

MINISTÉRIO DA EDUCAÇÃO
SECRETARIA DE EDUCAÇÃO PROFISSIONAL E TECNOLÓGICA
INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA DA PARAÍBA
CAMPUS GUARABIRA

RELATÓRIO TÉCNICO
Projeto Integrador em Sistemas para Internet (PISI)

Integrantes

Marcella da Costa Gomes
Juliana de Fatima Bandeira Barbosa

Orientador

Prof. Dr. João Ricardo Freire de Melo

Co-orientadora

Prof^a. Dr^a. Gabriela Guedes de Souza

MINISTÉRIO DA EDUCAÇÃO
SECRETARIA DE EDUCAÇÃO PROFISSIONAL E TECNOLÓGICA
INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA DA PARAÍBA
CAMPUS GUARABIRA

RELATÓRIO TÉCNICO

Projeto Integrador em Sistemas para Internet (PISI)

Marcella da Costa Gomes
Juliana de Fatima Bandeira Barbosa

Orientador

Prof. Dr. João Ricardo Freire de Melo

Co-orientadora

Prof^a. Dr^a. Gabriela Guedes de Souza

Relatório técnico apresentado à Coordenação do Curso Superior de Tecnologia em Sistemas para Internet, no âmbito do IFPB - Campus Guarabira, para a conclusão da disciplina de Projeto Integrador em Sistemas para Internet (PISI), em cumprimento às exigências do Projeto Pedagógico do Curso.

Guarabira/PB
2023.1

FICHA CATALOGRÁFICA ELABORADA PELA BIBLIOTECA DO IFPB - GUARABIRA

G633r Gomes, Marcella da Costa
Relatório Técnico: Projeto Integrador em Sistemas para Internet (PISI).
Marcella da Costa Gomes; Juliana de Fatima Bandeira Barbosa.- Guarabira,
2023.
24 f.: il; color.

Trabalho de Conclusão de Curso (Superior de Tecnologia em
Sistemas para Internet). – Instituto Federal da Paraíba, Campus
Guarabira, 2023.

"Orientação: Prof. Dr. João Ricardo Freire de Melo"
"Co-orientadora: Profa. Dra. Gabriela Guedes de Souza."

Referências.

1. Sistemas para Internet. 2. Sistemas de Gerenciamento. 3. Página Web.
4. Hospedagem de Animais. I. Título.

CDU 004.4(0.067)



SERVIÇO PÚBLICO FEDERAL
INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA DA PARAÍBA
COORDENAÇÃO DO CURSO SUPERIOR DE TECNOLOGIA EM SISTEMAS PARA
INTERNET - CAMPUS GUARABIRA



ATA 11/2023 - CCSTSI/DDE/DG/GB/REITORIA/IFPB

ATA DE DEFESA
CST Sistemas para Internet

Aos 07 de julho de 2023, às 09:30, em sala virtual, reuniram-se João Ricardo Freire de Melo (orientador), Gabriela Guedes de Souza (co-orientadora), Dornelles Moisés Alves Nonato (membro da banca), Vandilson Bezerra de Lima (membro da banca), para avaliarem a apresentação do Relatório Final da Disciplina de Projeto Integrador em Sistemas para Internet (PISI) da aluna **MARCELLA DA COSTA GOMES**, intitulado "Projeto de Implementação de Produto de Software: PetConfy", protocolado para apresentação de acordo com os requisitos expostos no Projeto Pedagógico de Curso de Tecnologia em Sistemas para Internet. Após a apresentação, a banca apresentou, por unanimidade, pareceres a favor da aprovação do trabalho. Desta forma, o Relatório Final foi **aprovado** e definiu-se a **nota final 95 (noventa e cinco)**. Cabe salientar, como expõe o Projeto Pedagógico de Curso de Tecnologia em Sistemas para Internet, que o aluno do curso pode optar por utilizar o Relatório Final da Disciplina de Projeto Integrador em Sistemas para Internet (PISI) em substituição ao Trabalho de Conclusão de Curso.

Nada mais havendo a tratar, às 10:00, encerraram-se os trabalhos, determinando a lavratura desta ata, que, após lida e considerada conforme, será assinada pelos presentes. Eu, João Ricardo Freire de Melo, lavrei a presente ata.

Guarabira/PB, em 13 de julho de 2023.

Documento assinado eletronicamente por:

- **Joao Ricardo Freire de Melo**, PROFESSOR ENS BASICO TECN TECNOLOGICO, em 13/07/2023 11:21:41.
- **Gabriela Guedes de Souza**, PROFESSOR ENS BASICO TECN TECNOLOGICO, em 13/07/2023 12:59:55.
- **Dornelles Moises Alves Nonato**, PROF ENS BAS TEC TECNOLOGICO-SUBSTITUTO, em 13/07/2023 13:51:11.
- **Vandilson Bezerra de Lima**, PROF ENS BAS TEC TECNOLOGICO-SUBSTITUTO, em 14/07/2023 09:19:22.

Este documento foi emitido pelo SUAP em 06/07/2023. Para comprovar sua autenticidade, faça a leitura do QRCode ao lado ou acesse <https://suap.ifpb.edu.br/autenticar-documento/> e forneça os dados abaixo:

Código 446286
Verificador: a70d56346c
Código de Autenticação:



“O conhecimento é uma riqueza que não pode ser roubada; ele crescerá quando você tentar doá-lo.”

Debasish Mridha

RESUMO

O Projeto Integrador em Sistemas para Internet pretende desenvolver a capacidade de aplicação dos conceitos e teorias adquiridas durante o curso de forma integrada. Dessa maneira, o presente relatório é consubstanciado em duas partes: a documentação de um produto implementado na forma de software (parte I) e as percepções acadêmicas dos autores sobre os variados tópicos e áreas presentes no curso superior em Sistemas para Internet (parte II). O projeto de implementação se refere ao desenvolvimento de um software, página web e/ou aplicativo híbrido com o objetivo de auxiliar donos de animais domésticos na hospedagem e proteção com os mesmos. Por se tratar de projeto de implementação meramente com fins educacionais, a comercialização/venda do programa de computador oriundo desta implementação - transferência de tecnologia - não é cabível neste trabalho. Por sua vez, as percepções sobre a integração de tópicos e áreas presentes no curso superior em Sistemas para Internet são descritas neste documento.

Palavras-chave: Pet sitting; Cuidado animal; Software

DADOS BÁSICOS	6
PARTE I - Documentação de um Produto de Software	7
I.1 Introdução ao Documento	7
I.1.1 Tema	7
I.1.2 Objetivo do Projeto	7
I.1.3 Delimitação do Problema	8
I.1.4 Justificativa da Escolha do Tema	8
I.1.5 Método de Trabalho	8
I.1.6 Organização do Trabalho	9
I.1.7 Sobre os objetivos de desenvolvimento sustentável	9
I.1.8 Sobre a Lei Geral da Proteção de Dados	9
I.2 Descrição Geral do Sistema	10
I.2.1 Descrição do Problema	10
I.3 Requisitos do Sistemas	11
I.3.1 Perfis de Usuário	11
I.3.2 Requisitos Funcionais	11
I.3.3 Requisitos Não-funcionais	12
I.4 Análise e Design	13
I.4.1 Diagramas de Classe	13
I.4.2 Diagramas de Atividades	15
I.4.3 Modelo de Dados	16
I.4.4 Ambiente de Desenvolvimento	17
I.5 Implementação	17
I.6. Conclusão	19
REFERÊNCIAS	21
ANEXOS	22
ANEXO A - Cronograma de Acompanhamento das Atividades	22

DADOS BÁSICOS

Natureza do Projeto Integrador em Sistemas para Internet (PISI)¹

() Produção Acadêmica

(X) Projeto de Implementação

IDENTIFICAÇÃO

Produto de Software: PetConfy

Integrantes

Marcella da Costa Gomes

Juliana de Fatima Bandeira Barbosa

Orientador

Prof. Dr. João Ricardo Freire de Melo

Co-orientadora

Prof^a. Dr^a. Gabriela Guedes de Souza

Membro Interno

Prof. Dornelles Moisés Alves Nonato

Membro Interno

Prof. Vandilson Bezerra de Lima

Disciplina Vinculada

Projeto Integrador em Sistemas para Internet (PISI) - 67h

Período de Construção

Calendário Acadêmico: 2023.1

Calendário Civil: fevereiro/2023 a julho/2023

Data da Apresentação

07/07/2023

Banca de Avaliação

Prof. João Ricardo Freire de Melo

Prof. George Candeia de Sousa Medeiros

Prof. Dornelles Moisés Alves Nonato

Prof. Vandilson Bezerra de Lima

¹ Segundo o Projeto Pedagógico de Curso - Tecnologia em Sistemas para Internet (IFPB, 2019), o PISI pode ser originado por:

I. **Produção Acadêmica**, resultante de pesquisa científica sobre um determinado objeto, ato, fato ou fenômeno da realidade ou da produção técnica ou tecnológica, visando a aplicabilidade nos diversos campos do saber, com atendimento aos padrões técnicos de intervenção. A produção acadêmica expressa as competências e as habilidades desenvolvidas (ou os conhecimentos adquiridos) pelos estudantes durante o período de formação.

II. **Projeto de Implementação**, em sentido lato, no qual se busca encontrar uma resposta prática para um problema técnico-profissional, tecnológico ou técnico-científico.

PARTE I - Documentação de um Produto de Software

I.1 Introdução ao Documento

O objetivo deste capítulo é apresentar o projeto. Para tal, será apresentado o tema, objetivo do projeto, delimitação do problema, justificativa da escolha do tema, método de trabalho e organização do trabalho.

I.1.1 Tema

"PetConfy" busca sanar o problema de falta de disponibilidade para a indispensável assistência animal, mais especificamente animais domésticos de pequeno porte. Toda a aplicação é centrada exclusivamente na prestação de serviços de hospedagem animal, tendo em vista que muitos dos donos ficam sem opções para salvar seus bichos de estimação em situações corriqueiras ou específicas. Dessa forma, com a escolha de um ConfyAmigo, a área, e fornecimento de agendamento com o mesmo, a segurança e o bem estar dos pequenos companheiros estarão garantidas durante todo o tempo. Além disso, toda a informação básica será fornecida pela plataforma antes da contratação e das observações específicas de cada dono para seu(s) respectivo(s) animal(ais).

I.1.2 Objetivo do Projeto

A aplicação "PetConfy" tem como propósito auxiliar donos de animais domésticos na hospedagem e proteção com os mesmos, buscando prestar serviços em tempo integral a partir do gerenciamento de ajudantes listados na plataforma.

Tem-se como principais objetivos do "PetConfy":

1. Auxiliar donos de animais: com o registro do(s) animal(ais) de estimação, o "PetConfy" busca oferecer serviços específicos para suprir quaisquer necessidades no que tange a estalagem e recreação dos mesmos;
2. Administrar ajudantes: na figura do "ConfyAmigo", os inscritos na plataforma podem cadastrar sua localidade, disponibilidade, e quando existente, suas experiências com hospedaria animal.
3. Desburocratização de acordos: a plataforma oferece simplicidade e praticidade para firmar compromissos, agilizando processos de maneira direta e segura;

4. Serviços específicos: para toda e qualquer necessidade, o "PetConfy" trará soluções desde: hospedaria diurna, noturna, visitas em horários específicos e limitados, além de estalagem por um tempo maior.

Em suma, a aplicação "PetConfy", a partir de demandas características, poderá fornecer atendimento voltado para cada situação. A gestão de clientes prestadores de serviço para clientes donos de animais de estimação vem a ser otimizada, evitando transtornos/burocracias.

I.1.3 Delimitação do Problema

Donos de animais, por muitas vezes, acabam se deparando com dificuldades comuns da rotina, como uma viagem, trabalho e etc. Ou seja, são ocasiões em que os donos precisarão se ausentar por determinado tempo. Este tipo de situação pode se tornar um problema, tendo em vista a escassez de locais próprios para a instalação de um animal.

A concentração do problema é a dificuldade em encontrar um ambiente que seja apropriado para o alojamento de animais, em que os donos possam os deixar em segurança e conforto.

I.1.4 Justificativa da Escolha do Tema

A justificativa da escolha do tema "Petconfy" como sistema de hospedagem para animais se dá à necessidade da criação de um ambiente que esteja qualificado para resguardar os mesmos. Atualmente, em localidades mais distantes dos grandes centros não é muito comum que existam ambientes caseiros, ou até mesmo centrados, na causa de instalação animal. A escassez de um serviço como esse acaba por gerar mais dificuldades no dia a dia dos donos de animais que, eventualmente, possam precisar se ausentar.

O sistema "Petconfy" tem como objetivo sanar esses problemas, entregando uma plataforma que gerencia serviços para de hospedagem para animais. Com o sistema será possível que os usuários se cadastrem, tanto para prestar o serviço de acolhimento para os animais, como para solicitar o serviço de acolhimento.

Em suma, a escolha de "Petconfy" como tema, justifica-se pela necessidade da criação de ambientes focados no acolhimento animal, oferecendo segurança e conforto.

I.1.5 Método de Trabalho

Visando agilidade e prezando organização, a metodologia de trabalho adotada para o desenvolvimento da referida aplicação é o SCRUM. Eficiência, flexibilidade e colaboração são algumas das características da metodologia, que tem sua estrutura o funcionamento a partir de ciclos e interações de trabalho. A abordagem prática e evolutiva do SCRUM possibilita que o produto seja entregue ao cliente em estado funcional, a partir de necessidades discutidas em momento anterior. Levando em consideração sprints quinzenais de pouca duração, adotam-se as cerimônias de Sprint Planning Meeting, Daily SCRUM e Sprint Review; alguns artefatos como Product Backlog, Sprint Backlog também estão presentes. (SBROCCO; MACEDO, 2012, p.160-169)

A modelagem consiste na orientação a objetos, “uma linguagem visual utilizada para modelar softwares.” (GUEDES, 2011, p. 19) Partindo do paradigma, de acordo com Guedes em UML 2: uma abordagem prática, qualquer objeto no mundo real pode ser abstraído, sendo dessa forma, apreendido por classificações. (GUEDES, 2011, p. 43-44)

I.1.6 Organização do Trabalho

A organização do trabalho consiste nos pontos de descrição geral do sistema, constituindo a descrição do problema, perfis de usuário e requisitos funcionais e não funcionais; o ponto de análise e design, constituindo diagrama de classes, diagrama de atividades e modelo de dados; a implementação do sistema, e conclusão.

I.1.7 Sobre os objetivos do desenvolvimento sustentável

O desenvolvimento da ferramenta produzida está relacionado com os Objetivos de Desenvolvimento Sustentável (ODS) e suas metas. Especificamente, verifica-se interfaces e sinergias com o 3 (Assegurar uma vida saudável e promover o bem-estar para todos, em todas as idades), 8 (Promover o crescimento econômico sustentado, inclusivo e sustentável, emprego pleno e produtivo e trabalho decente para todas e todos) e 10 (Reduzir a desigualdade dentro dos países e entre eles).

I.1.8 Sobre a Lei Geral da Proteção de Dados

O objetivo deste projeto é implementar o sistema PetConfy de forma a garantir sua conformidade com a Lei Geral de Proteção de Dados (LGPD) e assegurar a privacidade dos

dados pessoais dos usuários por meio da implementação das funcionalidades relacionadas ao tratamento de dados pessoais dos clientes, colaboradores e fornecedores. Esses dados pessoais incluem, mas não se limitam a: nome, endereço, CPF/CNPJ, e-mail, telefone, dados bancários e informações sobre compras e transações.

A base legal para o tratamento dos dados pessoais será obtida por meio do consentimento dos titulares, sendo este obtido por meio de opção clara e inequívoca no momento do cadastro no sistema. Os titulares dos dados podem exercer seus direitos garantidos pela LGPD, como o direito de acesso, retificação, exclusão e portabilidade dos dados pessoais. Os dados pessoais são retidos apenas pelo tempo necessário para a finalidade específica para a qual foram coletados, conforme previsto na política de retenção de dados.

Será elaborada uma política de privacidade e um aviso de privacidade claros e acessíveis aos usuários do sistema. Esses documentos explicarão de forma transparente como os dados pessoais serão coletados, utilizados, compartilhados e protegidos. O sistema incluirá um mecanismo de obtenção e registro do consentimento dos titulares para o tratamento dos dados pessoais, assim como a possibilidade de revogação do consentimento a qualquer momento.

Serão implementadas medidas de segurança técnica e organizacional para proteger os dados pessoais dos usuários, incluindo criptografia de dados, controle de acesso, monitoramento de atividades e backups regulares. Será estabelecido um procedimento para detecção, relato e notificação de violações de dados, em conformidade com a legislação vigente. Da mesma forma, serão realizadas auditorias regulares para verificar a conformidade do sistema PetConfy com as políticas de privacidade e com a LGPD. Monitoramentos contínuos também serão implementados para identificar potenciais vulnerabilidades. Ficam designadas, Juliana de Fátima Bandeira Barbosa e Marcella da Costa Gomes, como encarregadas de Proteção de Dados (DPO) para supervisionar as atividades relacionadas à proteção de dados e assegurar a conformidade com a LGPD.

I.2 Descrição Geral do Sistema

O objetivo deste capítulo é a descrição geral do sistema, escopo e funções principais.

I.2.1 Descrição do Problema

Muitos donos de animais passam por dificuldades quando o assunto é a hospedagem dos mesmos. Atualmente, não é tão comum a questão de ambientes que ofertem esse tipo de serviço, o que acaba se tornando, algumas vezes, um problema para quem possui animais de

estimação e precise se ausentar por um período maior em algum momento. Resumidamente, o principal destaque do problema é a solução da ausência de alojamentos para hospedagem de animais.

I.3 Requisitos do Sistema

I.3.1 Perfis de Usuário

- Perfil 01: Prestador de serviços - O usuário faz moderado uso da web, usando a plataforma para cadastrar sua disponibilidade de espaço e horário para acomodar determinada quantidade de animais. Este usuário tem conhecimento regular da web, acessando diariamente a plataforma para ficar a par de solicitações;
- Perfil 02: Usuário dono do animal - O usuário faz moderado uso da web; este contratante de quaisquer modalidades de pet sitting cadastrará o(s) animal(is), data, horário e melhor local para aloca-lo junto ao contratado. Este usuário tem conhecimento regular da web, e acessa periodicamente e/ou diariamente a plataforma, a depender de suas presentes necessidades;
- Perfil 03: Administrador do sistema - O administrador terá controle da aplicação, podendo visualizar, editar ou deletar perfis de usuários, além de manter a plataforma atualizada. Este usuário tem alto conhecimento da web, e acessa diariamente e frequentemente o sistema.

I.3.2 Requisitos funcionais

Requisitos referentes às funcionalidades da aplicação.

- **RF_01:** Cadastro de usuário - O usuário preencherá o formulário de cadastro contendo nome, e-mail, endereço e senha;
- **RF_02:** Login de usuário - Após o cadastro, ambos os usuários poderão acessar seus painéis correspondentes;
- **RF_03:** Cadastro de animal - O sistema deve permitir que o usuário realize o cadastro do animal informando nome, idade, espécie e/ou observações adicionais;
- **RF_04:** Painel do usuário – O sistema deve permitir o acesso ao painel relativo ao usuário;

- **RF_05:** Painel do administrador – Após login, o sistema deverá exibir painel do administrador contendo as opções relativas;
- **RF_06:** Aba agenda - O sistema deve permitir a visualização da agenda do usuário;
- **RF_07:** Aba transações – O sistema deverá permitir visualização do(s) serviço(s) selecionado(s);
- **RF_08:** Opção visualizar do administrador - A opção deve permitir o administrador de visualizar a lista dos usuários registrados assim que acessa o painel;
- **RF_09:** Opção editar para administrador - A opção deve permitir o administrador fazer edições nos perfis de usuários registrados;
- **RF_10:** Opção deletar do administrador - A opção deve permitir o administrador de deletar perfis de usuários registrados;
- **RF_11:** Opção de serviço diaLegal – O sistema deverá disponibilizar o serviço diaLegal;
- **RF_12:** Opção de serviço boaNoiteCompanheiro - O sistema deverá disponibilizar o serviço boaNoiteCompanheiro;
- **RF_13:** Opção de serviço aBordo - O sistema deverá disponibilizar o serviço aBordo;
- **RF_14:** Opção de serviço passeioNoParque - O sistema deverá disponibilizar o serviço passeioNoParque;
- **RF_15:** Opção atualizar pet do usuário – O painel do usuário deverá exibir um botão com a função para atualizar o(s) pet(s) cadastrado(s) do usuário;
- **RF_16:** Opção visualizar pets do usuário - O painel do usuário deverá exibir um botão com a função para visualizar o(s) pet(s) cadastrado(s) do usuário;
- **RF_17:** Opção deletar pets do usuário -O painel do usuário deverá exibir um botão com a função para deletar o(s) pet(s) cadastrados do usuário.

I.3.3 Requisitos Não-funcionais

Os requisitos não-funcionais apresentados são:

- **RNF_01:** O sistema deverá se manter disponível por pelo menos 99% do tempo;
- **RNF_02:** O sistema deverá ser desenvolvido na linguagem Python com framework Flask;
- **RNF_03:** a aplicação deverá funcionar em qualquer plataforma web;

- **RNF_04:** a aplicação deverá se relacionar com o banco de dados Postgresql.

I.4 Análise e Design

Este terceiro capítulo, a partir dos requisitos listados no capítulo anterior (I.2), visa esmiuçar toda a estrutura do sistema.

I.4.1 Diagrama de Classe

Gilleanes Guedes explica este diagrama:

Serve de apoio para a maioria dos demais diagramas. Como o próprio nome diz, define a estrutura das classes utilizadas pelo sistema, determinando os atributos e métodos que cada classe tem, além de estabelecer como as classes se relacionam e trocam informações entre si. (GUEDES, 2011, p. 31)

Em suma, o diagrama de classes contém a abstração do real no sistema. Como exposto em seu nome, suas classes contêm atributos (características) e métodos (operações) que irão solucionar alguma situação específica de forma integrada. Será possível visualizar o diagrama de classes da aplicação PetConfy na figura 1, logo após a listagem das suas respectivas classes:

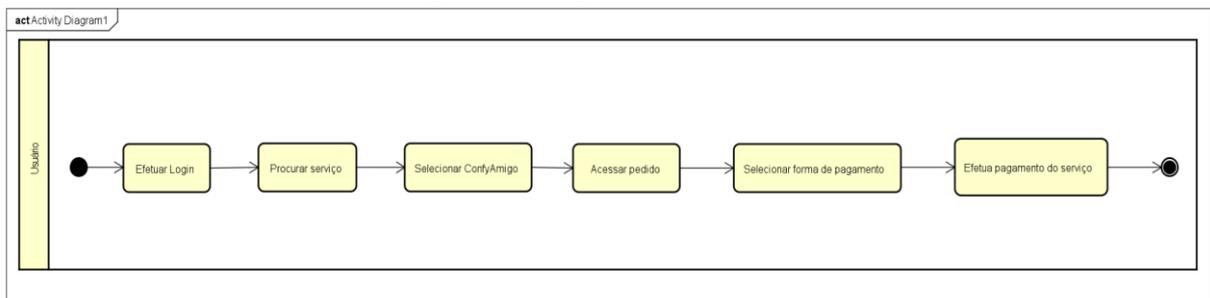
- Usuário: Atributos - nome; e-mail; endereço; senha; observacoesAdicionais; Controller - index, home, register, login, logout, perfil, agenda;
- Administrador: Atributos - admEmail; admSenha;
- Endereço: Atributos - estado; cidade; rua; numero;
- Serviço: Atributos - horarioInicial; horarioFinal; diaInicial; diaFinal; telefone, user_id; submit; Controller - passeioparque
- Pet: Atributos - nomePet; idade; espécie; observações; user_id; Controller - cadastroPet; getPet; deletePet; updatePet;
- Espécie: Pets: Atributos - espécie; raça;
- Pedido: Atributos - data;
- Pagamento: Atributos - valor;
- Boletim: Atributos - nome; código;
- Crédito: Atributos - bandeira; dataValid.

I.4.2 Diagrama de Atividades

Martin Fowler (2007, p. 118) explica que “os diagramas de atividades são uma técnica para descrever lógica de procedimento, processo de negócio e fluxo de trabalho.” O autor ainda relaciona os diagramas de atividades a fluxogramas, destacando que as suas diferenças residem no “comportamento paralelo” que este diagrama suporta, pontuando também regras a serem seguidas. (FOWLER, 2007, p. 118)

Em outras palavras, é um passo-a-passo de como realizar determinada atividade; no caso específico para PetConfy, a atividade principal da aplicação. Para esta representação, o diagrama de atividades foca no usuário que quer contratar o serviço de *pet sitting* selecionando algum ConfyAmigo cadastrado em um ou mais dos serviços disponíveis na plataforma. Todos os serviços disponíveis para cadastro detêm as mesmas regras ou regras semelhantes.

Figura 2 – Diagrama de atividades



Fonte: Elaborado pelo autor.

I.4.3 Modelo de Dados

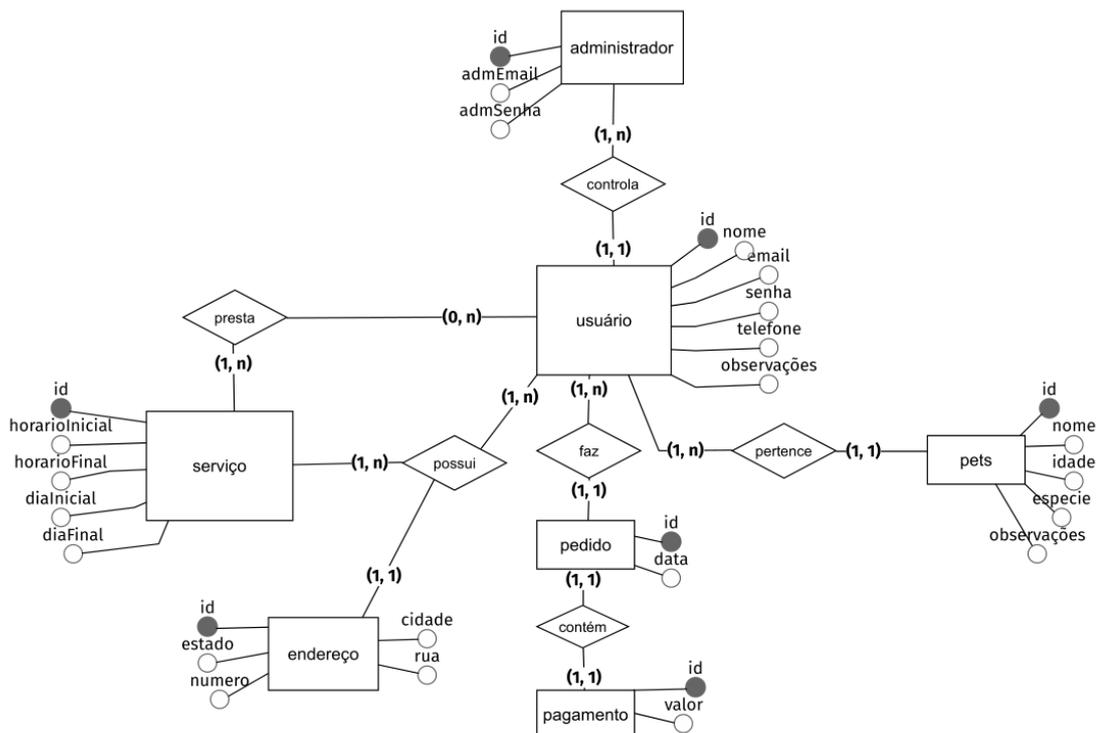
Quanto à modelagem do banco de dados temos o modelo Entidade-Relacionamento (ER) descrito por Toby Teorey em dois níveis: simples e complexo. O simples sendo entendido como:

aquele usado pela maioria das ferramentas de projeto atuais. É muito útil para o projetista de banco de dados que precisa se comunicar com os usuários finais sobre seus requisitos de dados. Nesse nível, você simplesmente descreve, em forma de diagrama, as entidades, atributos e relacionamentos que ocorrem no sistema a ser conceitualizado, usando uma semântica que pode ser definida em um dicionário de dados. (TEOREY et al., 2014)

A figura 3 ilustra o modelo ER da aplicação PetConfy com suas entidades e atributos: Administrador(atributos – id, admEmail, admSenha); Usuário(atributos – id, email, senha1, senha2, telefone, observacoes); Serviço(atributos – id, horarioInicial, horarioFinal, diaInicial, diaFinal); Pets(atributos- id, nome, idade, espécie, observacoes); Endereco(atributos – id, rua, numero, cidade, estado); Pedido(atributos - id, data); Pagamento(atributos – id, valor).

O complexo “inclui conceitos dos modelos semânticos da inteligência artificial e de modelos de dados conceituais concorrentes.” (TEOREY et al., 2014) que não será tratado neste trabalho.

Figura 3 – Modelo Entidade-Relacionamento.



Fonte: Elaborado pelo autor.

I.4.4 Ambiente de Desenvolvimento

A ferramenta utilizada para o desenvolvimento *backend* e *frontend* da aplicação foi o *VSCode*. Esta IDE (*integrated development environment*), com suas variadas extensões auxiliou totalmente no desenvolvimento da aplicação. Python foi a linguagem de programação escolhida, sendo ela eficaz e de abordagem simples para programação orientada a objetos. O micro *framework* Flask para desenvolvimento web foi selecionado para o projeto, ele que oferta duas completas bibliotecas externas: o motor de template *Jinja* e o conjunto de ferramentas *WSGI Werkzeug*. PostgreSQL também foi escolhido para compor o banco de dados da aplicação, sendo um dos mais avançados mercado, com alta confiabilidade, robustez e desempenho.

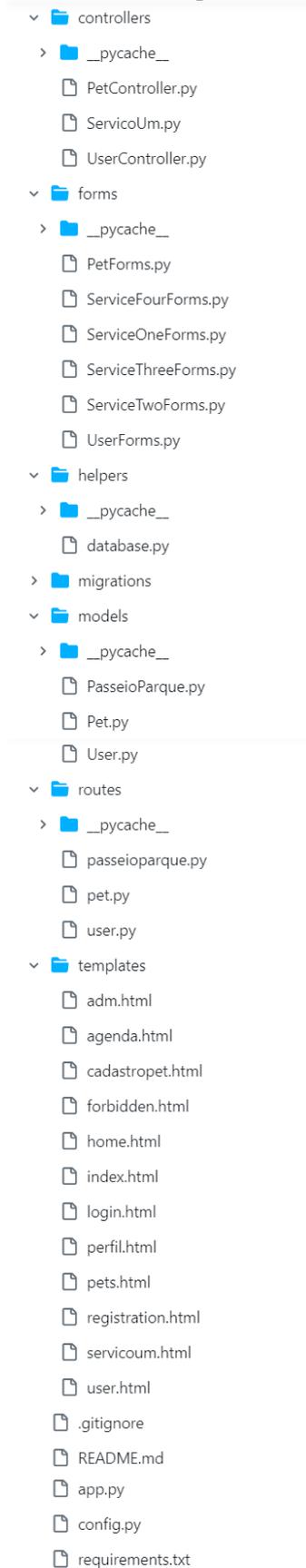
Em suma, todo o conjunto supriu as necessidades do projeto, principalmente levando em consideração a praticidade, aplicabilidade e qualidade.

I.5 Implementação

O projeto teve sua implementação efetuada a partir do *framework* Flask para a parte do *backend*, e a biblioteca *Bootstrap* (com auxílio do template *Jinja*) para a parte do *frontend*. A estrutura do código está descrita na figura 4 da seguinte maneira: tem-se o diretório principal, *petconfy*, que dentro dele encontramos os principais arquivos: *config.py* (que contém toda a informação de configuração da aplicação); *app.py* (arquivo que executa a aplicação web) e *requirements.txt* (aqui contém as dependências necessárias para o funcionamento da aplicação). Além destes arquivos, os diretórios: *controllers* (armazenando os métodos referentes às suas respectivas classes); *forms* (contém arquivos de formulários); *helpers* (arquivo que auxilia na instanciação da database e demais classes referentes ao *flask*, *flask_sqlalchemy*, *flask_migrate* e *flask_login*); *migrations* (que contém arquivos para gerência estrutural do banco de dados, podendo ser alterado sem maiores dificuldades); *models* (aqui estão os arquivos das classes para representação de tabela no banco de dados, contendo seus atributos e relacionamentos) e *routes* (aqui estão arquivos que detém as rotas/endpoints dos métodos de cada classe).

Mais abaixo está o último diretório *templates*, referente ao *frontend*, onde se encontram os arquivos html que contém a parte visual da aplicação, fazendo comunicação com o diretório *routes* a partir dos *links* que o template *Jinja* disponibiliza e adapta, permitindo a navegabilidade entre estes arquivos e suas respectivas visualizações.

O código-fonte da aplicação está hospedado na plataforma *GitHub*. (GITHUB, 2023)

Figura 4 – Organização do *backend* e *frontend* da aplicação

Fonte: acervo do autor

I.6 Conclusão

Em resumo, a aplicação PetConfy tem como objetivo sanar o problema da impossibilidade de acomodação animal assim como procura manter seus cuidados, segurança e prover recreação com o auxílio de pessoas-parceiras que ofertam seus serviços nas diferentes categorias disponíveis na plataforma. PetConfy tende a ser uma ótima opção quando se trata de responsabilidade para com os animais ao reunir muitas alternativas flexíveis para diferentes donos de pets, além do apreço pela segurança e organização.

No decorrer da execução do projeto houveram algumas impossibilidades de implementação, tais como: incorporar o calendário de eventos para o usuário prestador de serviços, visando facilitar a visualização das atividades contratadas e suas respectivas datas e horários; desenvolvimento do *frontend* da página de perfil do usuário; implementação dos serviços “Dia legal”, “Boa noite, companheiro” e “À bordo”; barra de pesquisa do usuário, para explorar por determinado serviço, além de filtro para buscas específicas; e a implementação dos sistemas de pagamento (cartão de crédito e boleto bancário).

REFERÊNCIAS

SBROCCO, J.; MACEDO, P. **Metodologias Ágeis**: engenharia de software sob medida. 1. ed. São Paulo: Ética, 2012.

GUEDES, Gilleanes T. A. **UML 2**: uma abordagem prática. 2. ed. São Paulo: Novatec Editora, 2011.

FOWLER, Martin. **UML essencial**: um breve guia para a linguagem-padrão de modelagem de objetos. 3. ed. Porto Alegre: Bookman, 2007.

TEOREY, Tobey, J. **Projeto e modelagem de banco de dados**. 2. ed. Rio de Janeiro: Elsevier, 2014.

ARAÚJO, M. **Modelagem de dados** – Teoria e prática. Revista Saber Digital, [S. l.], v. 1, n. 01, p. 27–64, 2021. Disponível em: <<https://revistas.faa.edu.br/SaberDigital/article/view/1029>>. Acesso em: 6 ago. 2023.

Build fast, responsive sites with Bootstrap. **Bootstrap**. Disponível em: <<https://getbootstrap.com>>. Acesso em: 24 de jun. 2023.

A simple framework for building complex web applications. **Flask**. Disponível em: <<https://flask.palletsprojects.com/en/2.3.x/#>>. Acesso em: 20 de jun. 2023.

PostgreSQL: The World Most Advanced Open Source Relational Database. **PostgreSQL**, 2023.

Disponível em: <<https://www.postgresql.org>>. Acesso em: 20 de jun. 2023

GitHub. Código fonte do sistema. **PetConfy**, 2023. Disponível em: <<https://github.com/marcellacg/petconfy>>. Acesso em: 25 de jun. 2023.

Code editing. Redefined. **Visual Studio Code**, 2023. Disponível em: <<https://code.visualstudio.com/>>. Acesso em: 25 de jun. 2023.

A very fast and expressive template engine. **Jinja**, 2023. Disponível em: <<https://jinja.palletsprojects.com/en/3.1.x/>> Acesso em: 24 de jun. 2023.

Werkzeug is a comprehensive WSGI web application library. **Werkzeug**, 2023. Disponível em: <<https://werkzeug.palletsprojects.com/en/2.3.x/>>. Acesso em: 28 de jun. 2023

Trello. **PetConfy**, 2023. Disponível em: <<https://trello.com/b/071LV5fP/petconfy>>. Acesso em 20 de jun. 2023.

ANEXO A - Cronograma de Acompanhamento das Atividades

< Baseado no Formulário Google >

ATIVIDADES	QUINZENAS (fevereiro/2023 a julho/2023)								
	1	2	3	4	5	6	7	8	9
A	X								
B		X							
C			X						
D				X					
E					X				
F						X			
G							X		
H								X	
I									X

Descrição

Atividade A - Discussão do tema: análise e objetivo do projeto; Pesquisa: visualizações de aplicações afins e suas possibilidades; preparação do ambiente: tecnologias a serem usadas para a configuração do ambiente de desenvolvimento;

Atividade B - Listagem de requisitos funcionais e não funcionais da aplicação; Construção do diagrama entidade-relacionamento; descrição da primeira parte do relatório referente ao tema e os objetivos;

Atividade C - Organização do quadro *Trello* da aplicação com os casos de testes referentes aos requisitos funcionais; Construção do diagrama de atividades; descrição da delimitação do problema referente ao projeto;

Atividade D - Desenvolvimento do *frontend* e *backend* da aplicação, referente ao login e ao cadastro de usuário; descrição da primeira parte do relatório (I.1.4) referente à justificativa da escolha do tema;

Atividade E - Desenvolvimento do *frontend* e *backend* da aplicação, referente ao cadastro de animais e a listagem de animais; descrição da primeira parte do relatório (I.1.5) referente ao método do trabalho;

Atividade F - Desenvolvimento do *frontend* e *backend* da aplicação referente ao cadastro de animais e a listagem de animais de cada usuário; descrição da primeira parte do relatório (I.1.5) referente ao método do trabalho;

Atividade G - Desenvolvimento do *frontend* e *backend* da aplicação referente às opções de deletar registro de animal e a edição de informações de animais de cada usuário; descrição da primeira parte do relatório (I.1.6) referente à organização do trabalho;

Atividade H - Desenvolvimento do *frontend* e *backend* da aplicação referente ao painel de usuário, contendo todas as opções de oferta de serviços, opção de cadastro animal, acesso ao perfil e edição de informações gerais; descrição da parte (I.2.1) do relatório, referente à descrição do problema, correspondente à (I.2);

Atividade I - Desenvolvimento do *frontend* e *backend* da aplicação referente à opção da agendamentos na página principal do usuário; implementação da área do administrador (com a listagem de usuários e barra de pesquisa de usuários); descrições das seguintes partes referentes ao relatório: I.3 sobre Requisitos do sistema (I.3.1 - Perfis de usuário, I.3.2 - Requisitos funcionais, I.3.4 - Requisitos não-funcionais); I.4 sobre Análise e Design (I.4.1 - Diagrama de classe, I.4.2 - Diagrama de Atividades, I.4.3 - Modelo de dados, I.4.5 - Ambiente de desenvolvimento) e I.5 sobre a Implementação.