

UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ  
PROJETO FINAL DE ANÁLISE E PROJETO ORIENTADO A OBJETOS

**IAGO ORTEGA CARMONA**

**SISTEMA DE GERENCIAMENTO DE UMA ACADEMIA DE  
MUAYTHAI E BOXE (GYM MANAGER)**

CAMPO MOURÃO  
2022

# Sistema automatizado de gerenciamento de alunos, treinos e pagamentos de uma academia de muaythai e boxe(Gym Manager)

Iago Ortega Carmona 2277980

## 1 Introdução

Sistema de gerenciamento de usuários de uma academia de Muaythai e Boxe, na qual está localizada na cidade de Cianorte no estado do Paraná. Esta academia se encontram usuários de dois tipos diversos, nos quais temos o aluno do Boxe e MuayThai, sendo assim, então foi necessário a criação de um sistema visando melhorar o monitoramento da academia e facilitar o serviço do instrutor da mesma,. Neste arquivo será apresentado as funcionalidades do sistema, onde os mesmos foram requisitados pelo cliente.

### 1.1 Propósito

O nosso objetivo é atender os requisitos de um sistema automatizado de gerenciamento de usuários em uma academia de Muaythai e Boxe, proporcionado facilidade de uso e acesso dos alunos da academia e também melhorando a comunicação com o instrutor.

### 1.2 Escopo

O sistema permitirá coletar informações dos alunos sendo nome, sobrenome, telefone, rg, cpf, endereço e data de nascimento, para realizar o cadastro na academia, o instrutor poderá atualizar, cadastrar, deletar e acessar os treinos semanais registrados no sistema, tendo as informações do treino, como a descrição, o tipo do treino, nível do treino(leve, mediano ou pesado) e o horário. Sendo assim, o instrutor como administrador geral da academia, poderá atualizar, deletar, cadastrar e acessar os dados dos alunos no sistema.

## 2 Descrição geral do sistema

Devido ao crescimento de clientes na academia, se deu a necessidade de criar um sistema para melhor gerenciar. O objetivo do sistema é controlar os pagamentos mensais dos clientes da academia e gerenciar quais planos o usuário

está cadastrado, incluindo também o gerenciamento facilitado dos horários dos treinos semanais.

## **2.1 Funções do produto**

O Sistema Gym Manager tem como principal objetivo facilitar a vida do dono de academia.

- Gerenciamento de clientes, sendo o cadastro, remoção, consulta, atualização.
- Gerenciamento de horário dos treinos, sendo criação, remoção, consulta e atualização.
- Gerenciamento de pagamentos, sendo autenticação, emissão de nota fiscal.

## **2.2 Características do usuário**

O sistema Gym Manager tem como principal público alvo academias que necessitam de um controle de clientes, horários e pagamentos, para facilitar o gerenciamento.

# **3 Requisitos específicos**

## **3.1 Requisitos funcionais**

### **Cadastro de usuários**

- RF1. O sistema deve permitir a inclusão, alteração e remoção de usuários do sistema. Os dados do usuário consistem de: nome completo, apelido, endereço, CPF, cidade, bairro, estado, telefone fixo, telefone celular e e-mail.

### **Gerenciamento de horários dos treinos**

- RF2. O sistema deve permitir a inclusão, alteração, remoção e consultas de horários de treinos internos da academia.

### **Gerenciamento de pagamentos**

- RF3. O sistema deve permitir ao instrutor da academia gerar e consultar notas fiscais de pagamentos.

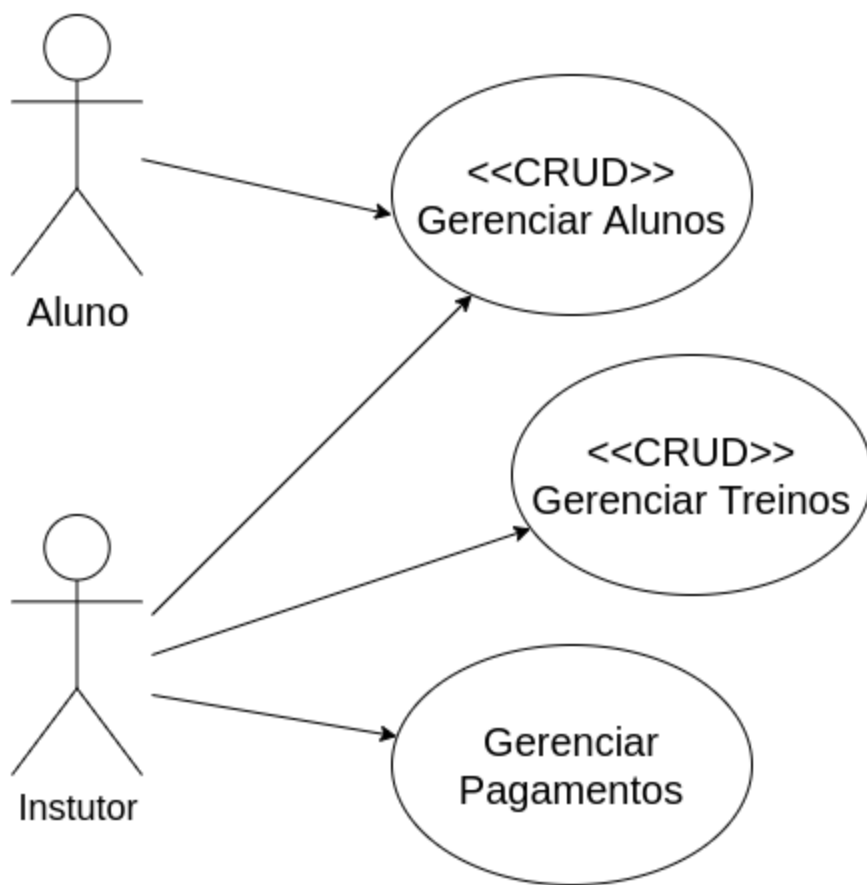
## **3.2 Requisitos não funcionais**

- RN1. O tempo de resposta para qualquer as operações relacionadas à inserção, alteração, exclusão e consulta não deve exceder 1 segundo.

- RN2. O sistema deve ter capacidade para recuperar os dados perdidos da última operação que realizou em caso de falha.
- RN3. O sistema deve fornecer facilidades para a realização de cópias de segurança dos dados do sistema pelo administrador.
- RN4. O sistema deve ser capaz de emitir notas fiscais em pelo menos 4 segundos.

# Diagrama de Casos de Uso

Aluno: Iago Ortega Carmona RA: 2277980



## Diagrama de casos de uso textuais

**Caso de uso:** gerenciar alunos

**Visão geral:** O instrutor deseja cadastrar, ler, atualizar e deletar no sistema, os dados dos alunos. Os dados são fornecidos pelo aluno, e posteriormente são atualizados em caso de necessidade.

Gerenciar treinos

X

Horário

Data

Tipo de treino

Finalizar



**Caso de uso:** gerenciar treinos

**Ator Principal:** instrutor

**Interesses e Interessados:**

Instrutor: deseja criar horários de treinos para os alunos acessarem no sistema.

Aluno: deseja acessar os horários criados pelo instrutor da academia.

**Pré-condições:** os alunos devem estar cadastrados no sistema.

**Pós-condições:** os dados cadastrados para os treinos serão armazenados no sistema para os alunos terem acesso.

**Cenário de Sucesso Principal:**

- 1- O instrutor acessa o sistema para cadastrar treinos.
- 2- O instrutor insere no campo “Horário”, o horário do treino.
- 3- O instrutor insere no campo “Data”, a data do treino.
- 4- O instrutor insere no campo “Tipo de treino”, o tipo do treino, sendo: leve, mediano ou pesado.
- 5- O instrutor clica no botão de “Finalizar”, para finalizar o cadastro do treino.

**6-** O sistema devolve em caso de sucesso "Treino cadastrado com sucesso!", em caso contrário, devolve "Impossível realizar cadastro do treino...".

**7-** O sistema armazena os dados inseridos em um banco de dados.

**Caso de uso:** gerenciar pagamentos

**Ator principal:** Instrutor

**Interesses e Interessados:**

Instrutor: deseja armazenar os últimos pagamentos dos alunos, tendo um controle das mensalidades e podendo requisitar ao sistema o se o ultimo pagamento já foi efetuado.

Aluno: deseja realizar o pagamento mensal da academia

**Pós-Condições:** os dados dos pagamento são armazenados no sistema.

**Cenário de Sucesso Principal:**

**1-** O instrutor solicita os dados do aluno para verificar se há cadastro no sistema.

**2-** Caso haja cadastro, o sistema retorna os dados do aluno, se não, será necessário fazer cadastro do mesmo.

**3-** O aluno fornece seus documentos.

**4-** O instrutor realiza o cadastro do aluno no sistema.

**5-** O instrutor solicita ao aluno o método de pagamento.

**6-** O aluno fornece o método desejado.

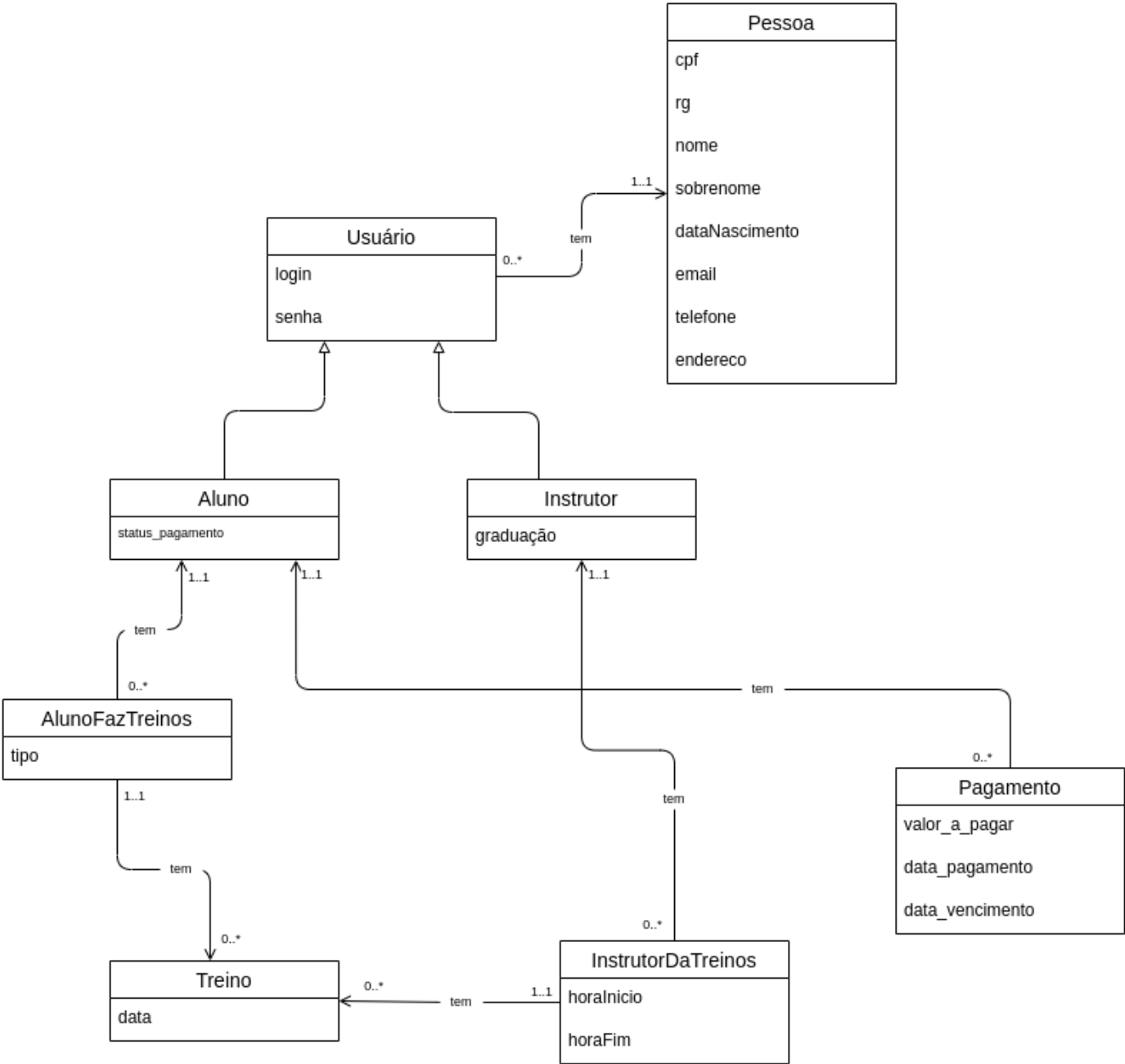
**7-** O instrutor realiza o pagamento e armazena no sistema os dados do pagamento.

**Fluxo alternativo 1:**

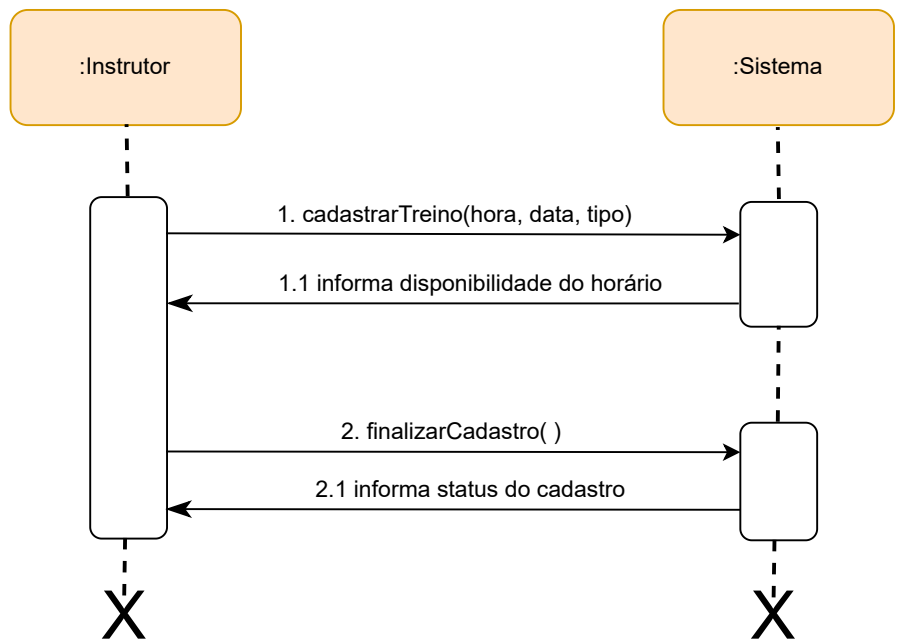
**1.1-** O aluno não tem seus documentos em mãos, sendo assim a operação é cancelada.

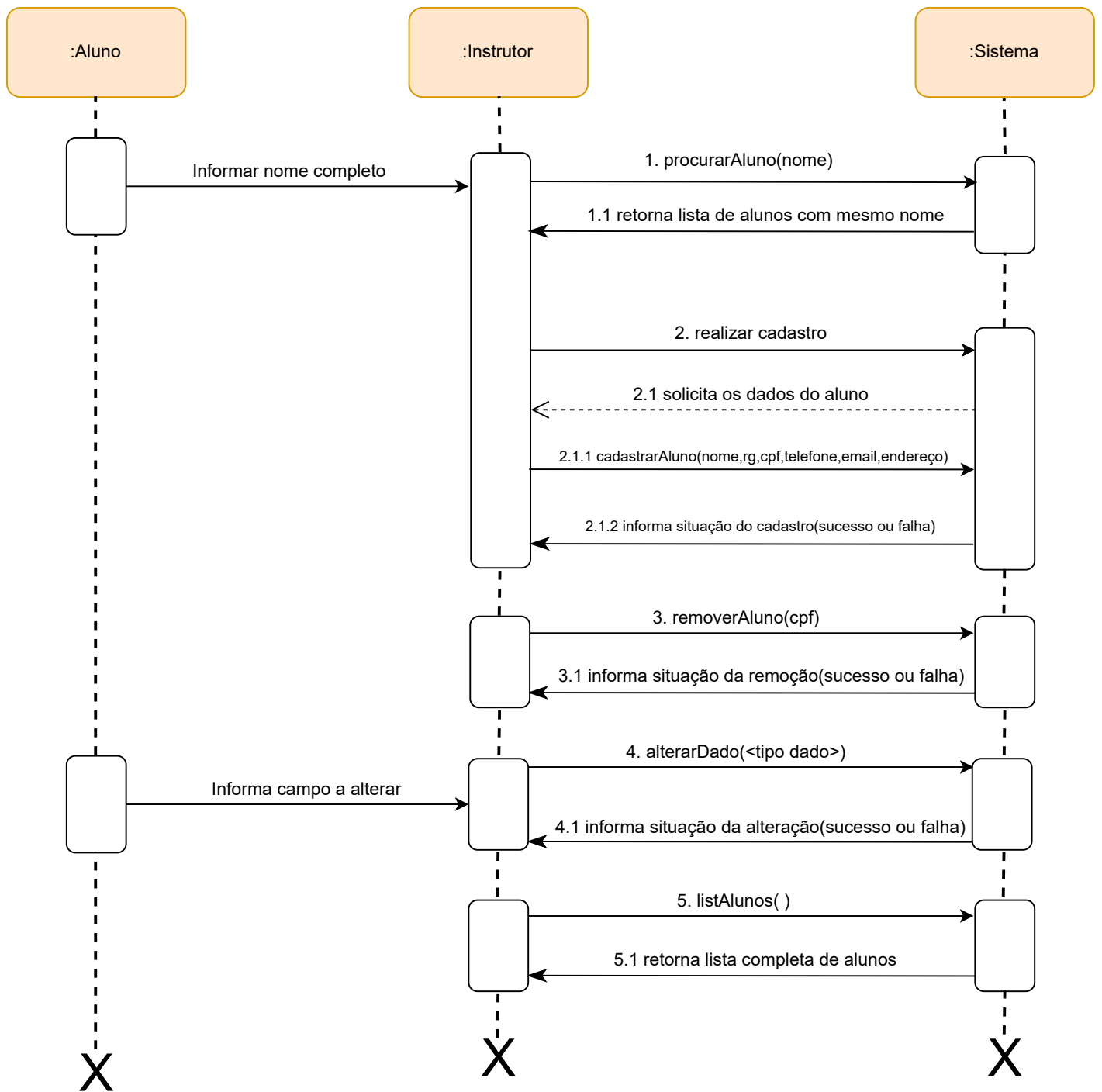
# Modelo Conceitual

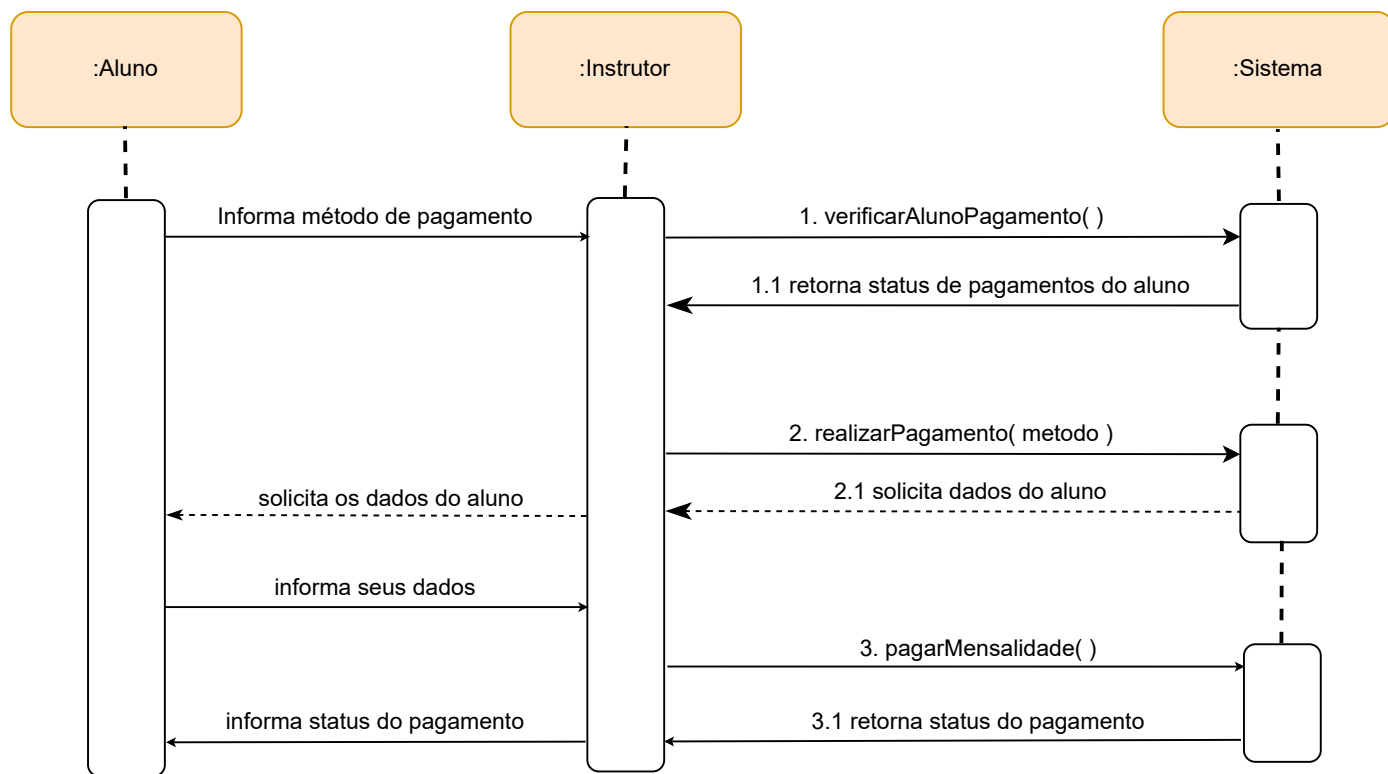
Aluno: Iago Ortega Carmona    RA: 2277980











## Contrato de Operações

### 1. Operações

- verificarAlunoPagamento(cpf)
- realizarPagamento(metodo)
- pagarMensalidade()
- procurarAluno(nome)
- listAlunos()
- cadastrarTreino()

### 2. Contrato de Operações

#### 2.1 verificarAlunoPagamento(cpf)

<b>Operação:</b> verificarAlunoPagamento()
<b>Parâmetros:</b> cpf → Identificação do aluno
<b>Referências Cruzadas:</b> Caso de uso: “Gerenciar Pagamentos”
<b>Pré-condições:</b> O aluno especificado deve estar cadastrado no sistema.
<b>Pós-condições:</b> O aluno especificado estará disponível o pagamento da mensalidade.

#### 2.2 realizarPagamento(metodo)

<b>Operação:</b> realizarPagamento(metodo)
<b>Parâmetros:</b> metodo → metodo de pagamento desejado pelo aluno
<b>Referências Cruzadas:</b> Caso de uso: “Gerenciar Pagamentos”
<b>Pré-condições:</b> Deve haver disponibilidade de pagamento da mensalidade.
<b>Pós-condições:</b> O pagamento estará disponível para o aluno.

#### 2.3 pagarMensalidade(valor)

<b>Operação:</b> pagarMensalidade(valor)
<b>Parâmetros:</b> valor → Valor da mensalidade do aluno
<b>Referências Cruzadas:</b> Caso de uso: “Gerenciar Pagamentos”
<b>Pré-condições:</b> O pagamento deve estar disponível.
<b>Pós-condições:</b> O status de pagamento do aluno será alterado para ‘PAGO’

### 2.3 procurarAluno(nome)

<b>Operação:</b> procurarAluno(nome)
<b>Parâmetros:</b> nome → nome do aluno desejado para realizar busca
<b>Referências Cruzadas:</b> Caso de uso: “Gerenciar Alunos”
<b>Pré-condições:</b> O aluno deve estar cadastrado no sistema.
<b>Pós-condições:</b> O sistema retornará uma lista de alunos com o mesmo nome.

### 2.4 listAlunos()

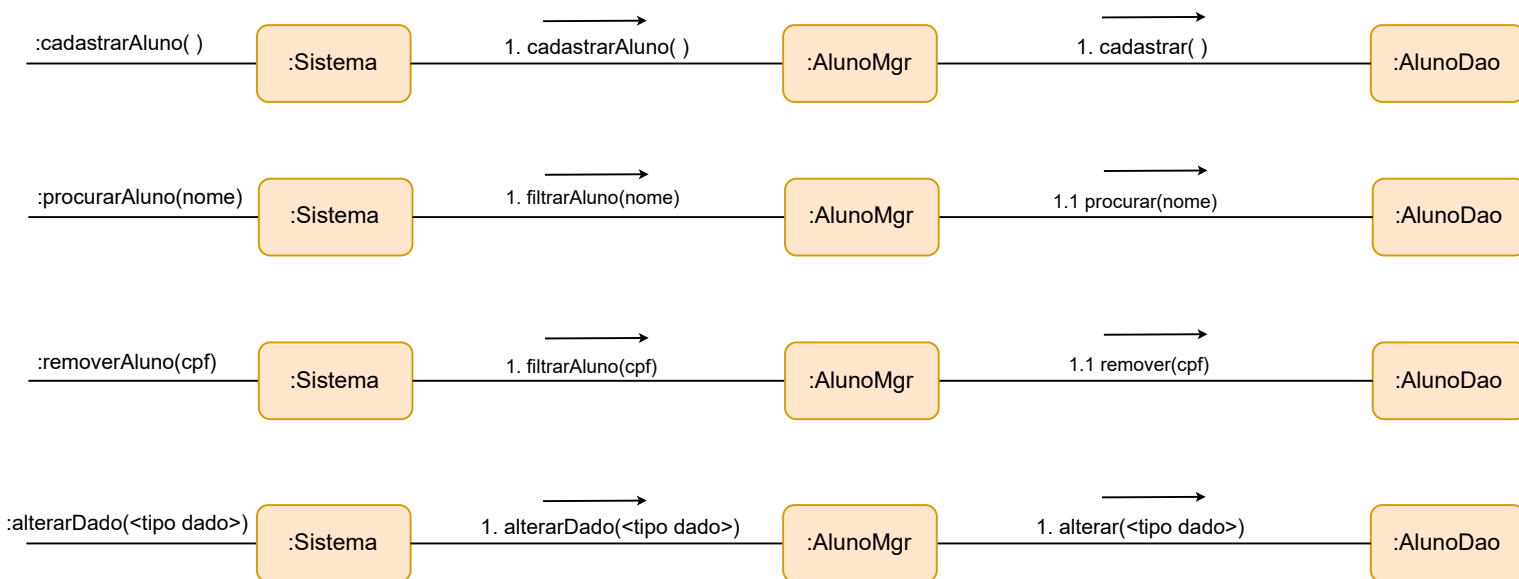
<b>Operação:</b> listAlunos()
<b>Referências Cruzadas:</b> Caso de uso: “Gerenciar Alunos”
<b>Pré-condições:</b> O aluno deve estar cadastrado no sistema.
<b>Pós-condições:</b> O sistema retornará uma lista de todos os alunos cadastrados no sistema.

### 2.5 cadastrarTreino(hora, data, tipo)

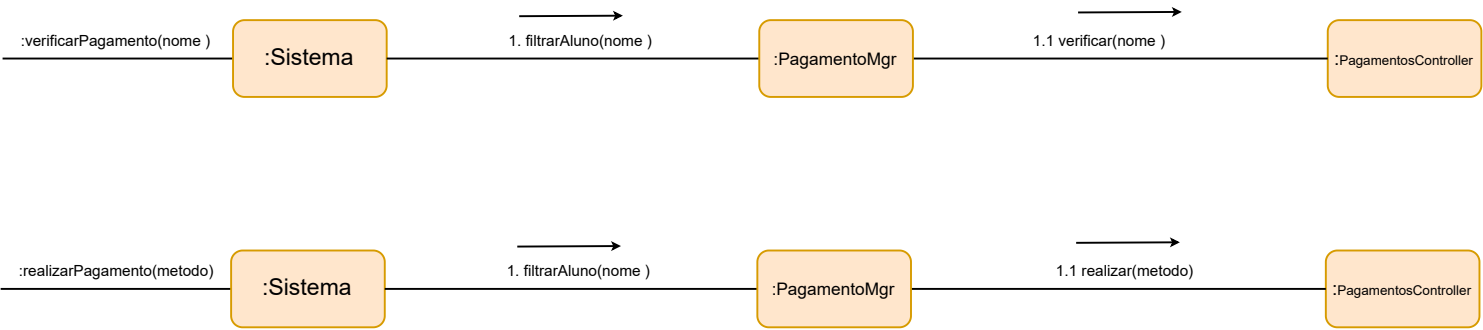
<b>Operação:</b> cadastrarTreino(hora, data, tipo)
<b>Parâmetros:</b> hora → horário do treino desejado pelo instrutor data → data do treino desejado pelo instrutor tipo → tipo do treino desejado pelo instrutor, sendo: boxe ou muaythai.
<b>Referências Cruzadas:</b> Caso de uso: “Gerenciar Treinos”
<b>Pré-condições:</b> O instrutor estar devidamente autenticado no sistema.
<b>Pós-condições:</b> O sistema retornará uma lista dos treinos cadastrados naquela semana.

## Diagrama de comunicação

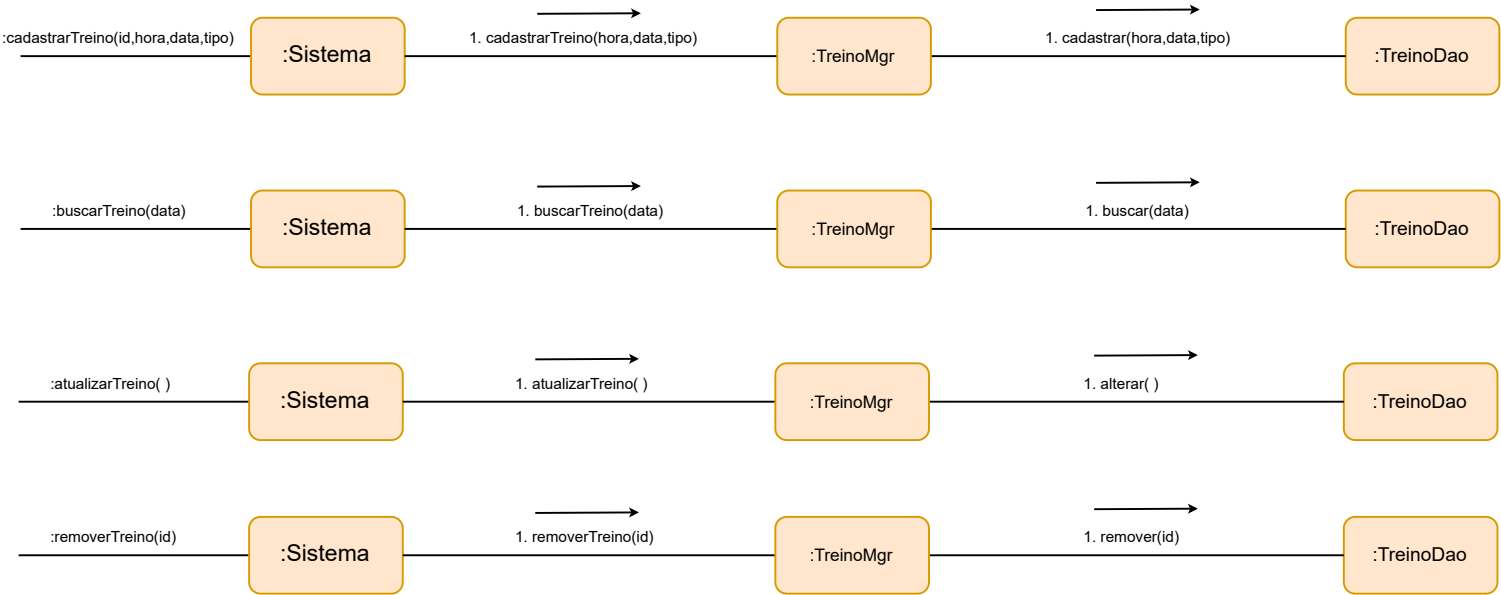
### Caso de uso: Gerenciar Alunos



Caso de uso: Gerenciar Pagamentos



# Caso de uso: Cadastrar Treino





# Diagrama de classes

Aluno: Iago Ortega Carmona RA: 2277980

