

CENTRO UNIVERSITÁRIO OPET – UNIOPET

GABRIEL CESAR GONÇALVES RODRIGUES

JÚLIA ISABEL ZANINI

PROJETO INTEGRADOR – PI

AUTOMATIZAÇÃO DE CONTROLE DE ANIMAIS DEPENDENTES

CURITIBA

2023

GABRIEL CESAR GONÇALVES RODRIGUES - gabriel.rodrigues4@uniopet.edu.br

JÚLIA ISABEL ZANINI - juliazanini1@uniopet.edu.br

PROJETO INTEGRADOR: AUTOMATIZAÇÃO DE CONTROLE DE ANIMAIS DEPENDENTES

Projeto interdisciplinar, apresentado para o curso
de Análise e Desenvolvimento de Sistemas

Orientador: Antonio Lima

O

CURITIBA

2023

Sumário

1. Introdução	4
1.1. Especificações	4
Objetivos	4
1.2. Objetivos gerais	4
1.3. Objetivos específicos	5
2. Justificativa	5
3. Escopo	5
4. Fluxo de Processo de Negócio	6
5. Descrição geral	6
5.1. Restrições	6
6. Considerações finais	7
ATORES DO SISTEMA	8
8. Descritivo de caso de uso	8
8.1. REALIZAR LOGIN	8
8.2. VERIFICAR NOTIFICAÇÕES	8
8.3. ADICIONAR ANIMAIS	8
	9
9. Redes	9
9.1. Descrição do projeto de redes	9
10. Fundamentos de TI	10
10.1. A tecnologia no setor empresarial	10
10.2. Características requisitadas para o bom funcionamento do software	10
11. Banco de dados	10
11.1 Apresentação das tabelas	11
12. Solução Técnica	13
13. Divisão de projetos em GIT.	13
14. Referências	14

1. Introdução

A falta de controle de vacinas, remédios e alimentação prejudica a saúde dos animais que, independentemente da espécie, precisam de cuidados. Tanto pet, quanto os animais da área rural, não são fáceis de administrar com a correria do dia a dia, principalmente quando são em grande quantidade.

1.1. Especificações

Com esse sistema será possível administrar a saúde e alimentação dos animais dependentes do criador.

O usuário do sistema poderá cadastrar seus animais e de acordo com a espécie do animal, o próprio sistema irá sugerir uma lista (editável) de cuidados.

Objetivos

1.2. Objetivos gerais

Temos por objetivo desenvolver um sistema de controle não apenas para o público doméstico, mas também para o agronegócio. Neste sistema ofertamos ferramentas para o bem-estar do animal.

1.3. Objetivos específicos

A presente proposta tem como objetivo apresentar um sistema de controle de animais com uma interface dinâmica e intuitiva, para que o usuário consiga dominar e interagir com facilidade.

Os recursos oferecidos incluem notificações em forma de lembretes que apontam alimentar o animal, cuidar da higiene, vacinação e remédios de acordo com a necessidade de cada animal, além de mostrar a localização de clínicas veterinárias, aviários e Pet Shops mais próximos.

2. Justificativa

Considerando as atividades frenéticas do ser humano nos tempos atuais, ocorrem com frequência esquecimentos de rotinas relacionadas

aos seus animais de estimação, bem como outros animais de criação comercial sob seus cuidados.

O sistema proposto visa exatamente sanar essa possível falha, trazendo um bem-estar maior aos animais, dando suporte para que suas necessidades sejam atendidas.

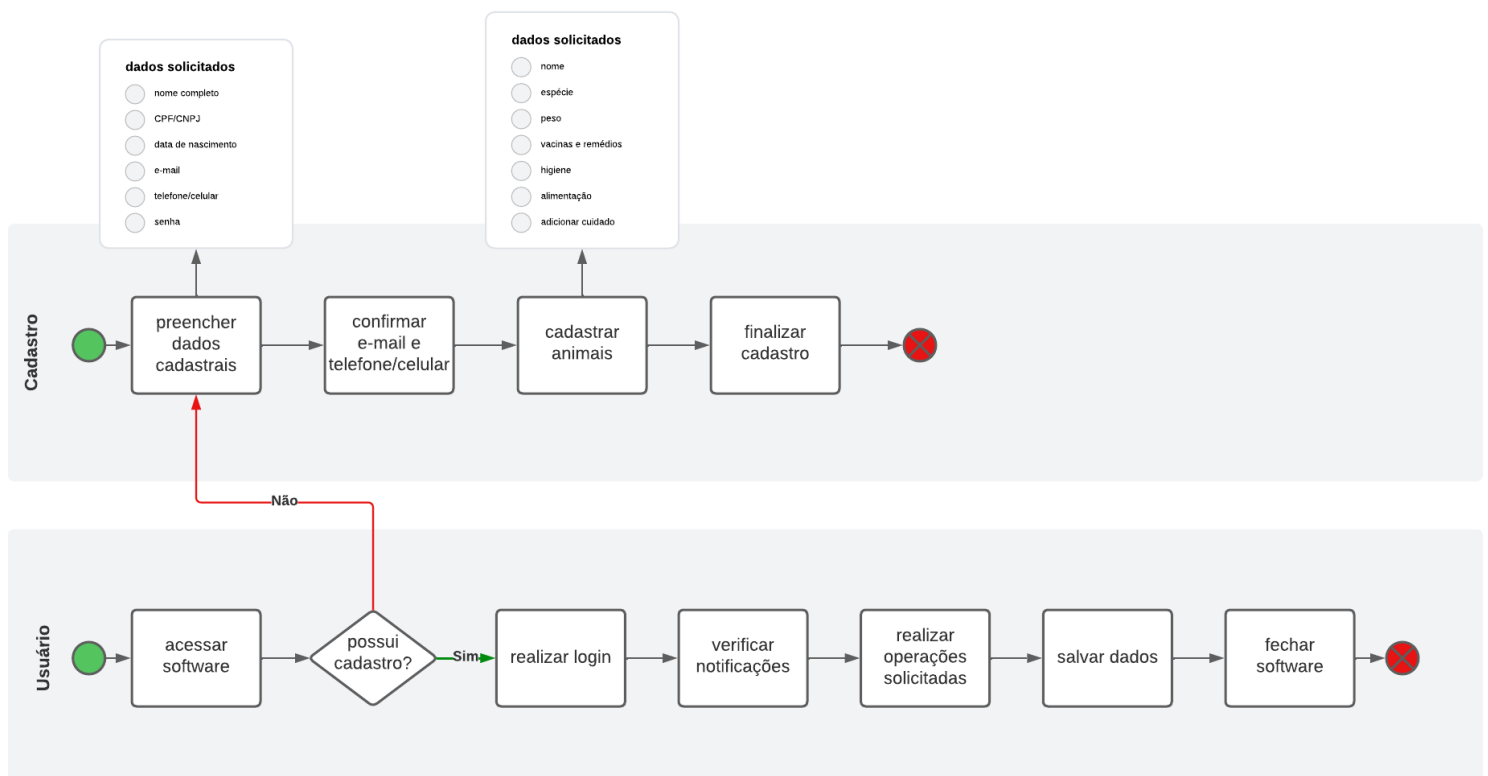
3. Escopo

A plataforma desenvolvida visa fazer a interação com o usuário apresentando a ele as atividades a serem realizadas no calendário, promovendo uma organização e um cuidado maior com seus animais, de maneira que o responsável pelo cuidado e trato possa acompanhar melhor as questões básicas como alimentação, saúde e higiene dos mesmos.

Para ter acesso ao perfil de cada animal, o usuário fará um login que será liberado após o fornecimento de seu e-mail e senha. A partir de então terá acesso ao perfil de cada animal e ao painel de notificações, onde será apresentado cada necessidade do animal correspondente.

No painel de notificações, que será exibido a partir de um clique no ícone de notificações, aparecerá todas as necessidades e procedimentos a serem feitos a cada animal. No perfil do animal será possível preencher nome, espécie, idade, tipo de alimentação, rotinas de higiene, saúde e outras observações e dados que julgar pertinente.

4. Fluxo de Processo de Negócio



5. Descrição geral

O Sistema é um meio de acompanhar a administração e cuidados dos animais, que visa melhorar a saúde e bem-estar dos mesmos. O sistema será intuitivo, facilitando a usabilidade do usuário.

5.1. Restrições

Não haverá banco de dados para armazenamento de informações no momento, pois todas as telas são apenas protótipos.

6. Considerações finais

Levando em conta a rotina frenética do ser humano nos tempos atuais, ocorrem esquecimentos de rotinas relacionadas aos seus animais de estimação, bem como outros animais de criação comercial sob seus cuidados.

O sistema proposto visa exatamente sanar essa possível falha, trazendo um bem-estar maior aos animais, dando suporte para que suas necessidades sejam atendidas.

7. Requisitos

Nome	Descrição	Prioridade
[RF.001] efetuar login.	Espaço utilizado para acessar o sistema.	Alta.
[RF.002] alterar senha.	Menu destinado à alteração de senha.	Média.
[RF.003] recuperar senha.	Ferramenta para recuperação de senha. O usuário entrará com a matrícula e o CPF.	Média.
[RF.004] adicionar animais.	Referente ao menu “adicionar animal”: link destinado a página .	Alta.
[RF.005] adicionar animal.	Referente ao menu “adicionar animal”: o botão “adicionar” envia o usuário ao menu de “dados do animal”.	Alta.

[RF.006] Anular o animal	Referente a página de “Meus animais”: o botão “anular” executa o cancelamento do animal.	Alta.
[RF.007] Solicitar notificações	Referente a página de “notificações”: o botão “notificar” ativa o processo de envio de notificações	Alta.
[RF.008] Localizar postos para animais	Referente a página “Mapa”: o botão “procurar” faz a procura de estabelecimentos relacionados a animais	Alta.
[RF.009] . Trocar senha.	Referente ao menu “Meu perfil”: esta opção remete o usuário ao campo de alteração de senha.	Média.
[RF.010] . Sair.	Referente ao menu a opção “sair” faz logout do usuário no sistema.	Média.

REQUISITOS NÃO FUNCIONAIS

Nome	Descrição	Prioridade
[RNF.001] Sistema operacional.	Windows 7 ou superior.	Alta.
[RNF.002] Rede.	O sistema necessita de uma conexão estável à internet.	Alta.
[RNF.003] Implementação.	O sistema será desenvolvido em HTML.	Alta.

ATORES DO SISTEMA

Ator	Responsabilidade
Usuário.	Realizar login; adicionar animais; conferir notificações;
Programador/Suporte.	Realizar login e alimentar dados no sistema.
Sistema.	Realizará o gerenciamento dos dados informados e automatizará os carregamentos.

8. Descritivo de caso de uso

8.1. REALIZAR LOGIN

Nome do UC 001:	Realizar login.
Ator:	Usuário
Pré-requisitos:	Possuir login e senha.
Cenário principal:	a) O usuário deverá utilizar login e senha para acessar o sistema.
Cenário alternativo:	a) Caso o usuário tenha esquecido a senha deverá recadastrá-la clicando no botão “Esqueceu sua senha?”

Caso de testes:	a) O sistema irá verificar se os dados estão corretos para realização do login.
------------------------	---

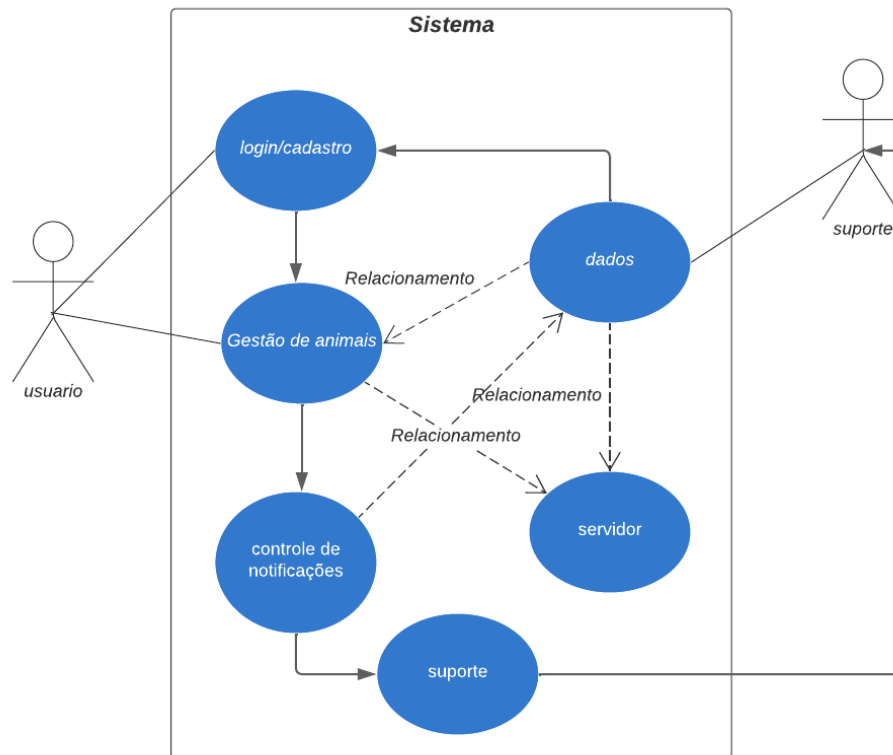
8.2. VERIFICAR NOTIFICAÇÕES

Nome do UC 002:	Verificar notificações.
Ator:	usuário
Pré-requisitos:	a) Estar logado no sistema.
Cenário principal:	a) O sistema solicita login e senha; b) Com login realizado irá abrir a página “Notificações”.
Cenário alternativo:	a) Caso o usuário não encontre o animal nas notificações, se direcionar ao programador/suporte;
Caso de testes:	a) O sistema irá mostrar as notificações caso o login seja realizado corretamente.

8.3. ADICIONAR ANIMAIS

Nome do UC 003:	Conferir rota.
Ator:	Usuário
Pré-requisitos:	a) Estar logado no sistema.
Cenário principal:	a) O usuário deverá utilizar login e senha para o acesso; b) Após o login o usuário adiciona o animal desejado e preenche as informações c) Clicar em “concluir” para adicionar o animal
Caso de testes:	a) O sistema deverá abrir um mapa no “Google Maps”, quando a função “mapa” for acionada, clicando em “mapa” no menu indicando a localização exata, assim que o usuário clicar no respectivo link.

Data	Descrição	Autor(es)
09/09/2022	Início do desenvolvimento da documentação.	Gabriel, Júlia
16/09/2022	Descrição dos requisitos funcionais, caso de uso descritivo e diagrama de casos de uso.	Gabriel, Júlia
31/06/2023	Finalização do documento.	Gabriel, Júlia
23/06/2023	Revisão final do documento.	Gabriel, Júlia



9. Redes

9.1. Descrição do projeto de redes

Usaremos uma lista para permitir que apenas 1 dos ip's da rede possa acessar o servidor de banco de dados, este ip será do profissional responsável pelo suporte e manutenção do mesmo, e por isso deverá ser o único que pode ter acesso ao banco de dados.

10. Fundamentos de TI

10.1. A tecnologia no setor empresarial

Armazenaríamos os dados dos usuários em nuvem pela praticidade de acesso aos dados caso necessário. Assinaríamos o Google Drive Enterprise, que nos permite armazenamento ilimitado com mais alguns recursos avançados de suporte por U\$25,00 ao mês.

A arquitetura pensada para o armazenamento dos dados dos animais seria que os dados dos mesmos fossem divididos em classes-matrizes, por exemplo, mamíferos, insetos, reptéis e etc., e então teriam mais subgrupos como canídeo, equídeo e etc. até chegar na espécie exata do animal.

Precisaríamos de um profissional em banco de dados (média salarial de 4 à 7 mil reais), de um profissional para front-end (média salarial de 3 à 6 mil reais) e de um profissional para back-end (média salarial de 4 à 8 mil reais).

10.2. Características requisitadas para o bom funcionamento do software

PC/notebook

Sistema operacional

- Windows 10 ou superior.

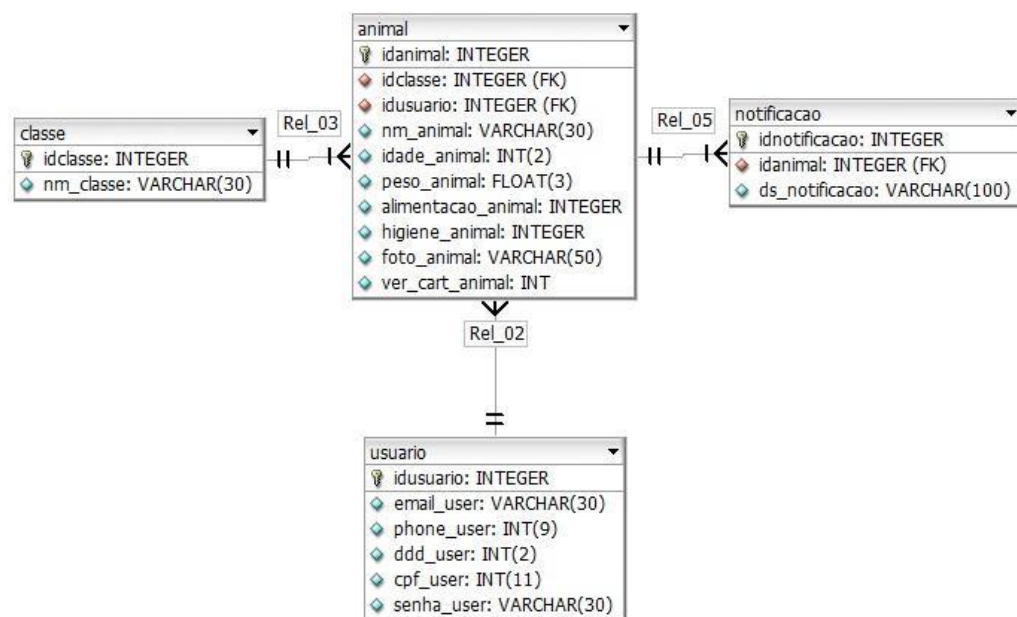
Processador

- Intel Pentium G4560 Dual-core (Desktop).
- Intel Core i3-1115G4 Dual-core (Notebook).

Memoria

- 4gb de memória ram.

11. Banco de dados



11.1 Apresentação das tabelas

Tabela "animais":

A tabela "animal" contém informações sobre os animais.

Possui as seguintes vozes:

"idanimal": um número de identificação único para cada animal, que é gerado automaticamente.

"idclasse": um número inteiro não nulo que representa a classe do animal.

"idusuario": um número inteiro não nulo que representa o usuário associado ao animal.

"nm_animal": uma string de até 30 caracteres que representa o nome do animal. Pode ser nulo.

"idade_animal": um número inteiro de até 2 dígitos que representa a idade do animal. Não pode ser nulo.

"peso_animal": um número de ponto flutuante com até 3 dígitos que representa o peso do animal. Não pode ser nulo.

"alimentacao_animal": um número inteiro não nulo que representa o tipo de alimentação do animal.

"higiene_animal": um número inteiro não nulo que representa o nível de higiene do animal.

"foto_animal": uma string de até 50 caracteres que armazena o caminho da foto do animal. Pode ser nulo.

"ver_cart_animal": um número inteiro que indica se o animal possui um cartão de vacinação. Pode ser nulo.

Possui a chave primária "idanimal" e dois índices: "animal_FKIndex1" referente à coluna "idusuario" e "animal_FKIndex2" referente à coluna "idclasse".

Tabela "aula":

A tabela "classe" armazena informações sobre as classes dos animais.

Possui as seguintes vozes:

"idclasse": um número de identificação único para cada classe, que é gerado automaticamente.

"nm_classe": uma string de até 30 caracteres que representa o nome da classe. Pode ser nulo.

Possui a chave primária "idclasse".

Tabela "notificação":

A tabela "notificação" contém informações sobre as notificações relacionadas aos animais.

Possui as seguintes vozes:

"idnotificação": um número de identificação único para cada notificação, que é gerado automaticamente.

"idanimal": um número inteiro não nulo que representa o animal associado à notificação.

"ds_notificacao": uma string de até 100 caracteres que representa a descrição da notificação. Pode ser nulo.

Possui a chave primária "idnotificação" e um índice "notificação_FKIndex1" referente à coluna "idanimal".

Tabela "usuário":

A tabela "usuário" armazena informações sobre os usuários.

Possui as seguintes vozes:

"idusuario": um número de identificação único para cada usuário, que é gerado automaticamente.

"email_user": uma string de até 30 caracteres que representa o email do usuário. Pode ser nulo.

"phone_user": um número inteiro de até 9 dígitos que representa o telefone do usuário. Pode ser nulo.

"ddd_user": um número inteiro de até 2 dígitos que representa o DDD do telefone do usuário. Pode ser nulo.

"cpf

"senha_user": uma string de até 30 caracteres que representa a senha do usuário. Pode ser nulo.

Possui a chave primária "idusuario".

12. Solução Técnica

12.1. PROGRAMAS UTILIZADOS

Para a realização da etapa 2 do projeto foram utilizados os seguintes programas na parte de desenvolvimento WEB:

- Software Visual Studio Code (HTML, CSS, BOOTSTRAP, PHP);
- XAMPP

Na modelagem do banco de dados:

- Db Designer;
- SQL Server e Maria Db (phpMyAdmin)

13. Divisão de projetos em GIT.

O APP está disponível no GitHub, na pasta principal chamada adms, está dividido em modelo MVC.

Modelo: Sua responsabilidade é gerenciar e controlar como os dados se comportam por meio de funções, lógica e regras de negócios estabelecidas. O guardião dos dados recebe informações do responsável pelo tratamento, válidas sejam elas corretas ou não, e envia a resposta mais adequada. No nosso caso, todas as telas iniciais são modeladas para que a lógica de negócios não seja perdida.

Controller: A camada de controle é responsável por mediar as requisições enviadas pela view com as respostas fornecidas pelo model, processando os dados informados pelo usuário e repassando-os para outras camadas.

Visão: Esta camada é responsável por apresentar visualmente as informações ao usuário. Outros contêm funções e ícones de modelo e banco de dados para exibir aplicativos. Aplicações práticas da abordagem que o GitHub usa:

Ramificação: para que possamos realizar testes e modificações sem alterar a linha do tempo natural.

Commit: carrega o diretório no GitHub, mantendo as alterações feitas.

Pull request: permite ver a diferença entre o projeto original e o branch do recurso.

<https://projetointegradorjg.github.io/projetointegrador/>

14. Referências

EXEMPLO DE PROJETO DE PI. **Disponibilizados pelo orientador.**

BOOTSTRAP. **Disponível em:** <https://getbootstrap.com/>.

CISCO PACKET TRACER. **Disponível em:** <https://www.netacad.com/>.

GITHUB. **Disponível em:** <https://github.com/>.

LUCIDCHART. **Disponível em:** <https://lucid.app/>.