Código do Animal	INT	10	NULL

2.5 Diagrama de atividades

Como o próprio nome já diz, esse Diagrama consiste em representar visualmente o fluxo de atividades em um sistema. De acordo com Lazaro, é o diagrama “com mais ênfase no nível de algoritmo da UML e provavelmente um dos mais detalhistas”. (LAZARO, 2009, p. 306).

Figura 4 – Diagrama de Atividades

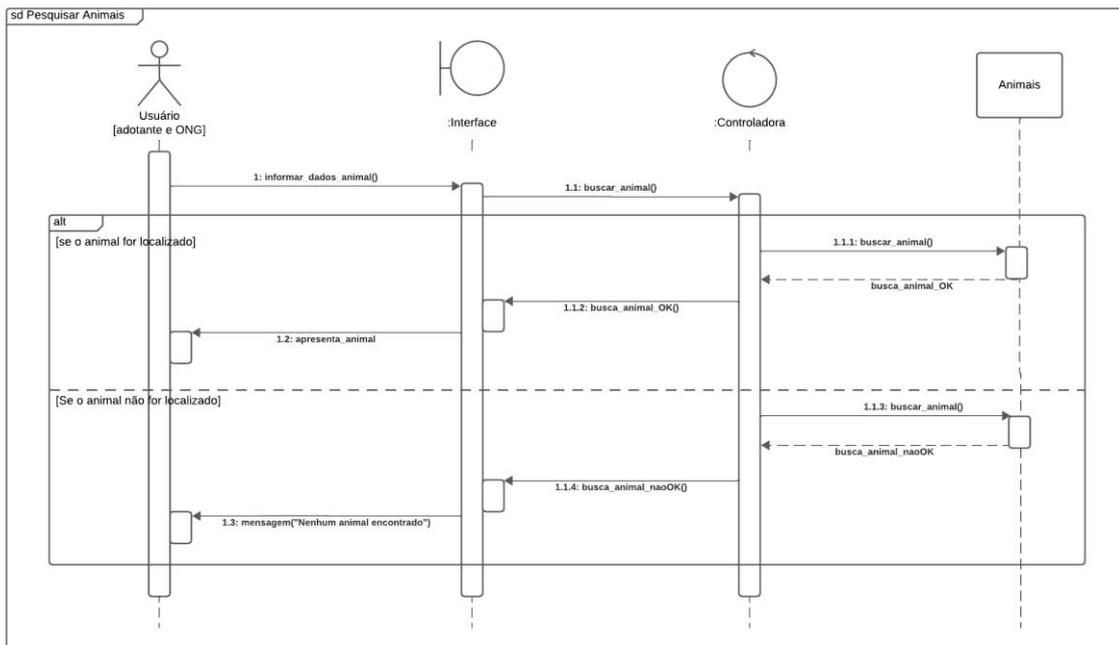


Fonte: Elaborado pelo autor (2024)

2.6 Diagrama de seqüências

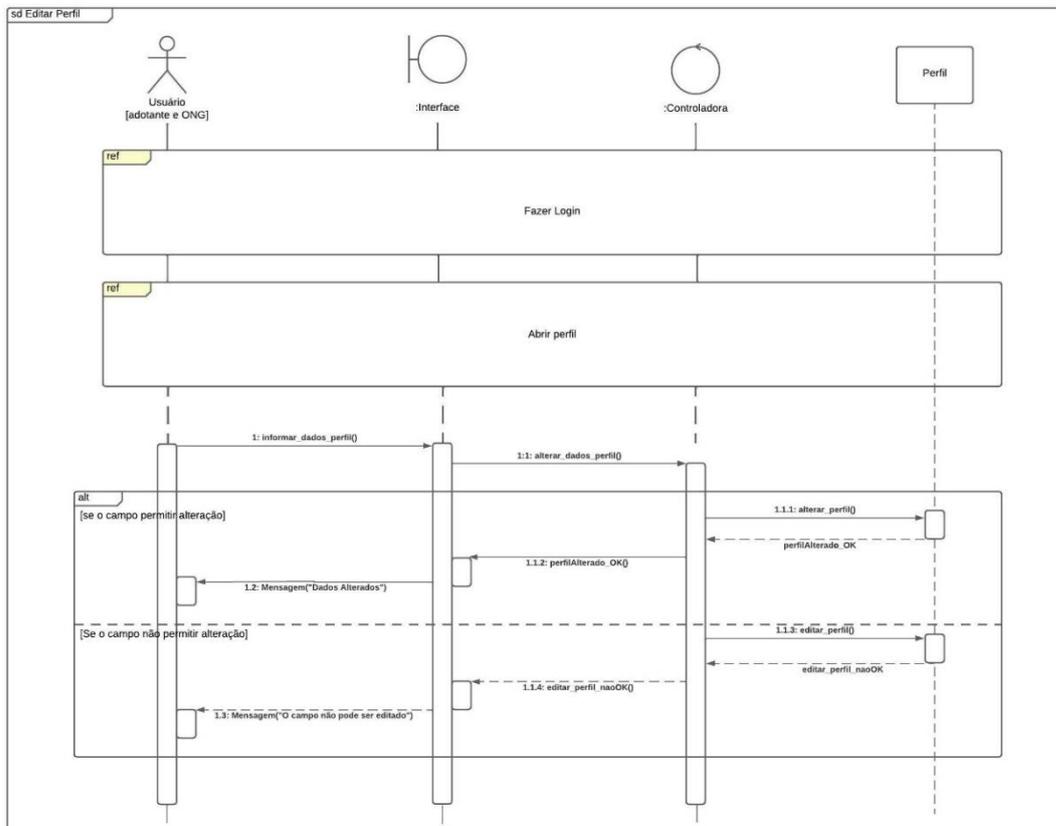
O diagrama de seqüência é um componente essencial na modelagem comportamental, visando determinar a ordem e interações de eventos em um processo específico. Ele identifica mensagens entre elementos e a seqüência em que são disparadas, detalhando os métodos chamados e a interação entre objetos. Normalmente, relaciona-se a um caso de uso, fornecendo uma visão detalhada do processo vinculado a um ator. (LAZARO, 2009, p. 215).

Figura 5 – Pesquisar Animal



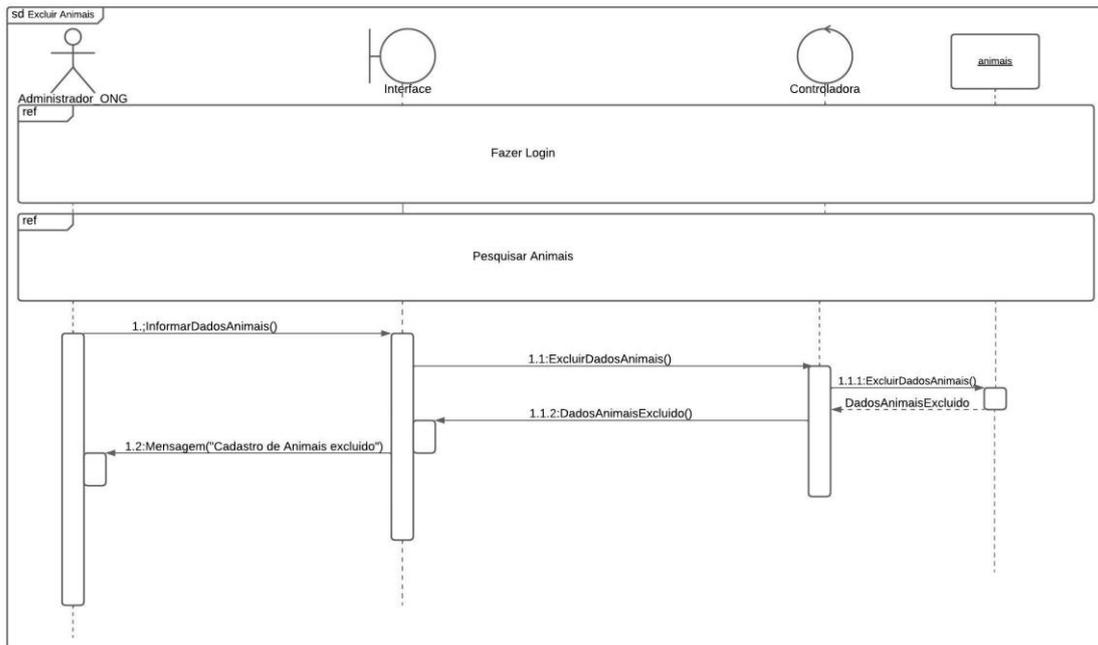
Fonte: Elaborado pelo autor (2024)

Figura 6 – Editar Perfil



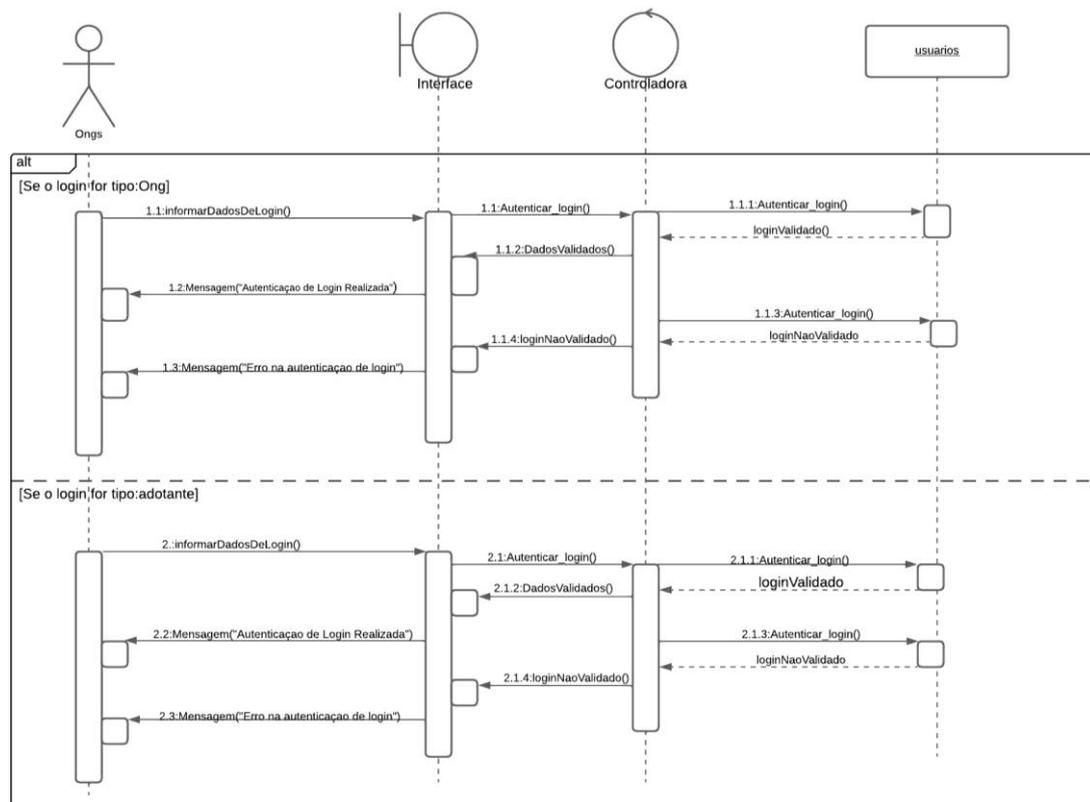
Fonte: Elaborado pelo autor (2024)

Figura 7 – Excluir Animais



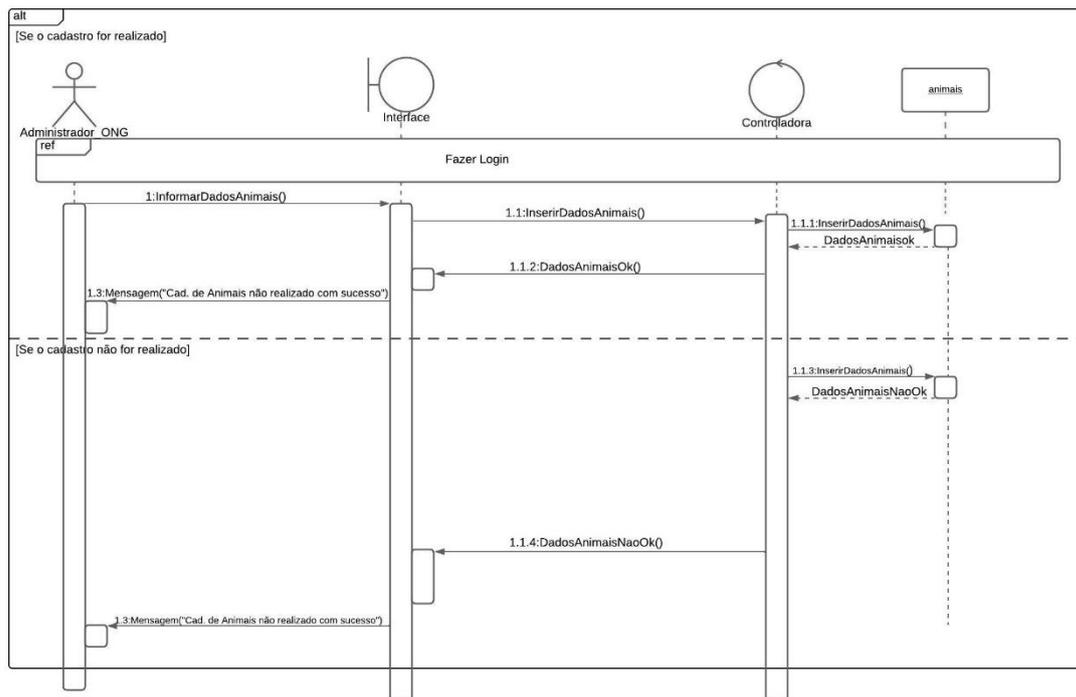
Fonte: Elaborado pelo autor (2024)

Figura 8 – Login



Elaborado pelo autor (2024)

Figura 9 – Cadastrar Animais



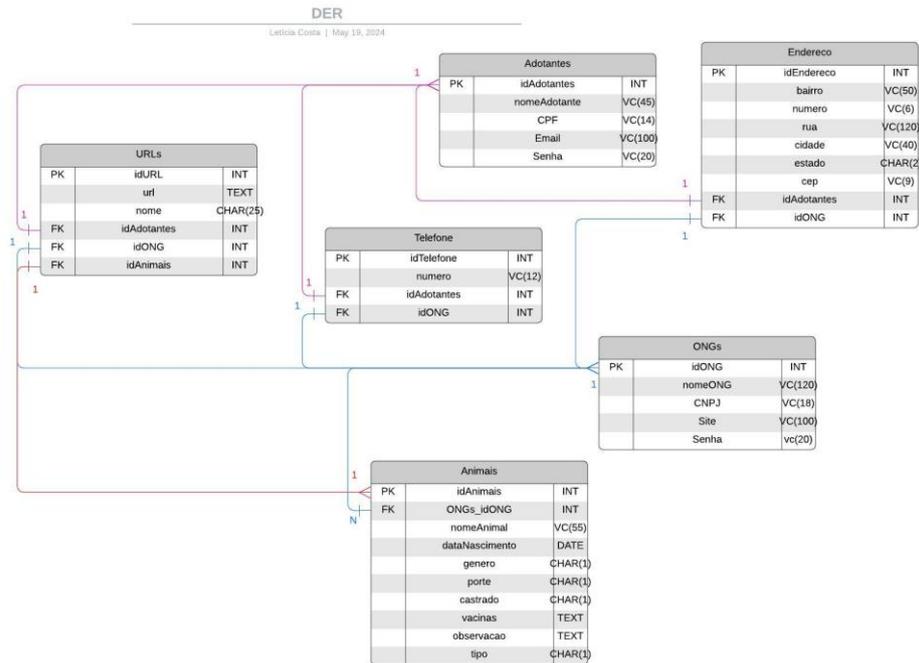
Elaborado pelo autor (2024)

2.7 Estruturação da Modelagem de Dados

A modelagem de dados cria uma representação abstrata e estruturada dos dados que serão armazenados e manipulados pelo sistema. Ela visa entender e documentar as informações necessárias para o funcionamento do sistema, identificando entidades, relacionamentos e características dos dados. A modelagem de dados ajuda a garantir a consistência e a integridade dos dados, proporcionando uma base sólida para o design e a implementação de sistemas.

O Diagrama de Entidade-Relacionamento é uma ferramenta gráfica utilizada na modelagem de dados para representar as entidades envolvidas em um sistema e os relacionamentos entre elas. Desenvolvido por Peter Chen, o modelo ER é composto por entidades, que são objetos do mundo real ou conceitual, e relacionamentos, que são associações significativas entre essas entidades. (CHEN, 1990).

Figura 10 – DER



Elaborado pelo autor (2024)

2.8 Arquitetura de software MVC

A arquitetura Model-View-Controller (MVC) é um padrão de arquitetura de software que contribui para otimizar a velocidade entre as requisições dos usuários. Por padrão, ela é composta por três componentes, que são descritos por Erich Gamma e seus colegas sendo “O Modelo, o objeto de aplicação, a Visão é a apresentação na tela e o Controlador é o que define a maneira como a interface do usuário reage às entradas do mesmo” (GAMMA, 1994, p. 42). Dentro do contexto da arquitetura MVC, alguns padrões de design são especialmente relevantes, como mencionado no livro "Design Patterns, escrito por Erich Gamma e seus colegas, como O Factory Method, que é utilizado na camada Model (Modelo). Esse padrão oferece uma abordagem para criar objetos sem especificar a classe exata do objeto que será criado. No Controller (Controlador), o padrão Adapter é aplicado para integrar diferentes interfaces ou sistemas. Isso é particularmente útil ao lidar com diversas fontes de dados ou serviços externos. O padrão Composite permite tratar objetos individuais e composições de objetos de maneira uniforme. No contexto do MVC, pode ser aplicado para lidar com a complexidade das relações entre diferentes elementos

da interface do usuário (GAMMA, 1994, p. 46). Por isso, definimos as camadas da seguinte forma:

Model (Modelo)

Gerencia dados e controla seu comportamento com base em funções e regras de negócios. Recebe informações do Controller, valida e fornece respostas apropriadas. O padrão Factory Method é usado nesta etapa para instanciar objetos e auxiliar em futuras expansões do sistema.

Controller (Controlador)

Intermedeia requisições entre View e Model, processando dados do usuário e repassando para outras camadas. Analogicamente, atua como o maestro da orquestra. Implementamos padrões estruturais, como o Adapter, para integrar diferentes interfaces, e o Composite, para lidar com hierarquias de informações.

View (Visão)

Apresenta informações visualmente ao usuário, focando em recursos de aparência. É a interface de comunicação com o usuário, transmitindo questionamentos ao Controller e entregando respostas. Ao gerenciar a interação entre os componentes MVC, é aplicado padrões comportamentais como o Observer. Esse padrão permite que a visão (View) seja notificada e atualizada quando houver alterações nas informações dos animais no modelo (Model).

A interação entre esses componentes segue um fluxo sequencial: o usuário interage com a View, o Controller recebe as informações e as encaminha para o Model, que avalia os dados e transmite uma resposta. O Controller, por sua vez, notifica a View sobre a validação, permitindo que esta apresente graficamente o resultado (LARMAN, 2004).

Optamos pela escolha desta arquitetura pois, além de ser a mais utilizada para aplicações web, seus benefícios são diversos incluindo a facilidade de manutenção ao decorrer do desenvolvimento pois alterações em uma camada não devem afetar diretamente as outras. Isso também contribui para a escalabilidade, permitindo que diferentes partes do sistema evoluam de forma independente. A separação de responsabilidades facilitando a realização de testes unitários e de integração em cada componente individual. Além disso, ela

torna o projeto escalável, por conta da clara definição de responsabilidades, proporciona um sistema mais adaptável a mudanças nos requisitos, sem afetar desnecessariamente outras partes do código.

O MVC pode ser facilmente usado em diferentes contextos além da programação web, como na mobile ou desktop, utilizando frameworks específicos, como o Spring MVC (Java) ou Ruby on Rails (Ruby). Dessa forma, a escolha da arquitetura MVC se mostra vantajosa no projeto, proporcionando uma base sólida para o desenvolvimento do sistema, alinhando-se aos princípios de segurança, organização e eficiência na manipulação de dados e na interação com os usuários.

METODOLOGIAS

3.1 Princípios da Metodologia Ágil

O desenvolvimento de software requer uma metodologia para gerenciar seus processos e etapas. No processo, o tempo é mais consumido em análises do que no desenvolvimento e testes. Mudanças significativas podem ocorrer ao longo do projeto, alterando o fluxo de trabalho e as metodologias, sendo algumas preventivas e corretivas. Poole e Huisman (2001) afirmam:

A metodologia ágil no desenvolvimento de software é um conjunto de abordagens e práticas que enfatizam a flexibilidade, a colaboração e a entrega incremental de software. Ela se concentra em responder a mudanças, valorizar a interação com os stakeholders e criar produtos de alta qualidade de forma eficiente.

Portanto, é possível afirmar que os métodos ágeis aplicados ao desenvolvimento de software destacam-se pela adaptabilidade, trabalho em equipe e entregas frequentes e graduais de funcionalidades. Essas práticas favorecem a flexibilidade, permitindo que o projeto responda rapidamente às mudanças e aos feedbacks contínuos dos usuários. A colaboração estreita entre todos os participantes do projeto é crucial para garantir que as entregas incrementais atendam às necessidades reais e urgentes, proporcionando um desenvolvimento mais eficiente e alinhado com os objetivos definidos. (ANDERSON, 2005).

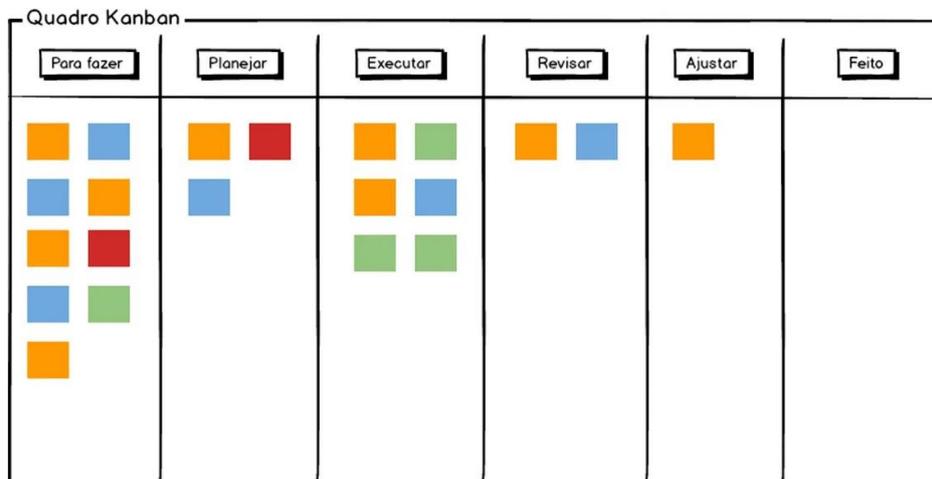
A metodologia ágil de desenvolvimento de software teve suas raízes no Manifesto Ágil, que foi elaborado em fevereiro de 2001 por um grupo de 17 desenvolvedores de software que estavam insatisfeitos com as abordagens tradicionais de desenvolvimento de software. Esse manifesto foi criado durante uma reunião em Utah, nos Estados Unidos, e estabeleceu os princípios e valores-chave para o desenvolvimento ágil (HUSMAN,2001).

Embora o Manifesto Ágil tenha sido redigido em 2001, muitas das práticas e ideias que compõem a metodologia ágil têm origens anteriores, em conceitos como Lean Thinking, Scrum, Extreme Programming (XP) e outras abordagens de desenvolvimento iterativo e incremental. Portanto, embora o termo "ágil" tenha sido formalizado em 2001, as práticas e ideias por trás do desenvolvimento ágil podem ser rastreadas até décadas anteriores, com várias contribuições de diferentes metodologias e pensadores na área de desenvolvimento de software (BOEHM,2002).

3.2 Implementação da Metodologia Kanban

O Kanban no desenvolvimento de software utiliza quadros visuais, sejam físicos ou digitais, divididos em colunas que representam diferentes estágios do processo de desenvolvimento. Nos modelos mais básicos, essas colunas incluem 'A Fazer', 'Em Progresso' e 'Concluído'. As tarefas são representadas por cartões ou post-its que são movidos pelas colunas conforme o trabalho avança, facilitando a visualização (HUSMAN, 2001). Um quadro mais completo envolve ainda mais colunas como na figura 11 a baixo:

Figura 11 – Quadro Kanban



Fonte: <https://hbfs.com.br/o-que-e-kanban/>

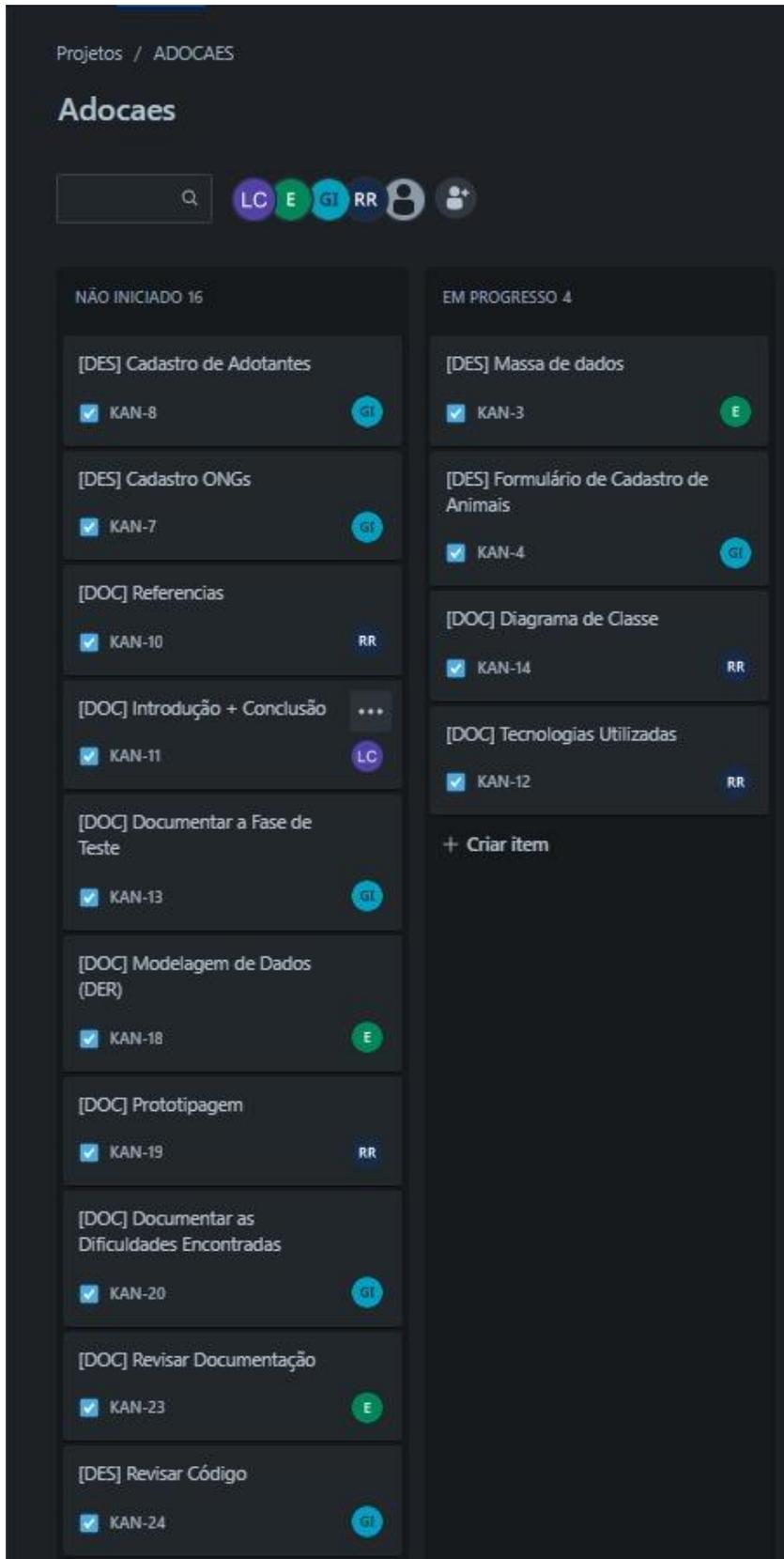
No exemplo acima, inicialmente, temos a coluna "A Fazer," onde listamos tarefas necessárias, mas ainda não iniciadas. À medida que começamos a trabalhar nelas, movemos para a coluna "Em Execução." Quando as tarefas avançam, passam para a revisão precisam de aprovação para serem definidas como prontas, entretanto, caso seja reprovada é direcionada para a fase de ajuste.

Tal metodologia teve sua origem na Toyota nos anos 1950 e foi desenvolvida como parte do Sistema Toyota de Produção (Toyota Production System - TPS), focado em otimizar a eficiência da produção industrial. O sistema Kanban foi usado na fabricação para controlar o fluxo de peças e materiais, assegurando que as quantidades corretas fossem produzidas no momento certo, evitando excessos e reduzindo desperdícios. O conceito de Kanban e como ele pode ser aplicado não apenas em processos de produção industrial, mas também em cenários diferentes. No desenvolvimento de software, o sistema Kanban é usado para limitar o trabalho em progresso (WIP), embora não haja cartões físicos como em sistemas tradicionais de Kanban. Em vez de cartões físicos, esses kanbans virtuais são representações visuais dos itens de trabalho e são usados para inferir quando acrescentado mais tarefas. (ANDERSON, 2005).

Por conta dos benefícios encontrados e pela flexibilidade, escolhemos utilizar a metodologia Kanban para o nosso projeto de desenvolvimento de software.

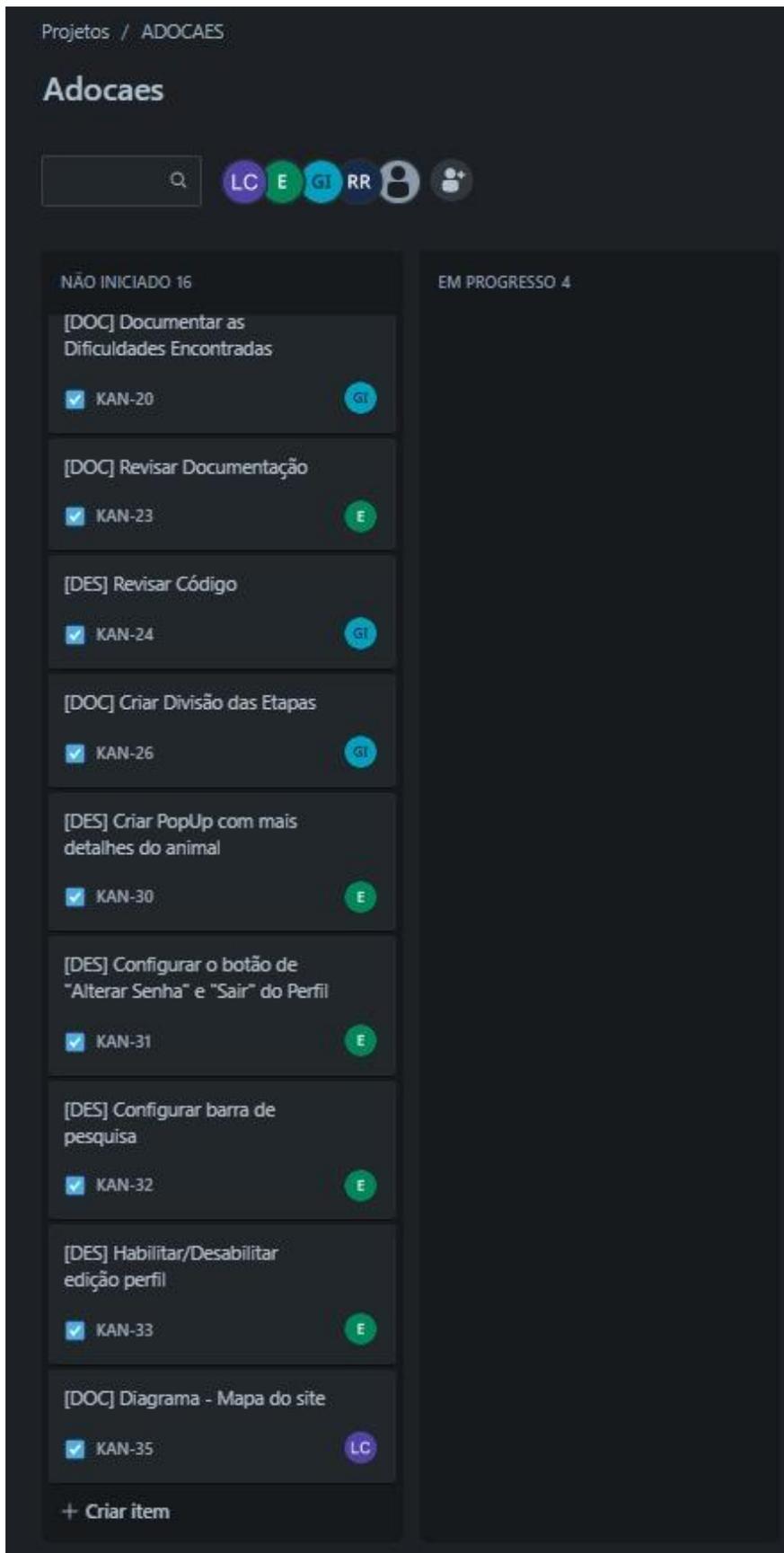
3.2.1 Planejamento e Distribuição de Tarefas

Figura 12 – Cronograma Kanban (A fazer e Em Progresso)



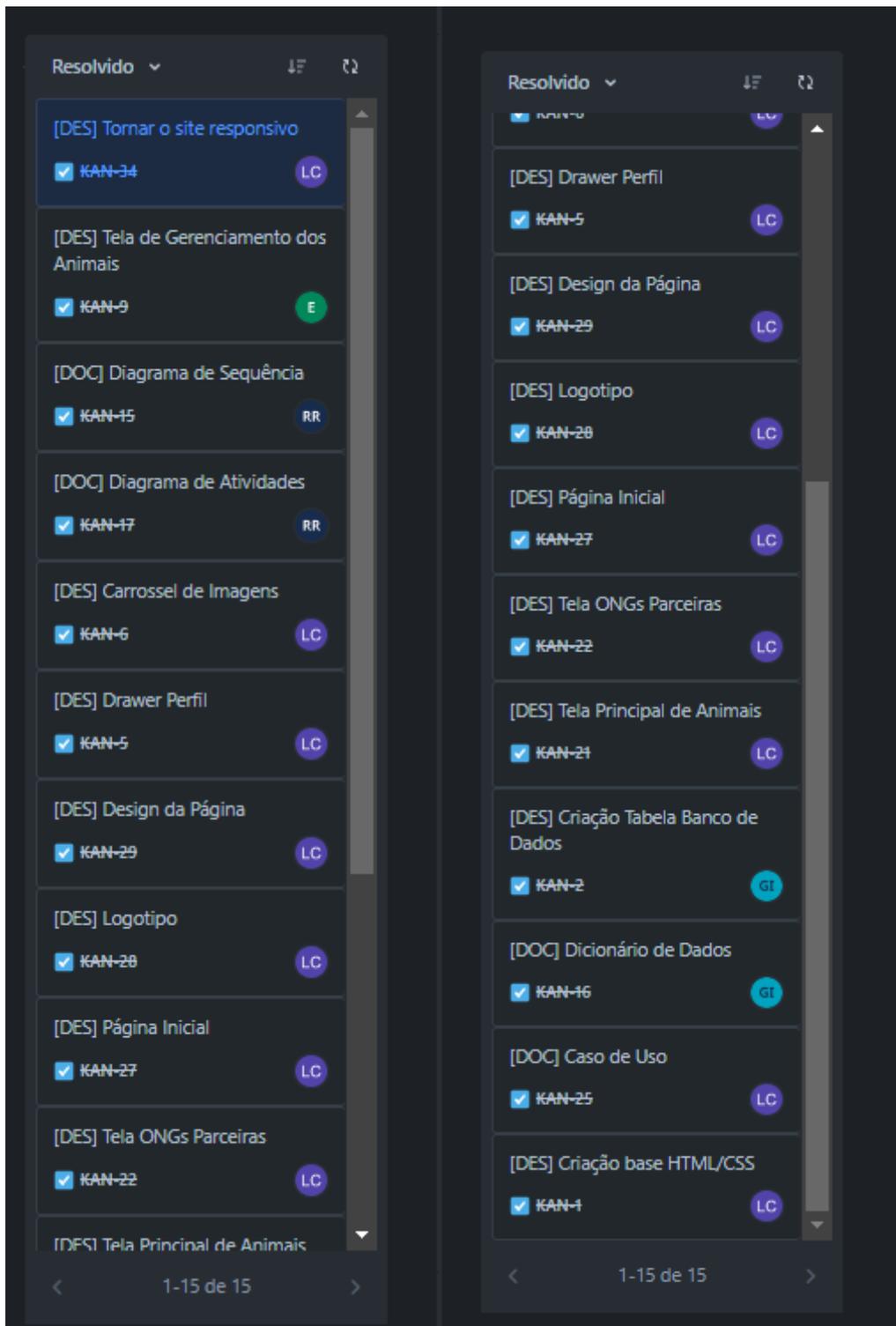
Elaborado pelo autor (2024)

Figura 13 – Continuação Cronograma Kanban (A fazer e Em Progresso)



Elaborado pelo autor (2024)

Figura 14 – Cronograma Kanban (Concluídas)



Elaborado pelo Autor (2024)

3.3 Ferramentas e Tecnologias

Ao escolher as tecnologias para desenvolver nossa aplicação web, consideramos diversos fatores, incluindo funcionalidade, desempenho e praticidade. Optamos por utilizar o HTML como linguagem padrão para estruturar nossos documentos na web, dada sua capacidade de definir a estrutura básica das páginas e permitir a inclusão de elementos multimídia, facilitando a criação de interfaces completas e acessíveis (DUCKET, 2011).

Além disso, incorporamos o CSS em nossos projetos para adicionar estilo e design às páginas web, aproveitando sua abordagem prática e visualmente rica, que nos permite desde os fundamentos básicos até técnicas avançadas de estilização responsiva (DUCKET, 2011). Também, para complementar, escolhemos o Google Fonts, uma plataforma colaborativa oferecida pelo Google, que disponibiliza uma ampla variedade de fontes e ícones gratuitos para uso em design e desenvolvimento web.

Para tornar nossas páginas mais interativas e dinâmicas, escolhemos o Javascript, uma linguagem de programação versátil e amplamente utilizada. Com o Javascript, conseguimos manipular o conteúdo, a estrutura e o estilo das páginas após o carregamento, oferecendo aos nossos usuários uma experiência altamente interativa e responsiva (LINDSEY, 2013).

Utilizamos o Jira da Atlassian como nossa plataforma principal para montar o quadro kanban. Optamos por essa ferramenta devido à sua reputação consolidada e robustez no gerenciamento de projetos de software (LAVERTY, 2017). Com isso, foi promovido a colaboração contínua e acompanhamento do progresso do trabalho em tempo real, visualizar e gerenciar nossas tarefas de forma eficiente, fornecendo uma melhor organização e produtividade dentro da equipe.

3.4 Estrutura XAMPP

A escolha do XAMPP como solução de desenvolvimento web se deu pela sua conveniência e praticidade. Com ele, conseguimos configurar rapidamente um ambiente de desenvolvimento web local, integrando todas as ferramentas necessárias, como Apache, MySQL e PHP, em um único pacote (KOCH, 2016).

3.4.1 Configuração do Apache

Integrar o Apache em nosso ambiente de hospedagem foi uma escolha natural devido à sua reputação como um servidor web de código aberto, altamente flexível e confiável. Com o Apache, podemos hospedar nossos sites e aplicativos web com facilidade, garantindo alta disponibilidade e desempenho consistente (PATTERSON, 2011).

3.4.2 Utilização do PHP e phpMyAdmin

O PHP (Hypertext Preprocessor) foi necessário por sua capacidade de processar formulários web, interagir com bancos de dados e gerar conteúdo dinâmico em tempo real, ideal para o que foi proposto no projeto. Além disso, é uma linguagem de fácil aprendizado, adequada para iniciantes e oferece uma sintaxe clara, intuitiva e possui compatibilidade com diferentes sistemas operacionais e servidores web. (SMITH, 2018).

Conforme destacado por Marc Delisle em "Mastering phpMyAdmin for Effective MySQL Management" (2016), o PHPMyAdmin é uma ferramenta de código aberto escrita em PHP, projetada para administrar bancos de dados MySQL. Foi utilizado para a administração de banco de dados, como criar e gerenciar tabelas, consultas SQL e usuários.

3.5 Estruturação do Banco de Dados

Decidimos utilizar o MySQL como nosso sistema de gerenciamento de banco de dados relacional pois, de acordo com o artigo 'MySQL: The World's Most Popular Open Source Database', publicado no site oficial do MySQL por Michael "Monty" Widenius e David Axmark (2003), o MySQL é reconhecido por sua confiabilidade, desempenho e facilidade de uso em uma ampla variedade de cenários de aplicação.

Com o MySQL, adotamos uma arquitetura cliente-servidor, na qual os clientes se conectam a um servidor MySQL para acessar e manipular os dados armazenados. Essa abordagem proporciona uma estrutura flexível que permite várias aplicações acessarem e compartilharem os dados de forma segura e eficiente. Uma das vantagens do MySQL é sua capacidade de suportar diversos mecanismos de armazenamento, como InnoDB, MyISAM e outros.

EXECUÇÃO DO PROJETO

4.1 Descrição do projeto

O projeto “Adocões” é uma plataforma online onde ONGs podem se cadastrar como administradores e compartilhar os perfis detalhados dos animais resgatados e prontos para adoção. Potenciais adotantes exploram os perfis dos animais, encontrando seu companheiro ideal e entrando em contato diretamente com as organizações para dar início ao processo de adoção. Com filtros intuitivos, a busca por um novo amigo peludo é simplificada, tornando a experiência de adoção mais acessível e gratificante para todos os envolvidos.

4.2 Principais Dificuldades Encontradas

Durante o desenvolvimento do projeto, enfrentamos diversas dificuldades. Uma delas foi compreender o fluxo das páginas, ou seja, estabelecer como os dados se relacionariam entre si. Além, disso, organizar o Banco de Dados de forma eficiente também se mostrou uma tarefa complexa, exigindo uma cuidadosa seleção de tabelas, chaves primárias e estratégias de normalização, foi necessário relembrar as aulas de BD para concluir esta etapa do projeto. Ademais, tornar o site responsivo e alinhar os elementos em diferentes dispositivos também foi complicado e demandou mais tempo do que o planejado, foi feito um estudo de como manipular os objetos de forma eficiente para obter os resultados planejados.

Em relação a documentação, os diagramas também carregaram uma bagagem de complexidade, pois no meio do desenvolvimento surgiram barreiras e dúvidas sobre como o sistema iria lidar com determinadas tarefas, por exemplo a validação das informações dos animais cadastrados pelas ONGs, questões de segurança e verificação de acessos.

4.3 Processo de Prototipagem

Prototipagem é uma maneira eficaz de tomar decisões de design. Ela fornece uma visão tangível do design para os desenvolvedores e para os usuários, e isso pode ajudar a evitar erros caros que são mais difíceis de corrigir quando o sistema está em desenvolvimento. (MCCONNELL, 1993).

A prototipagem é uma etapa crucial no processo de desenvolvimento de sistemas, permitindo que conceitos abstratos sejam transformados em

representações tangíveis e iterativas. Por meio da criação de versões simplificadas do produto final, conseguimos validar ideias e testar funcionalidades nas quais encontramos dificuldade para desenvolver, ajudando a identificar e corrigir tais problemas.

Abaixo encontra-se a apresentação do protótipo das telas do nosso sistema e suas funcionalidades.

4.3.1 Menu

A Figura 15 ilustra o Menu de Acesso destinado aos adotantes, com a única distinção em relação ao menu das ONGs sendo o perfil localizado à direita.

Figura 15 – Menu Adotantes



Elaborado pelo autor (2024)

Quando um usuário acessa o sistema como uma ONG, esse perfil é automaticamente substituído por uma opção intitulada "Administrar", conforme demonstrado na figura 16 abaixo. Essa diferenciação é fundamental para garantir que os adotantes e as ONGs tenham acesso às funcionalidades pertinentes às suas respectivas necessidades e responsabilidades dentro da plataforma.

Figura 16 – Menu ONGs



Elaborado pelo autor (2024)

Ambos os Menus são compostos por:

- **Logo e Página Inicial:** Sempre redireciona o usuário para a tela inicial
- **Adotar:** Redireciona o usuário para a tela que estão todos os animais disponíveis e com as opções de filtro.
- **Como Adotar:** Redireciona o usuário para a tela com explicações objetivas dos processos de adoção.
- **Porque Adotar:** Redireciona o usuário para a tela na qual incentiva e aborda os benefícios da adoção.
- **Ongs Parceiras:** Redireciona o usuário para a tela que está localizada todas as ONGs cadastradas no sistema, assim como suas informações.
- **Perfil:** Visível somente para adotantes, ao clicar aciona uma barra lateral com as informações pessoais do usuário (foto de perfil, contato etc.).
- **Administrar:** Visível somente para ONGs, redireciona o usuário para a tela de administração, onde ocorre o gerenciamento dos animais.

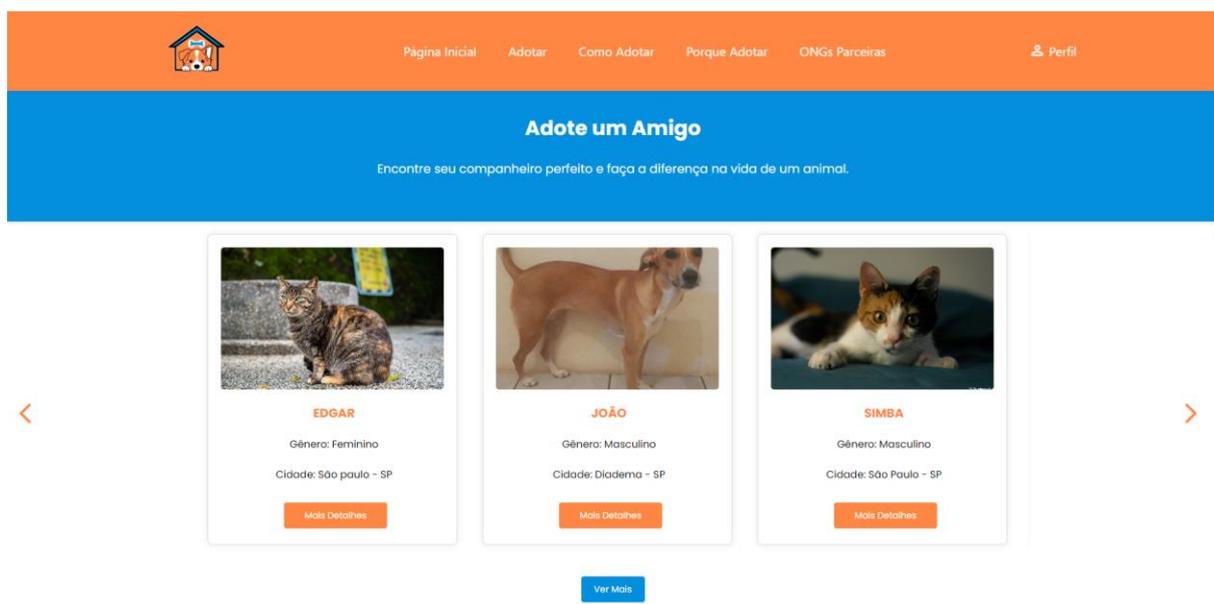
4.3.2 Página Inicial

A página inicial é estruturada em três seções distintas.

Carrossel de imagens

A primeira seção, intitulada "Adote um Amigo", apresenta um carrossel de imagens contendo alguns animais aleatórios. Essa abordagem visa despertar o interesse do usuário assim que a página é acessada. O botão "Ver mais" direciona o usuário para a tela "Adotar" para mais opções.

Figura 17 – Tela Carrossel de Imagens

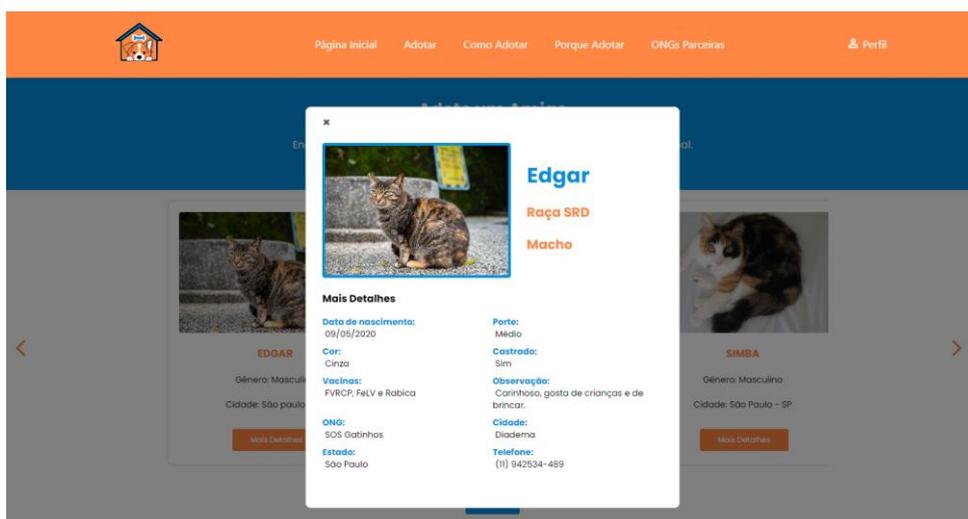


Elaborado pelo autor (2024)

Ao clicar no botão “mais detalhes” é aberto um *PopUp* com mais informações do animal selecionado, conforme apresentado na figura 18, com as seguintes informações a mais:

- Data de Nascimento
- Castrado Sim ou Não
- Tamanho do animal (pequeno, médio, grande).
- Cor predominante do animal
- ONG que pertence
- Raça do animal.
- Cidade na qual a ONG o abriga
- Vacinas: Indica se o animal recebeu alguma vacinação.
- Informações adicionais diversas.

Figura 18 – Popup Detalhes do Animal



Elaborado pelo autor (2024)

Como adotar

A segunda seção, chamada "Como Adotar", tem como objetivo fornecer explicações claras e objetivas sobre o processo de adoção. Nessa área da página inicial, os usuários encontrarão informações como taxas, restrições de idade para adotantes, quais documentos são necessários para o processo de adoção e as perguntas mais frequentes das entrevistas. Também é de suma importância destacar a importância da responsabilidade do adotante ao adotar um animal de estimação, por isso a seção é essencial para o tema do nosso sistema. A figura 19 ilustra como foi apresentado todas essas informações sem poluir a tela e de forma organizada.

Figura 19 – Tela Como Adotar

Como Adotar
Descubra como é fácil adotar seu novo amigo.

Idade e documentos

Animais de estimação só podem ser adotados por pessoas com idade igual ou superior a 21 anos. Mesmo assim, algumas instituições querem saber se os demais integrantes da família estão de acordo com a adoção do animal. Um conflito familiar pode resultar em mais um abandono ou maus tratos.

Para formalizar a adoção é necessário apresentar a cédula de identidade, CPF e o comprovante de residência. Com isso, a entidade controla os processos de adoção e, como muitas delas implantam microchips nos animais, fica mais fácil localizar o adotante caso o animal seja encontrado nas ruas novamente.



Responsabilidades e Entrevista

Outra exigência é a assinatura de um termo de responsabilidade, por meio do qual o adotante assume o compromisso de cuidar bem do animal. Ao assinar esse documento, o adotante está sujeito às penalidades legais, em caso de maus tratos e abandono do animal.

A entrevista faz parte do processo de adoção em algumas entidades. O objetivo é avaliar se a pessoa interessada realmente tem condições para cuidar do animal: alimentação, assistência veterinária, adestramento, proteção, entre outros cuidados.



Taxas

Algumas instituições cobram uma taxa de adoção para custear o microchip de identificação do animal e o Registro Geral de Animais. Por isso, antes de tomar a decisão, confirme aqui nossas organizações e entre em contato para mais informações.



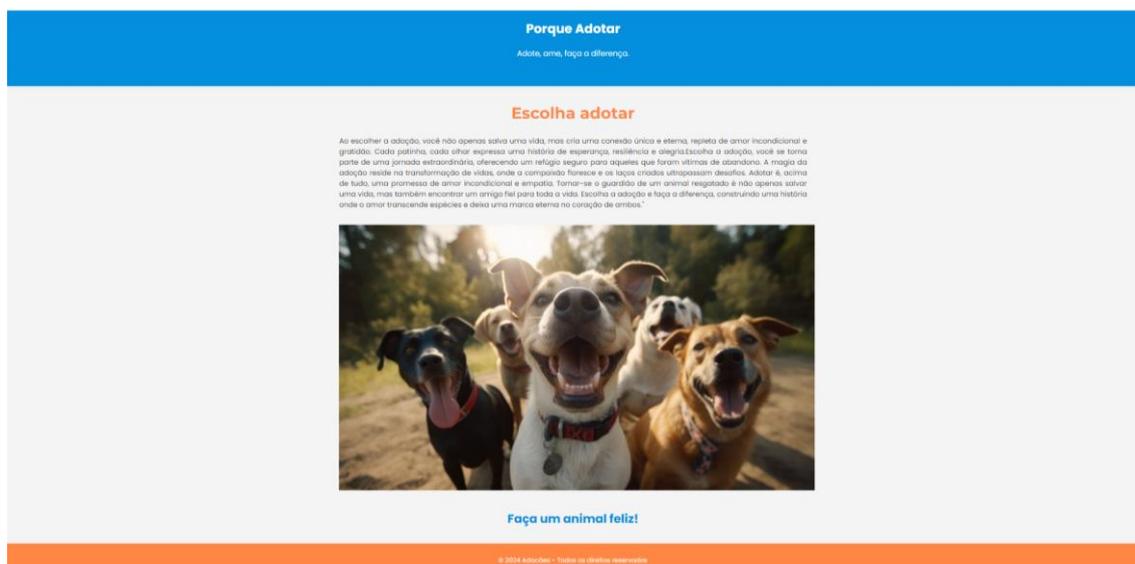
Elaborado pelo autor (2024)

Porque adotar

Por fim, a terceira seção, intitulada "Porque Adotar", destaca os benefícios da adoção e a importância desse ato. Nessa parte da página inicial, fornecemos informações detalhadas sobre as vantagens de adotar um animal de estimação,

destacando não apenas os benefícios para o animal, como também para o adotante e para a comunidade como um todo. Exploramos os aspectos emocionais, sociais e ambientais da adoção responsável, enfatizando como esse gesto pode transformar vidas e fazer a diferença no mundo. A frase no centro inferior “Faça um animal feliz!” Contém um link que ao ser clicado redireciona o usuário para a tela “Adotar”.

Figura 20 – Tela Porque Adotar

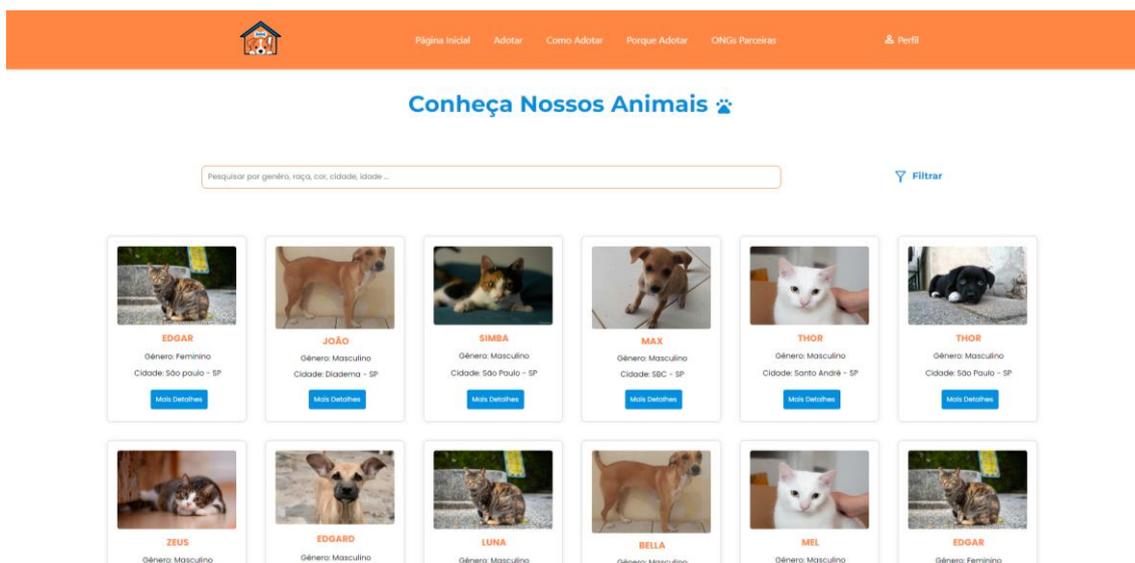


Elaborado pelo autor (2024)

4.3.3 Tela Adotar

A tela de adoção, conforme ilustrada na Figura 21, oferece uma visão abrangente de todos os animais disponíveis para adoção em formato de Cards. Nesta página, os usuários encontram dois filtros distintos para facilitar sua busca. O primeiro filtro, localizado na parte superior, é uma barra de pesquisa que permite aos usuários digitar características específicas, como cor, raça, tamanho, sexo, entre outros, para encontrar o animal desejado. Alternativamente, à direita da página, encontra-se um filtro de caixa de seleção que permite aos usuários filtrar os resultados com base no tipo de animal, seja gato ou cachorro. Ao clicar no botão “mais detalhes” é aberto um *PopUp* com mais informações detalhadas do animal selecionado, assim como a ONG, vacinas, raça etc (figura 17).

Figura 21 – Tela Adotar

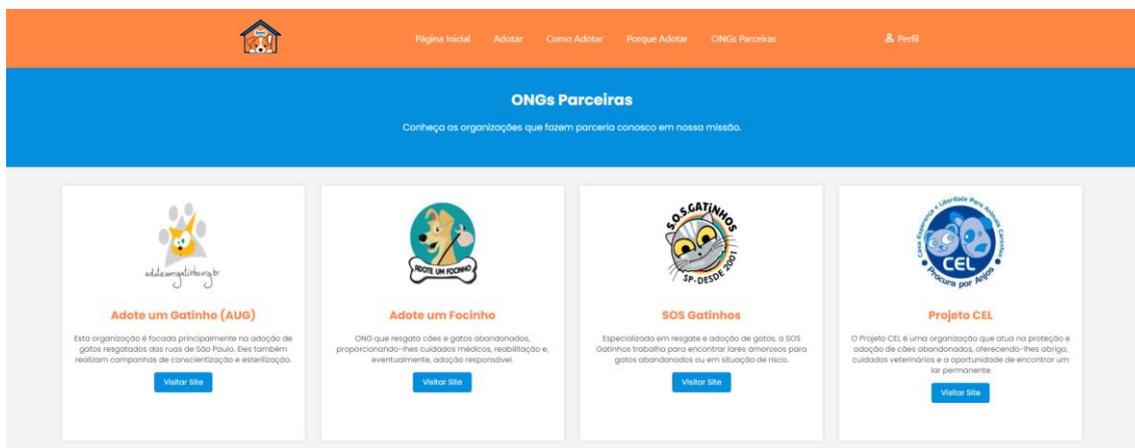


Elaborado pelo autor (2024)

4.3.4 Tela Ongs Parceiras

A tela de "ONGs Parceiras" exibe uma lista completa de todas as organizações não governamentais que colaboram com nosso site. Nessa página, os usuários encontrarão informações importantes de contato para cada ONG, como endereço de e-mail, número de telefone e endereço físico, quando disponíveis. Ao clicar no botão "Visitar Site" o usuário é redirecionado para o próprio site da organização, essa abordagem permite que os usuários acessem facilmente mais detalhes sobre o trabalho e os animais disponíveis para adoção de cada organização, incentivando a interação e o engajamento com as ONGs parceiras, além de proporcionar maior visibilidade para a causa.

Figura 22 – Tela ONGs Parceiras



Elaborado pelo autor (2024)

4.3.5 Perfil Adotantes

Quando os usuários clicam no perfil, a barra lateral, onde está localizado o perfil, é exibida, permitindo que eles visualizem e modifiquem facilmente suas informações de cadastro. A figura 23 apresenta os elementos que compõem o perfil do adotante. Exibe o nome completo do usuário, que geralmente é preenchido durante o processo de registro ou criação de conta e não pode ser alterado, assim como o CPF, que é necessário para fins de identificação e verificação de conta, além das informações básicas de contato como e-mail e telefone que podem ser alterados ao clicar no ícone de lápis. Uma alternativa de tornar a experiência do adotante melhor é permitir o upload da sua foto de perfil, oferecendo uma maneira de personalizar sua conta e complementar a personificação com mais informações no campo “sobre mim” (não obrigatório).

Figura 23 - Perfil

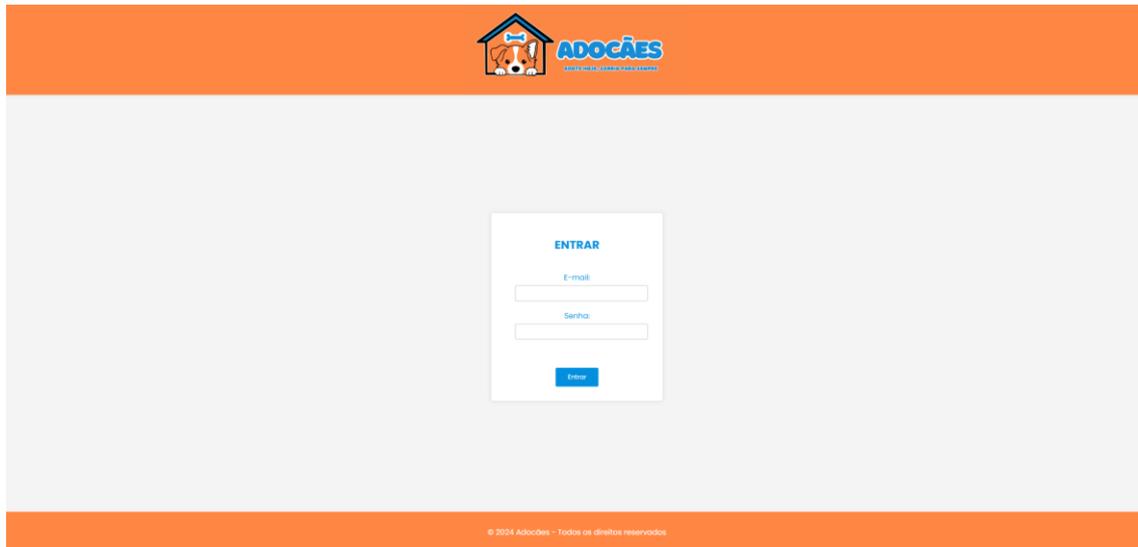


Elaborado pelo autor (2024)

4.3.6 Tela de Login

A tela de login é a entrada principal para acessar a plataforma e é aqui que ocorre o processo de verificação e validação. O sistema vai identificar o tipo de usuário (adotante ou ONG) e a partir disso habilitar ou desabilitar determinadas funções.

Figura 24 – Tela Login

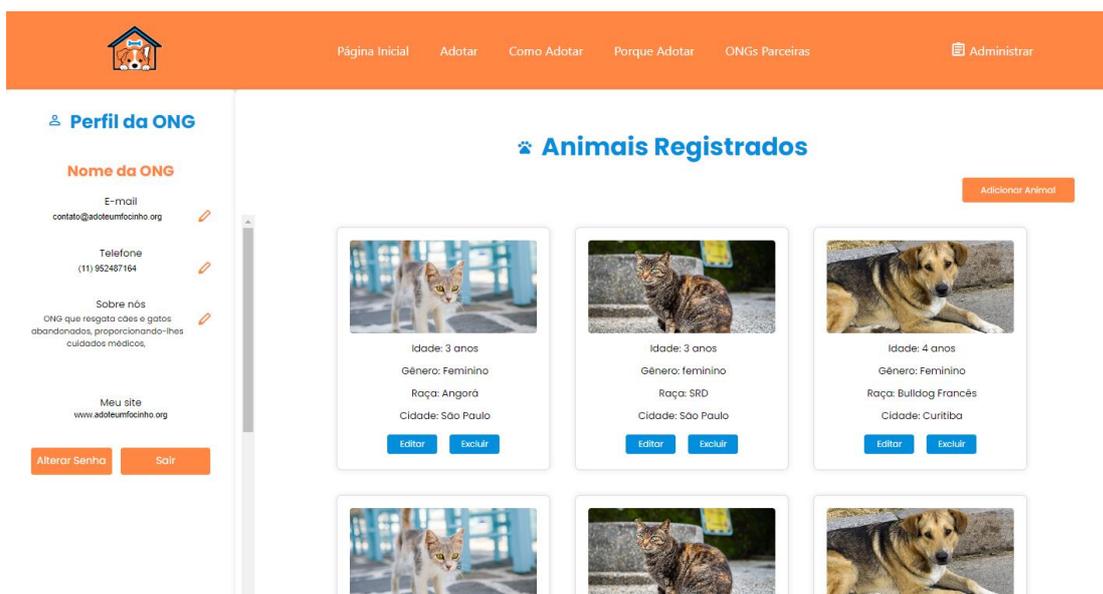


Elaborado pelo autor (2024)

4.3.7 Tela de Administração

A figura 25 apresenta o protótipo da tela de administração, que serve como o centro de controle da página de adoção de animais por ONGs.

Figura 25 – Tela de Administração



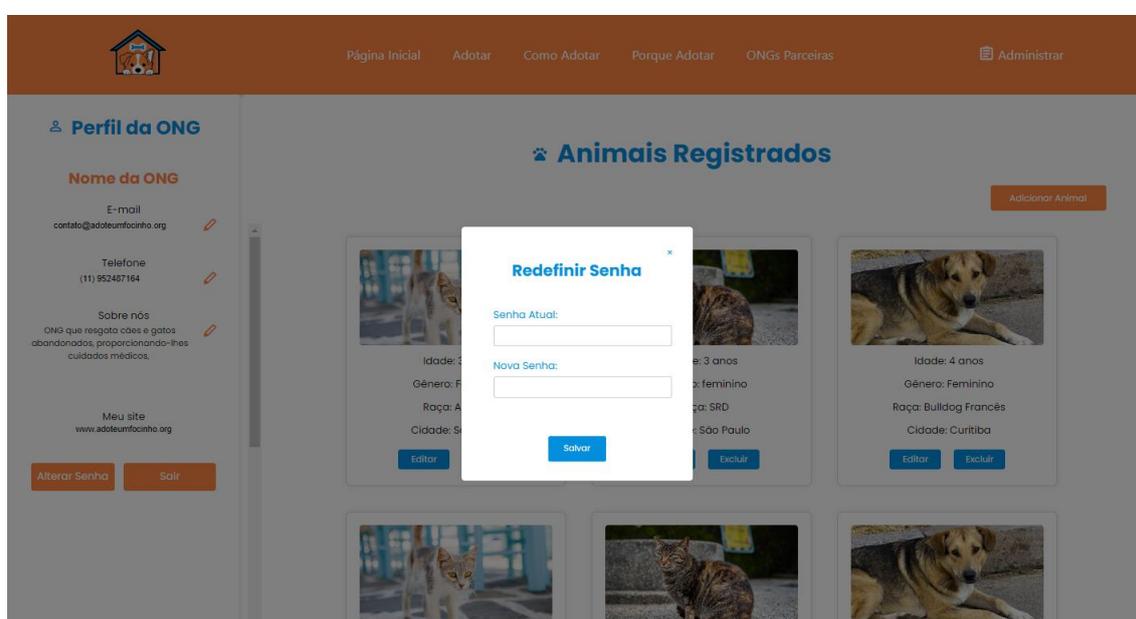
Elaborado pelo autor (2024)

Perfil ONGs

Na esquerda da página o perfil da ONG é exibido assim como suas informações de cadastro, os campos de e-mail, telefone e a descrição no campo "Sobre Nós" são editáveis, desta forma os administradores conseguem manter as informações atualizadas. Caso a organização tenha um site próprio, há um campo disponível para inserir o link no perfil.

O botão de "Alterar Senha" aciona um *PopUp* (Figura 26) que permite alterar a senha de acesso. Já o botão "Sair" realiza *logout* do sistema.

Figura 26 – *PopUp* Redefinir Senha



Elaborado pelo autor (2024)

Animais Registrados

Ao lado do perfil temos a área de animais registrados, que apresenta todos os animais vinculados à ONG. Em cada *card*, há um botão de "Excluir" usado para excluir o cadastro do animal da base de dados, caso ele for adotado. O botão "Editar" permite alterar as informações e mantê-las atualizadas, através do *PopUp* editar animais apresentado na figura 27.

Figura 27 – Popup Editar Animais

The image shows a web application interface with a navigation bar at the top containing links for 'Página Inicial', 'Adotar', 'Como Adotar', 'Porque Adotar', 'ONGs Parceiras', and 'Perfil'. Below the navigation bar, there is a main content area with a central pop-up form for editing animal information. The pop-up form features a photo of a dog and several input fields: 'Nome' (filled with 'Nikita'), 'Raça' (dropdown menu with 'Labrador'), 'Data de Nascimento' (calendar icon, filled with '03/03/2024'), 'Gênero' (dropdown menu with 'Macho'), 'Porte' (dropdown menu with 'Grande'), 'Cor' (dropdown menu with 'Caramelo'), 'Castrado' (dropdown menu with 'Sim'), 'Vacinas' (dropdown menu with 'Primeira dose V8'), 'Observação' (text area with 'Carinhoso e brincalhão'), and 'Tipo' (dropdown menu with 'Cachorro'). A blue 'Salvar' button is located at the bottom of the form. In the background, two animal profiles are visible: 'EDGAR' (a cat) and 'SIMBA' (a cat), each with a 'Mais Detalhes' button.

Elaborado pelo autor (2024)

No canto superior direito encontra-se o botão “Adicionar Animal” para adicionar novos gatos ou cachorros resgatados pela ONG que já possam ser adotados, o botão redireciona o usuário para o formulário de cadastro de animais, conforme apresentado na figura 28. Possui os seguintes campos:

- Nome: Nome do animal.
- Raça: Raça do animal.
- Data de Nascimento: Data de nascimento do animal ou aproximada.
- Castrado: Indicação se o animal é castrado ou não.
- Gênero: Sexo do animal (Macho ou Fêmea).
- Porte: Tamanho do animal (pequeno, médio, grande).
- Cor: A cor mais predominante da pelagem do animal.
- Tipo: Espécie do animal (gato ou cachorro).
- ONG: A ONG na qual o animal pertence.
- Vacinas: Indica se o animal recebeu alguma vacinação.
- Observação: Permite que a ONG inclua observações adicionais sobre o animal, como comportamento, preferências ou histórico médico.
- URL: Onde é inserida a imagem do animal, facilitando a visualização por parte dos adotantes interessados.

Figura 28 – Formulário Cadastrar Animal

The image shows a web form titled "CADASTRAR ANIMAL" on the "ADOCÕES" website. The form is set against a white background with an orange border. At the top, there is a logo for "ADOCÕES" featuring a dog's face and the tagline "ADOZE HOJE, SORRIA PARA SEMPRE". The form fields are as follows:

- Nome:** A text input field with the placeholder "Digite seu nome".
- Raça:** A text input field with the placeholder "Digite a raça".
- Data de nascimento:** A date picker field with the placeholder "dd/mm/aaaa".
- Castrado:** Radio buttons for "Sim" (selected) and "Não".
- Gênero:** Radio buttons for "Macho" (selected) and "Fêmea".
- Porte:** Radio buttons for "Pequeno" (selected), "Medio", and "Grande".
- Cor:** A text input field with the placeholder "Digite a cor do animal".
- Tipo:** A dropdown menu with the placeholder "Selecione o tipo:".
- Vacinas:** A text input field with the placeholder "Informe vacinas de seu animal".
- Observação:** A text input field with the placeholder "Digite observações do seu animal".
- Url:** A text input field with the placeholder "Digite a url da imagem do seu animal".

At the bottom of the form is a blue button labeled "Enviar". Below the form, there is a footer with the text "© 2024 Adocões - Todos os direitos reservados".

Elaborado pelo autor (2024)

4.3.8 Tela de Cadastro Adotante e ONG

Se o usuário acessar o sistema sem Login, o menu apresenta a opção de “Cadastre-se” na direita da página, conforme figura 29.

Figura 29 – Menu sem Login



Elaborado pelo autor (2024)

Cadastro Adotante

Assim que o usuário clica em “Cadastre-se” ele é redirecionado para a tela de cadastro de adotante, que também tem a opção de Login ou cadastrar-se como ONG, na parte inferior do formulário. É obrigatoriamente necessário preencher as seguintes informações no registro de adotante:

- Nome: Nome completo do adotante
- CPF: CPF do adotante
- E-mail: E-mail de contato do adotante
- Senha: A senha de preferência do adotante para acessar o sistema

- URL: Foto de perfil do adotante
- Telefone: Telefone de contato do adotante
- Endereço completo: CEP, rua, bairro, número, cidade e estado.

Figura 30 – Cadastro de Adotante Parte 1

Logo: **ADOCÇÕES** ADOTE HOJE. SOBRIA PARA SEMPRE.

CADASTRAR ADOTANTE

Nome:

CPF:

Email:

Senha:

URL:

Ja possui cadastro? clique aqui para fazer [Login](#) ou caso seja uma [ONG](#)

[Próximo](#)

© 2024 Adocções - Todos os direitos reservados

Elaborado pelo autor (2024)

Figura 31 – Cadastro de Adotante Parte 2

Logo: **ADOCÇÕES** ADOTE HOJE. SOBRIA PARA SEMPRE.

PREENCHA SEU ENDEREÇO

Telefone: Número:

Bairro:

Cidade:

Rua:

Estado:

CEP:

[Anterior](#) [Enviar](#)

© 2024 Adocções - Todos os direitos reservados

Elaborado pelo autor (2024)

Cadastro ONG

Para se cadastrar como ONG é necessário clicar no hiperlink “ong”, na parte inferior da página de cadastro, o usuário será redirecionado para o cadastro de ONGs e preencher as seguintes informações:

- Nome: Nome da ONG
- CNPJ: CNPJ da instituição para comprovar a veracidade da ONG
- Senha: A senha de preferência da ONG para acessar o sistema
- Site: Se a ONG possuir um site próprio, pode ser colocado neste campo, não é obrigatório
- URL: Logotipo da ONG
- Telefone: Telefone de contato para adoção de animais
- Endereço: Onde a ONG está localizada (CEP, rua, bairro, número, cidade e estado).

Figura 32 – Cadastro de ONG Parte 1



A imagem mostra a interface de usuário para o cadastro de uma ONG. No topo, há o logotipo 'ADOCÇÕES' com o slogan 'ADOPTAR HOJE, SOBREVIVER PARA SEMPRE'. O formulário principal, intitulado 'CADASTRAR ONG', possui os seguintes campos de entrada:

- Nome:** Digite o nome da ong
- CNPJ:** Digite seu cnpj
- Senha:** Digite sua senha
- Site:** Digite o site da sua ong
- URL:** Digite o url da sua imagem

Um botão azul 'Próximo' está localizado na base do formulário. Na parte inferior da página, há o rodapé: '© 2024 Adocções - Todos os direitos reservados'.

Elaborado pelo autor (2024)

Figura 33 – Cadastro de ONG Parte 2



A imagem mostra a interface de usuário para a segunda etapa do cadastro de uma ONG. No topo, há o logotipo 'ADOCÇÕES' com o slogan 'ADOPTAR HOJE, SOBREVIVER PARA SEMPRE'. O formulário principal, intitulado 'PREENCHA SEU ENDEREÇO', possui os seguintes campos de entrada:

- Telefone:** Digite seu telefone
- Número:** Número do seu endereço
- Bairro:** Digite seu bairro
- Cidade:** Digite o nome da sua cidade
- Rua:** Digite o nome da sua rua
- Estado:** Seleção de estado (Atualmente mostra 'Ace')
- CEP:** Digite o número do seu CEP

Dois botões azuis, 'Anterior' e 'Enviar', estão localizados na base do formulário. Na parte inferior da página, há o rodapé: '© 2024 Adocções - Todos os direitos reservados'.

Elaborado pelo autor (2024)

4.4 Fase de Teste

Caso de Teste	Nome do Teste	Descrição	Passos	Resultado Esperado	Resultado Obtido
CT1	Criar cadastro	Criar um perfil no sistema de ONG ou Adotante	1-Clicar no botão "Cadastrar ONG ou Adotante".	Caso todos os dados estejam digitados corretamente, o cadastro será realizado gerando uma mensagem de conclusão e confirmação de cadastro.	Realizado com sucesso
			2-Preencher os dados solicitados.		
			3-Clicar no botão "Cadastrar".		
CT2	Login no sistema	Realizar login no sistema utilizando credenciais válidas	1-Acessar a página de login	O usuário é autenticado e redirecionado para a página inicial	Realizado com sucesso
			2-Preencher os campos "E-mail" e "Senha".		
			3-Clicar no botão "Entrar".		
CT3	Adicionar novo animal	Adicionar um novo animal ao sistema	1-Navegar para a sessão "Administração"	O novo animal é adicionado ao sistema e uma mensagem de confirmação é exibida.	Realizado com sucesso
			2-Clicar no botão "Adicionar Animal"		
			3-Preencher os dados do animal		
			4-Clicar no botão "Salvar".		
CT4	Remover animal cadastrado	Remover um animal cadastrado do sistema.	1-Navegar para a sessão "Administração"	O animal é removido do sistema e uma mensagem de confirmação é exibida.	Realizado com sucesso
			2-Clicar no botão "Remover" no card do animal a ser removido.		
CT5	Editar perfil da ONG	Editar as informações do perfil de uma ONG cadastrada.	1- Navegar para a sessão "Administração"	As informações do perfil são atualizadas e uma mensagem de confirmação é exibida.	Realizado com sucesso
			2- Alterar as informações desejadas.		
			3-Clicar no botão "Salvar".		

CT6	Buscar animais para adoção	Realizar uma busca por animais disponíveis para adoção utilizando filtros.	1-Navegar para a sessão "Adotar"	O novo animal é adicionado ao sistema e uma mensagem de confirmação é exibida.	Realizado com sucesso
			2- Escrever no filtro de busca (espécie, porte, cor, etc.).		
			3- Clicar no botão "Buscar".		
CT7	Editar cadastro do animal	Editar as informações do animal após cadastro.	1-Navegar para a sessão "Administração"	O novo animal é adicionado ao sistema e uma mensagem de confirmação é exibida.	Realizado com sucesso
			2- Clicar no botão "Editar" no card do animal a ser editado.		
			3- Preencher os dados do animal.		
			4-Clicar no botão "Salvar".		

A fase de testes foi essencial para assegurar que o sistema funciona conforme os requisitos citados anteriormente. Os objetivos de verificação priorizaram verificar a conformidade, identificar possíveis defeitos, avaliar a usabilidade e testar o desempenho. O processo envolveu a execução das tarefas propostas pelo sistema, documentação de defeitos iniciais, correção e reteste, além de testes de regressão para confirmar que as correções não introduziram novos problemas.

CONSIDERAÇÕES FINAIS

O desenvolvimento do projeto 'Adocões' visou criar uma plataforma tecnológica que conectasse ONGs e possíveis adotantes de animais de estimação de forma mais eficiente. A motivação para este trabalho surgiu a partir da crescente demanda por adoção de animais e das dificuldades enfrentadas pelas ONGs em divulgar de maneira eficiente os animais disponíveis para adoção.

A aplicação web criada centraliza dados sobre animais disponíveis, simplificando a busca e a gestão dessas informações pelas ONGs. O uso da metodologia ágil Kanban permitiu uma organização eficiente das tarefas e uma maior flexibilidade para adaptações durante o desenvolvimento do projeto. Além disso, a aplicação de técnicas de Engenharia de Software, como a modelagem UML, a arquitetura MVC e o uso de um banco de dados robusto, garantiu a criação de um protótipo funcional e bem estruturado.

Os resultados obtidos com o protótipo foram promissores, mostrando que é possível centralizar e gerenciar informações sobre animais em situação de adoção de forma eficaz. Espera-se que, com a implementação completa do sistema, o número de adoções aumente significativamente, proporcionando um alívio para as ONGs e melhorando a qualidade de vida dos animais.

Em conclusão, o projeto demonstrou a viabilidade de uma plataforma digital para a divulgação de animais para adoção, e destaca a importância de soluções tecnológicas inovadoras no apoio a causas sociais. Futuras melhorias e expansões podem incluir a integração com redes sociais, funcionalidades adicionais para gestão de voluntários e doações, e a criação de aplicativos móveis para maior alcance e acessibilidade.

REFERÊNCIAS

ANDERSON, David J.. **Kanban: Mudança Evolucionária de Sucesso para Seu Negócio de Tecnologia**. 1. ed. Sequim, Washington: BLUE HOLE PRESS, 2011. p. 1-290.

ANDERSON. **Kaban essencial condensado**. 1. ed. [S.l.: s.n.], 2011. p. 1-280.

BOOK, Grady. **Best Of Booch: Designing Strategies for Object Technology**. 1. ed. Kingdom: ED Eykholt, 1998. p. 1-213.

DUCKET. **HTML and CSS: Design and Build Websites**. 1. ed. Indianapolis: [s.n.], 2011. p. 1-200.

FIGMA. **Protótipo**. Disponível em: <https://www.figma.com/proto/9XmnRsryTtL375PVDJn7HB/ADOC%C3%83ES-PROTOTIPO?node-id=0-68&starting-point-node-id=0%3A68>. Acesso em: 20 set. 2023.

FIREBASE UMA ABORDAGEM PRATICA . **Firestore uma abordagem pratica**. Disponível em: https://www.google.com.br/books/edition/Angular_11_e_Firebase/cRi9DwAAQB-AJ?hl=pt-BR&gbpv=1&dq=Firestore:+Uma+abordagem+prática+por+Jimmy+Porto:&printsec=frontcover. Acesso em: 26 nov. 2023.

G1.GLOBO.COM. **Brasil tem mais de 170 mil animais abandonados sob cuidados de ONGs, aponta instituto**. Disponível em: <https://g1.globo.com/sp/sao-paulo/noticia/2019/08/18/brasil-tem-mais-de-170-mil-animais-abandonados-sob-cuidado-de-ongs-aponta-instituto.ghtml>. Acesso em: 25 out. 2023.

GAMMA, Erich. **Padrões de projeto Soluções Reutilizáveis : Padrões de Projeto**. 1. ed. [S.l.]: bookman, 2000. p. 1-353.

GUDWIN, Ricardo R.. **Engenharia de Software: Uma nova Versão**. 2. ed. São Paulo: Unicamp, 2015. p. 1-155.

GUDWIN, Ricardo R.. **Engenharia de software: Uma visão prática**. 2. ed. [S.l.: s.n.], 2015. p. 1-148.

GUEDES, G. T. A. **UML 2: Uma Abordagem Prática**. 3. ed. São Paulo: Novatec, 2009. p. 1-27.

HEUSER, Carlos Alberto. **Projeto de banco de dados: Projeto banco de dados**. 6. ed. São Paulo: Artmed, 2009. p. 1-229.

INSTITUTO PET BRASIL. **Censo pet.** Disponível em: <https://institutopetbrasil.com/imprensa/censo-pet-1393-milhoes-de-animais-de-estimacao-no-brasil/>. Acesso em: 25 out. 2023.

INSTITUTO PET BRASIL. **Numero de animais de estimação em situação de vulnerabilidade mais do que dobra em dois anos, aponta pesquisa do IPB.** Disponível em: <https://institutopetbrasil.com/fique-por-dentro/numero-de-animais-de-estimacao-em-situacao-de-vulnerabilidade-mais-do-que-dobra-em-dois-anos-aponta-pesquisa-do-ipb/>. Acesso em: 25 out. 2023.

KOCH; S., K.; STEFFEN, & **XAMPP**: Introduction Guide. . 1. ed. [S.l.: s.n.], 2016. p. 1-200.

LAVERTY; R.. **JIRA Software Essentials**: JIRA Software Essentials. 1. ed. [S.l.: s.n.], 2017. p. 1-20.

LEWAGON. **O que é padrao mvc.** Disponível em: <https://blog.lewagon.com/pt-br/skills/o-que-e-padrao-mvc/>. Acesso em: 22 dez. 2023.

LINDLEY; CODY.. **DOM Enlightenment**: .DOM Enlightenment. 1. ed. [S.l.: s.n.], 2013. p. 1-20.

LUCIAPP. **Luciapp.** Disponível em: https://lucid.app/lucidchart/00bf0b17-e89b-40ac-8ad4-522795ff69bb/edit?viewport_loc=-41944%2C-6675%2C2994%2C1466%2C0_0&invitationId=inv_a8330054-8ab5-4ff6-a8c4-3acc146a1e1e. Acesso em: 26 dez. 2023.

NETO, Roque Maitino; GALINDO, Rodrigo; JÚNIOR, Márioghio. **Engenharia de software**: engenharia de software. 1. ed. Londrina: eGTB, 2016. p. 1-220.

OMG. **OMG Unified Modeling Language** : OML, UML. 2. ed. [S.l.: s.n.], 2007. p. 1-738.

PATTERSON *et al.* **Apache**: :The Definitive Guide.. 1. ed. [S.l.: s.n.], 2011. p. 1-30.

PRESSMAN; ROGER. **Analise de requisitos**: analise de requisitos. 8. ed. São Paulo: McGraw-Hill, 2011. p. 1-201.

REVISTA CASA JARDIM. **Cresce o numero de pets adotados no Brasil durante a pandemia.** Disponível em: <https://revistacasaejardim.globo.com/Casa-e-Jardim/Dicas/Pets/noticia/2021/10/cresce-o-numero-de-pets-adotados-no-brasil-durante-pandemia.html>. Acesso em: 25 out. 2023.

TRIMER, R. *et al.* **Software cengineering**: engenharia de software. 9. ed. Brasil: Pearson, 2011. p. 1-544.