

	UNIVERSIDADE FEDERAL DE SANTA CATARINA
	Disciplina: INE5605 – Desenvolvimento de Sistemas Orientados a Objetos I
	Professores: Thaís Idalino, Douglas Hiura Longo e André Bräscher

Apresentação do Tema – T1

Matheus Silvano Pereira – Handryos Godosi dos Santos

Problema:

Desenvolver um sistema orientado a objetos em Python para gerenciamento de um pet shop ou clínica veterinária, permitindo o cadastro e a organização de pets, donos, serviços oferecidos e agendamentos, além de funcionalidades extras como histórico, cálculo de valores e exportação de dados.

Escopo do desenvolvimento:

O pet shop atende clientes com diversos tipos de animais e oferece diferentes serviços, como banho, tosa e consultas veterinárias.

O sistema deve permitir:

- Inclusão, remoção e edição de pets e de seus respectivos donos.
- Inclusão, remoção e edição dos serviços oferecidos.
- Agendamento de serviços por data e horário.
- Organização das informações e facilidade na atualização de dados.

Regras do sistema:

1. Cada pet deve estar vinculado a um único dono.
2. Um dono pode possuir vários pets, mas um pet só pode ter um dono principal.
3. O cadastro de pets deve conter todos os dados obrigatórios: Nome, espécie, raça, idade e dono.
4. Somente donos cadastrados podem agendar serviços.
5. Os serviços devem estar previamente cadastrados para poderem ser agendados.
6. Não é permitido agendar dois serviços para o mesmo pet no mesmo horário.
7. O sistema deve verificar conflitos de horário.

8. A idade do pet deve ser um número inteiro positivo.
9. Os agendamentos só podem ser marcados para datas futuras.
10. O sistema deve permitir a alteração e exclusão de dados cadastrados.
11. O cancelamento de um agendamento deve ser registrado com data e hora.
12. Deve ser possível listar todos os serviços já realizados por um pet.
13. O valor total gasto por dono deve ser calculado com base nos serviços realizados.
14. O sistema deve permitir alteração rápida de telefone e endereço.

Restrições do Escopo:

Para manter o sistema simples, funcional e dentro de um escopo didático, algumas limitações foram definidas:

- Não será feita integração com bancos de dados externos.
- Não será implementado controle de acesso com login e senha. O foco está na estrutura e funcionamento interno das funcionalidades.
- O sistema não lidará com questões legais como termos de uso, consentimento de dados ou contratos de serviços.
- Não será considerado o estoque de produtos físicos (ração, brinquedos, medicamentos, etc). O foco será exclusivamente nos serviços prestados aos pets (banho, tosa, consulta etc).
- O sistema não contemplará pagamentos, faturas ou controle financeiro detalhado. Apenas o valor total dos serviços será calculado por dono, sem integração com métodos de pagamento.
- A persistência, se implementada, será feita em arquivos locais (ex: .csv, .json ou .txt).
- O histórico de agendamentos será limitado à sessão atual, salvo se for incluída exportação para CSV.

Associação, Agregação e Composição:

Associação: Agendamento associa Pet, Dono e Serviço

Agregação: Dono tem uma lista de pets (Pet pode existir sem Dono)

Composição: Agendamento é composto por Pet, Serviço, etc — se o agendamento for deletado, ele perde o vínculo direto.

Divisão das Responsabilidades:

Matheus – Estrutura de Dados e Lógica de Negócio:

1. Modelagem e implementação das classes principais:

Pet: nome, espécie, raça, idade, dono.

Dono: nome, telefone, endereço, lista de pets.

Serviço: pet, dono, preço.

Agendamento: data, hora, pet, serviço, status.

2. Lógica de regras de negócio:

Verificar conflitos de horário.

Impedir agendamento para datas passadas.

Validar idade dos pets como inteiro positivo.

Garantir que só donos cadastrados possam agendar.

Garantir que não haja dois serviços no mesmo horário para um pet.

3. Geração de relatórios:

Geração do relatório de pets, donos, agendamentos, etc.

Exportação de dados em CSV (se aplicável)

Handryos – Interface, Fluxo do Sistema e Persistência:

1. Interface do usuário:

Menu principal e navegação.

Telas para cadastro/edição de pets, donos, serviços e agendamentos.

2. Fluxo de operações:

Inclusão, remoção e edição de entidades.

Cancelamento de agendamentos com registro.

Alteração rápida de telefone e endereço.

3. Persistência de dados (se aplicável):

Salvar e carregar dados em arquivos locais (.json, .csv ou .txt).

A modelagem UML foi realizada em colaboração, com ambos participando.