

UNIVERSIDADE FEDERAL DE SANTA CATARINA
CENTRO TECNOLÓGICO – CTC
DEPARTAMENTO DE INFORMÁTICA E ESTATÍSTICA - INE
CURSO DE GRADUAÇÃO EM SISTEMAS DE INFORMAÇÃO

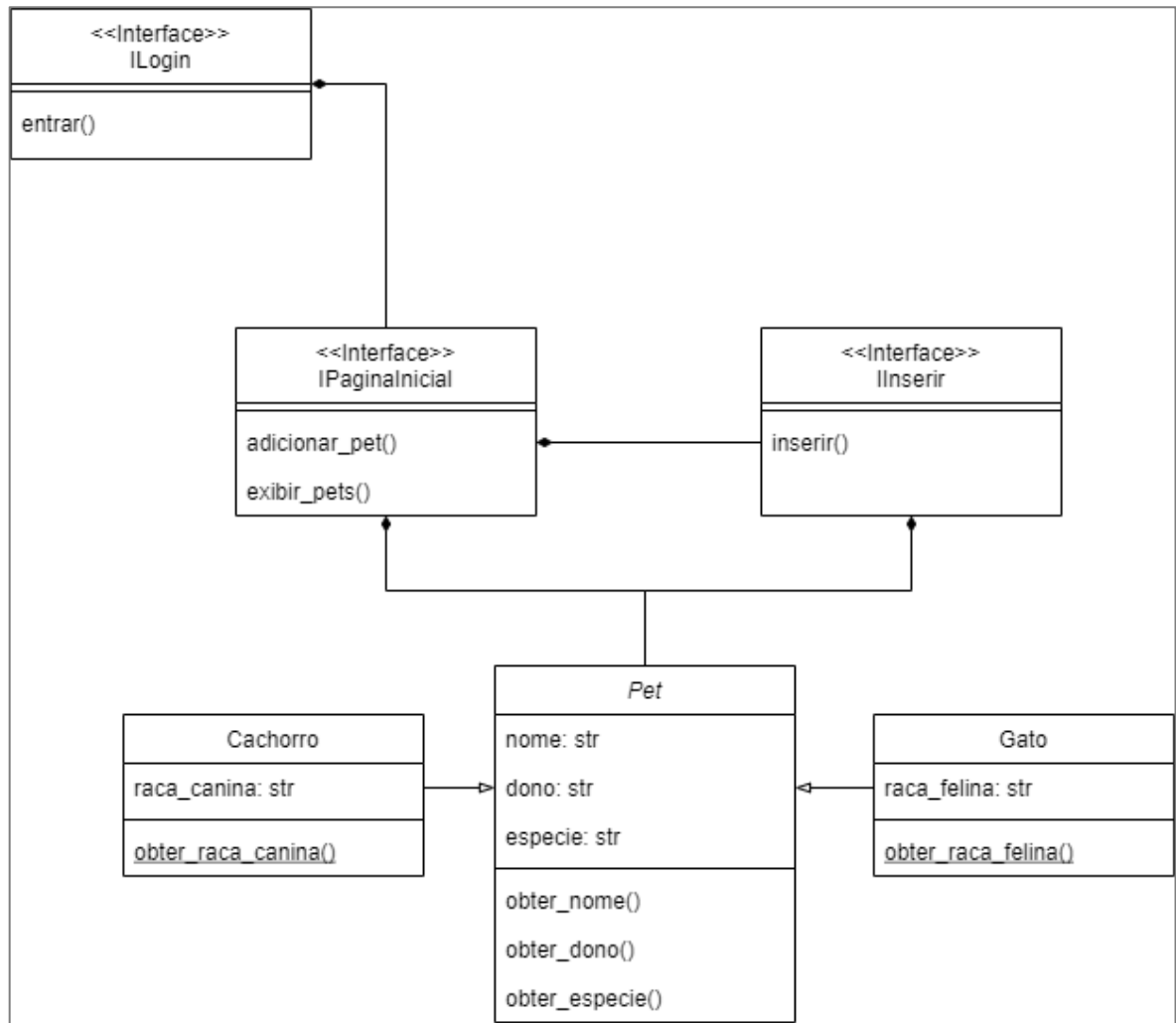
Beatriz Valio Weiss

Prova: Sistema de PetShop

Florianópolis

2021

UML – DIAGRAMA DE CLASSES



MANUAL

1.1 SOBRE O SISTEMA

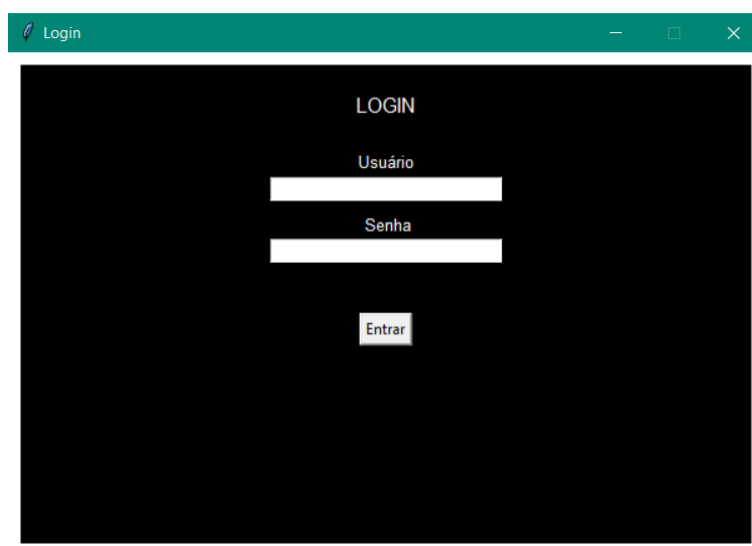
Para inicializar o sistema é preciso python 3, importar a biblioteca tkinter e ter uma IDLE configurada para a execução do código em python.

Há interface gráfica para consulta (IPaginaInicial), inserção (IInserção) e login (ILogin).

Todos os dados inseridos são armazenados na memória principal lista “lista_clientes”.

1.2 INTERFACE DE LOGIN: ILOGIN

O usuário deve inserir o nome de usuário e senha para então entrar no sistema.



1.3 INTERFACE PRINCIPAL: IPAGINAINICIAL

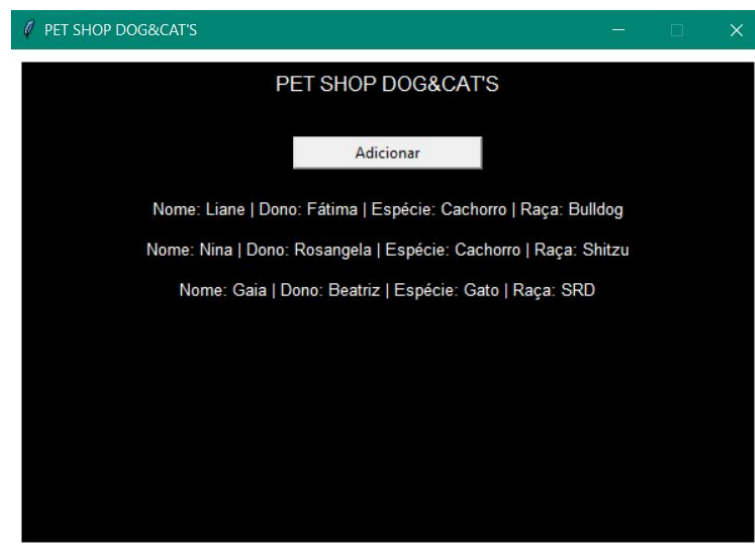
1.3.1 Primeira utilização

A página inicial do sistema é composta por um botão de adicionar novos clientes ao PetShop e um aviso de orientação para adicionar o primeiro cliente.



1.3.2Primeira utilização

A proposta é que nessa interface seja possível consultar os clientes do PetShop e adicionar novos clientes. Os clientes cadastrados são listados logo abaixo do botão de inserção.



1.4 INTERFACE INSERÇÃO DE NOVOS CLIENTES: IINSERIR

A página de cadastrar novos pets possui campos para inserir o nome do pet, o nome do dono, a espécie (cachorro ou gato) e a raça da espécie.



PET SHOP DOG&CAT'S

INSERIR NOVO PET

Nome do Pet:

Nome do Tutor:

Espécie:

Raça:

Pressione <Enter> para confirmar.

O usuário é informado de que confirmar a inserção deve pressionar a tecla Enter. Além disso, um aviso é mostrado caso não haja o preenchimento da ficha.



PET SHOP DOG&CAT'S

INSERIR NOVO PET

Nome do Pet:

Nome do Tutor:

Espécie:

Raça:

Pressione <Enter> para confirmar.

PREENCHA TODOS OS CAMPOS

A página inicial é encerrada ao abrir a página de cadastro e reaberta atualizada quando confirmada a adição do pet.

CÓDIGO PETSHOP.PY

```
import tkinter as tk
from tkinter import *
from tkinter import ttk

class Pet:

    def __init__(self, nome="", dono="", especie=""):
        self.nome = nome
        self.dono = dono
        self.especie = especie

    @property
    def obter_nome(self):
        return self.nome

    @property
    def obter_dono(self):
        return self.dono

    @property
    def obter_especie(self):
        return self.especie

class Cachorro(Pet):

    def __init__(self, nome="", dono="", especie="", raca_canina=""):
        super().__init__(nome, dono, especie)
        self.raca_canina = raca_canina

    @property
    def obter_raca_canina(self):
        return self.raca_canina

class Gato(Pet):

    def __init__(self, nome="", dono="", especie="", raca_felina=""):
        super().__init__(nome, dono, especie)
        self.raca_felina = raca_felina

    @property
    def obter_raca_felina(self):
        return self.raca_felina
```

```

class ILogin:

    def __init__(self):
        self.ILogin = tk.Tk()
        self.ILogin.title("Login")
        self.ILogin["background"] = "white"
        # Largura x Altura + distância Esquerda + distância Topo
        self.ILogin.geometry("600x400+500+200")
        self.ILogin.resizable(0, 0)

        self.frame_q1 = Frame(self.ILogin, borderwidth=1,
                                relief='solid', background='black')
        self.frame_q1.place(x=10, y=10, width=580, height=380)

        self.titulo = Label(self.ILogin, text="LOGIN",
                                background='black', foreground='white', font
="Arial 12")
        self.titulo.pack(pady=30, side=TOP)

        self.lblUsuario = Label(self.ILogin, text="Usuário",
                                background='black', foreground='white',
font="Arial 10")
        self.lblUsuario.place(x=275, y=76)
        self.usuario = Entry(self.ILogin, width=30,
                                text="")
        self.usuario.pack(pady=15, side=TOP)

        self.lblSenha = Label(self.ILogin, text="Senha",
                                background='black', foreground='white', fo
nt="Arial 10")
        self.lblSenha.place(x=280, y=126)
        self.senha = Entry(self.ILogin, width=30,
                                text="")
        self.senha.pack(pady=15, side=TOP)

        self.entrar = Button(self.ILogin, text='Entrar',
                                command=self.entrar)
        self.entrar.pack(pady=25, side=TOP)

        self.ILogin.mainloop()

    def entrar(self):
        self.ILogin.destroy()
        IPaginaInicial()

```



```

class IPaginaInicial:

    def __init__(self):

        self.IPaginaInicial = tk.Tk()
        self.IPaginaInicial.title("PET SHOP DOG&CAT'S")
        self.IPaginaInicial["background"] = "white"
        self.IPaginaInicial.geometry("600x400+500+200")
        self.IPaginaInicial.resizable(0, 0)

        self.frame_q2 = Frame(self.IPaginaInicial,
                               borderwidth=1, relief='solid', background=
'black')
        self.frame_q2.place(x=10, y=10, width=580, height=380)

        self.titulo = Label(self.IPaginaInicial, text="PET SHOP DOG&CAT'
S",
                               background='black', foreground='white', font
="Arial 12")
        self.titulo.pack(pady=15, side=TOP)

        self.adicionar_pet()
        self.exibir_pets()

        self.IPaginaInicial.mainloop()

    def adicionar_pet(self):
        self.adicionar = Button(self.IPaginaInicial, text='Adicionar', w
idth=20,
                               command=lambda: nova_window())
        self.adicionar.pack(pady=15, side=TOP)

    def nova_window():
        self.IPaginaInicial.destroy()
        IInserir()

    def exibir_pets(self):
        if lista_clientes != []:
            for n in lista_clientes:
                self.lbl = Label(self.IPaginaInicial, text="Nome: {} | D
ono: {} | Espécie: {} | Raça: {}".format(n[0], n[1], n[2], n[3]),
                               background='black', foreground='white',
font="Arial 10")
                self.lbl.pack(pady=5, side=TOP)
            else:
                self.lbl = Label(self.IPaginaInicial, text="[Adicione seu pr
imeiro cliente clicando no botão Adicionar]",

```

```

                                background='black', foreground='white', font
t="Arial 10")
        self.lbl.pack(pady=100, side=TOP)

class IInserir:

    def __init__(self):
        self.IInserir = tk.Tk()
        self.IInserir.title("PET SHOP DOG&CAT'S")
        self.IInserir["background"] = "white"
        self.IInserir.geometry("600x400+500+200")
        self.IInserir.resizable(0, 0)
        self.frame_q3 = Frame(self.IInserir, borderwidth=1,
                                relief='solid', background='black')
        self.frame_q3.place(x=10, y=10, width=580, height=380)
        self.titulo = Label(self.frame_q3, text="INSERIR NOVO PET",
                                background='black', foreground='white', font
="Arial 12")
        self.titulo.pack(pady=30, side=TOP)

        self.lblNome = Label(self.IInserir, text="Nome do Pet:",
                                background='black', foreground='white', font
t="Arial 10")
        self.lblNome.place(x=205, y=85)
        self.entreNome = Entry(self.frame_q3, width=30)
        self.entreNome.pack(pady=15, side=TOP)
        self.conteudoNome = tk.StringVar()
        self.conteudoNome.set("")
        self.entreNome["textvariable"] = self.conteudoNome
        self.entreNome.bind('<Key-Return>', self.inserir)

        self.lblDono = Label(self.IInserir, text="Nome do Tutor:",
                                background='black', foreground='white', font
t="Arial 10")
        self.lblDono.place(x=205, y=135)
        self.entreDono = Entry(self.frame_q3, width=30)
        self.entreDono.pack(pady=15, side=TOP)
        self.conteudoDono = tk.StringVar()
        self.conteudoDono.set("")
        self.entreDono["textvariable"] = self.conteudoDono
        self.entreDono.bind('<Key-Return>', self.inserir)

        self.lblEspecie = Label(self.IInserir, text="Espécie:",
                                background='black', foreground='white',
font="Arial 10")
        self.lblEspecie.place(x=205, y=185)
        self.entreEspecie = Entry(self.frame_q3, width=30)

```

```

self.entreEspecie.pack(pady=15, side=TOP)
self.conteudoEspecie = tk.StringVar()
self.conteudoEspecie.set("")
self.entreEspecie["textvariable"] = self.conteudoEspecie
self.entreEspecie.bind('<Key-Return>', self.inserir)

self.lblRaca = Label(self.IInserir, text="Raça:",
                     background='black', foreground='white', font="Arial 10")
self.lblRaca.place(x=205, y=235)
self.entreRaca = Entry(self.frame_q3, width=30)
self.entreRaca.pack(pady=15, side=TOP)
self.conteudoRaca = tk.StringVar()
self.conteudoRaca.set("")
self.entreRaca["textvariable"] = self.conteudoRaca
self.entreRaca.bind('<Key-Return>', self.inserir)

self.lblConfirmar = Label(self.IInserir, text="Pressione <Enter>
para confirmar.",
                          background='black', foreground='white'
, font="Arial 10")
self.lblConfirmar.place(x=200, y=285)

self.IInserir.mainloop()

def inserir(self, event):
    novo_cliente = []
    if self.conteudoNome.get() == "" and self.conteudoDono.get() == ""
and self.conteudoEspecie.get() == "" and self.conteudoRaca.get() == "
":
        self.lblAviso = Label(self.IInserir, text="PREENCHA TODOS OS
CAMPOS",
                              background='black', foreground='red',
font="Arial 9")
        self.lblAviso.place(x=205, y=345)
    else:
        novo_cliente.extend([self.conteudoNome.get(), self.conteudoD
ono.get(
), self.conteudoEspecie.get(), self.conteudoRaca.get()])
        lista_clientes.append(novo_cliente)
        self.IInserir.destroy()
        IPaginaInicial()

lista_clientes = []
ILogin()

```