

# ÂNIMA - UNIVERSIDADE POTIGUAR

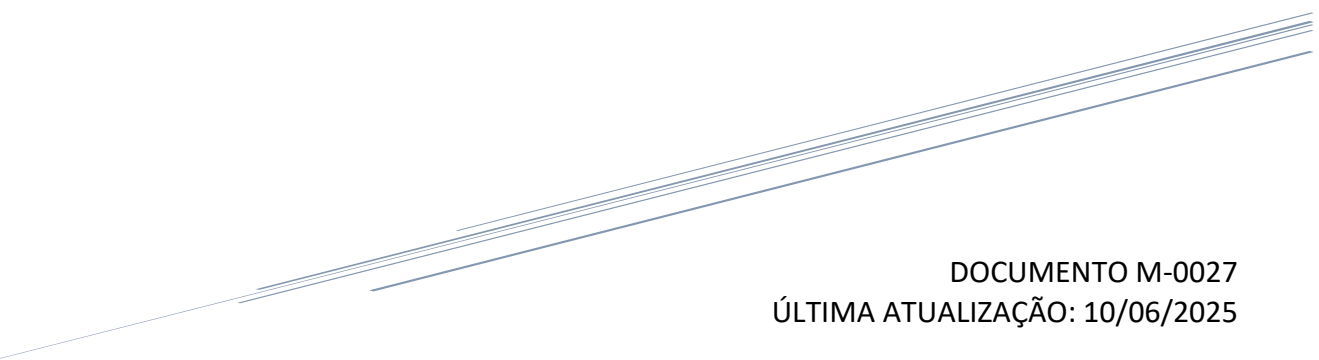
PROJETO DE PROGRAMAÇÕES E SOLUÇÕES COMPUTACIONAIS.

## Sistema de Academia Javasfit

Natal, 2025

Preparado por

Otávio Alves, Nuryan Yasmin, Diego Carvalho, Edna Menezes , Ciências da Computação, Noturno

A series of approximately five parallel blue lines that originate from the left side of the page and extend diagonally upwards towards the right, creating a sense of motion or a stylized underline.

DOCUMENTO M-0027  
ÚLTIMA ATUALIZAÇÃO: 10/06/2025

## Sumário

1. Introdução .....	3
Escopo do Produto .....	3
Especificações.....	3
2. Convenções, termos e abreviações .....	<b>Erro! Indicador não definido.</b>
Identificação dos Requisitos.....	<b>Erro! Indicador não definido.</b>
Prioridades dos Requisitos .....	<b>Erro! Indicador não definido.</b>
3. Descrição geral do sistema .....	<b>Erro! Indicador não definido.</b>
Abrangência e sistemas relacionados .....	4
Descrição dos usuários.....	4
4. Requisitos funcionais.....	4
Casos de Uso .....	7
5. Requisitos não funcionais.....	8
Usabilidade.....	8
Confiabilidade .....	8
Desempenho .....	8
Segurança .....	8
6. Artefatos UML .....	8
7. Banco de Dados .....	12
8. Descrição da interface com o usuário .....	12
9. CONCLUSÕES E RECOMENDAÇÕES .....	13
10. REFERÊNCIAS.....	13

## Histórico de Revisão

DATA	VERSÃO	DESCRIÇÃO DA ALTERAÇÃO	RESPONSÁVEL

## 1) Introdução

O Sistema em questão é um sistema de academia, planejado para que os usuários sejam os funcionários da mesma com o objetivo de que possam de forma mais dinâmica adquirir uma administração mais segura e precisa. O Sistema divide as funções da equipe de forma simples e

## 2) Escopo do Produto

O software especificado é um Sistema de Gerenciamento de Academia, projetado para gerenciar as operações administrativas, financeiras e de treinamento de uma academia. Seu propósito é otimizar a administração de alunos, treinos, pagamentos e planos de assinatura, promovendo eficiência operacional, organização dos dados e melhoria na experiência dos usuários (funcionários e membros). Os benefícios incluem maior controle sobre cadastros, pagamentos e treinos, além de facilitar a supervisão gerencial e a personalização de planos de treino para os alunos.

- Nome do Sistema: Sistema de Academia Javasfit(SAJ)
- O sistema permite:
- Gerenciamento de cadastros de alunos (registrar, atualizar, buscar e deletar).
- Gerenciamento de exercícios (registrar, atualizar, buscar e deletar).
- Supervisão de funcionários (cadastro, atualização e exclusão).

O sistema permitirá:

- Administração de treinos (criação, atualização, exclusão e consulta de treinos e exercícios).
- Gestão de pagamentos (registro, visualização e administração de mensalidades).
- Gerenciamento de planos de assinatura (criação, atualização e exclusão).
- Supervisão de funcionários (cadastro, atualização e exclusão).
- Visualização de históricos de treinos e relatórios financeiros.

O que o Produto não faz:

- Não gerencia reservas de equipamentos ou espaços físicos da academia (como salas de aula ou máquinas).
- Não inclui funcionalidades de integração com dispositivos de monitoramento de saúde (ex.: wearables).
- Não suporta agendamento de aulas coletivas ou eventos.
- Não oferece suporte a vendas de produtos (ex.: suplementos ou acessórios). Essas funcionalidades podem ser consideradas para projetos futuros, mas estão fora do escopo atual devido à priorização de processos administrativos e de treinamento.
- O sistema será usado por professores, secretários, gerentes e membros da academia para gerenciar operações diárias. Ele melhora a eficiência ao centralizar informações, reduz erros manuais e facilita a comunicação entre os atores. Os objetivos incluem aumentar a satisfação dos alunos com planos personalizados, melhorar o controle financeiro e agilizar processos administrativos.
- O software alinha-se às metas corporativas de aumentar a retenção de alunos, melhorar a gestão financeira e garantir a qualidade do serviço. Ele suporta os macroprocessos de administração de clientes, gestão financeira e planejamento de treinos, contribuindo para a competitividade da academia no mercado.

## 3) Especificações do Hardware

**Processador:** Processador multicore de 2 GHz ou superior (ex.: Intel Core i5 ou equivalente) para suportar o processamento de dados em tempo real.

**Memória RAM:** Mínimo de 8 GB para garantir desempenho fluido em operações simultâneas.

**Armazenamento:** Pelo menos 500 GB de armazenamento em disco (HDD ou SSD) para banco de dados com cadastros, treinos e histórico de pagamentos.

**Dispositivos de Entrada/Saída:** Teclado, mouse e monitor com resolução mínima de 1280x720 para interação com a interface.

**Conexão de Rede:** Conexão estável à internet (mínimo 10 Mbps) para backups em nuvem e acesso remoto, se aplicável.

**Sistema Operacional:** Compatível com Windows 7.

## 4) Abrangência e sistemas relacionados

Abrangência (escopo positivo)

- Cadastro e autenticação de usuários (membros e funcionários).
- Registro e controle de pagamentos, com aplicação de descontos/multas conforme regras de negócio.
- Planejamento, registro e histórico de treinos e exercícios.

Escopo negativo

- Não haverá, nesta versão, integração com sistemas de nutrição, monitoramento por sensores ou apps de terceiros.
- Relatórios de marketing e CRM ficam fora do escopo inicial

Sistemas relacionados

- Banco de Dados Relacional - MySQL - para armazenar informações de usuários, pagamentos, treinos e planos.

### Descrição dos usuários

Os usuários do sistema incluem funcionários da academia (professores, secretários e gerentes) e membros (alunos). Eles enfrentam desafios como falta de centralização de informações, processos manuais demorados e dificuldade em acompanhar históricos de treinos e pagamentos. O sistema visa resolver esses problemas ao oferecer uma plataforma integrada para gerenciar todas as operações.

#### 5.1 Professor

**Características:** Profissional responsável por criar e gerenciar treinos personalizados.

**Desafios:** Dificuldade em organizar treinos manualmente, acompanhar o progresso dos alunos e manter um histórico acessível.

**Necessidades:** Interface para registrar, atualizar e consultar treinos/exercícios, além de visualizar históricos de alunos.

#### 5.2 Secretário

**Características:** Funcionário administrativo que gerencia cadastros de alunos e pagamentos.

**Desafios:** Processos manuais de cadastro e controle de mensalidades, propensos a erros.

**Necessidades:** Ferramentas para registrar, buscar, atualizar e deletar informações de alunos e pagamentos com mensagens claras de validação.

#### 5.3 Gerente

**Características:** Responsável pela supervisão administrativa e financeira da academia.

**Desafios:** Falta de visibilidade sobre o status financeiro e dificuldades em gerenciar funcionários e planos.

**Necessidades:** Relatórios financeiros, gerenciamento de planos de assinatura e administração de dados de funcionários.

#### 5.4 Membro (Aluno)

**Características:** Cliente da academia que utiliza os serviços de treino.

**Desafios:** Acesso limitado a informações sobre seus treinos e status de pagamento.

**Necessidades:** Possibilidade de iniciar treinos vinculados a planos ativos e receber mensagens claras sobre status de pagamento.

## 5) Requisitos funcionais

### [RF001] Cadastro de Membro

Permite o cadastro de novos membros no sistema, garantindo que os dados sejam únicos e válidos.

**Ator:** Secretário

**Prioridade:** ☒ Essencial ☐ Importante ☐ Desejável

### [RF002] Atualização de Membro

Permite atualizar informações de membros existentes.

**Ator:** Secretário

**Prioridade:** ☐ Essencial ☒ Importante ☐ Desejável

### [RF003] Buscar Membro

Permite o cadastro de novos membros no sistema, garantindo que os dados sejam únicos e válidos.

**Ator: Secretário**

**Prioridade:** ☒ Essencial ☐ Importante ☐ Desejável

### [RF004] Exclusão de Membro

Permite deletar cadastros de membros.

**Ator: Secretário**

**Prioridade:** ☐ Essencial ☒ Importante ☐ Desejável

### [RF005] Registro de Treino

Permite criar treinos com exercícios específicos para membros.

**Ator: Professor**

**Prioridade:** ☒ Essencial ☐ Importante ☐ Desejável

### [RF006] Atualização de Treino

Permite modificar treinos existentes

**Ator: Professor**

**Prioridade:** ☐ Essencial ☒ Importante ☐ Desejável

### [RF007] Consulta de Treino

Permite visualizar treinos e seus detalhes.

**Ator: Professor**

**Prioridade:** ☒ Essencial ☐ Importante ☐ Desejável

### [RF008] Exclusão de Treino

Permite deletar treinos do sistema.

**Ator: Professor**

**Prioridade:** ☐ Essencial ☒ Importante ☐ Desejável

### [RF009] Registro de Exercício

Permite cadastrar novos exercícios no sistema.

**Ator: Professor**

**Prioridade:** ☒ Essencial ☐ Importante ☐ Desejável

### [RF010] Atualização de Exercício

Permite atualizar informações de membros existentes.

**Ator: Professor**

**Prioridade:** ☐ Essencial ☒ Importante ☐ Desejável

### [RF011] Consulta de Exercício

Permite buscar exercícios disponíveis.

**Ator: Professor**

**Prioridade:** ☒ Essencial ☐ Importante ☐ Desejável

### [RF012] Exclusão de Exercício

Permite deletar exercícios do sistema.

**Ator: Secretário**

**Prioridade:** ☐ Essencial ☒ Importante ☐ Desejável

### [RF013] Visualização de Histórico de Treinos

Permite consultar o histórico de treinos de um membro.

**Ator: Professor**

**Prioridade:** ☐ Essencial ☒ Importante ☐ Desejável

### [RF014] Registro de Pagamentos

Permite registrar pagamentos de mensalidades.

**Ator: Secretário**

**Prioridade:** ☒ Essencial ☐ Importante ☐ Desejável

### [RF015] Visualização de Pagamentos

Permite consultar o status e histórico de pagamentos.

**Ator: Gerente**

**Prioridade:** ☒ Essencial ☐ Importante ☐ Desejável

### [RF016] Criação do Plano

Permite criar novos planos de assinatura.

**Ator: Gerente**

**Prioridade:** ☒ Essencial ☐ Importante ☐ Desejável

### [RF017] Atualização do Plano

Permite modificar planos existentes.

**Ator: Gerente**

**Prioridade:** ☐ Essencial ☒ Importante ☐ Desejável

### [RF018] Exclusão do Plano

Permite deletar planos do sistema.

**Ator: Gerente**

**Prioridade:** ☐ Essencial ☒ Importante ☐ Desejável

### [RF019] Gerenciamento dos Funcionários

Permite registrar novos funcionários.

**Ator: Gerente**

**Prioridade:** ☒ Essencial ☐ Importante ☐ Desejável

### [RF020] Atualização de Funcionário

Permite modificar dados de funcionários

**Ator: Gerente**

**Prioridade:** ☐ Essencial ☒ Importante ☐ Desejável

### [RF021] Exclusão de Funcionário

Permite deletar dados de funcionários.

**Ator: Gerente**

**Prioridade:** ☐ Essencial ☒ Importante ☐ Desejável

## 6) Casos de Uso

<Utilize este espaço para apresentar os diagramas de caso de uso com a descrição das características e fluxos dos casos de uso.>

#### [USO001] Cadastro de Membro

O secretário realiza o cadastro de um novo membro, inserindo informações como CPF (único), plano ativo e valor do plano.

#### [USO002] Registro de Pagamento

O secretário registra o pagamento de uma mensalidade, incluindo valor, status e possíveis descontos/multas. O sistema libera o acesso ao treino após o pagamento.



### [USO003] Gerenciamento de Treino

O professor cria, atualiza, consulta ou deleta treinos, associando exercícios específicos a membros.  
Fluxo Principal

## 7) Requisitos não funcionais

### a) Usabilidade

#### [NF001] Interface Intuitiva

O sistema deve ter uma interface clara e amigável, com navegação simples para todos os usuários.

**Prioridade:** ☒ Essencial ☐ Importante ☐ Desejável

[USO001], [USO002], [USO003]

#### [NF002] Mensagens Claras

O sistema deve exibir mensagens de erro e sucesso claras, como especificado nos casos de uso.

**Prioridade:** ☐ Essencial ☒ Importante ☐ Desejável

[USO001], [USO002], [USO003]

### b) Confiabilidade

#### [NF003] Backup Diário

Dados devem ser backupados automaticamente todo dia às 2h.

**Prioridade:** ☒ Essencial ☐ Importante ☐ Desejável

### c) Desempenho

<Esta seção descreve os requisitos não funcionais associados à eficiência, uso de recursos e tempo de resposta do sistema.>

#### [NF004] Tempo de Resposta

Operações críticas tais como o login, ou pagamentos devem demorar no máximo 3 segundos.

**Prioridade:** ☐ Essencial ☒ Importante ☐ Desejável

### d) Segurança

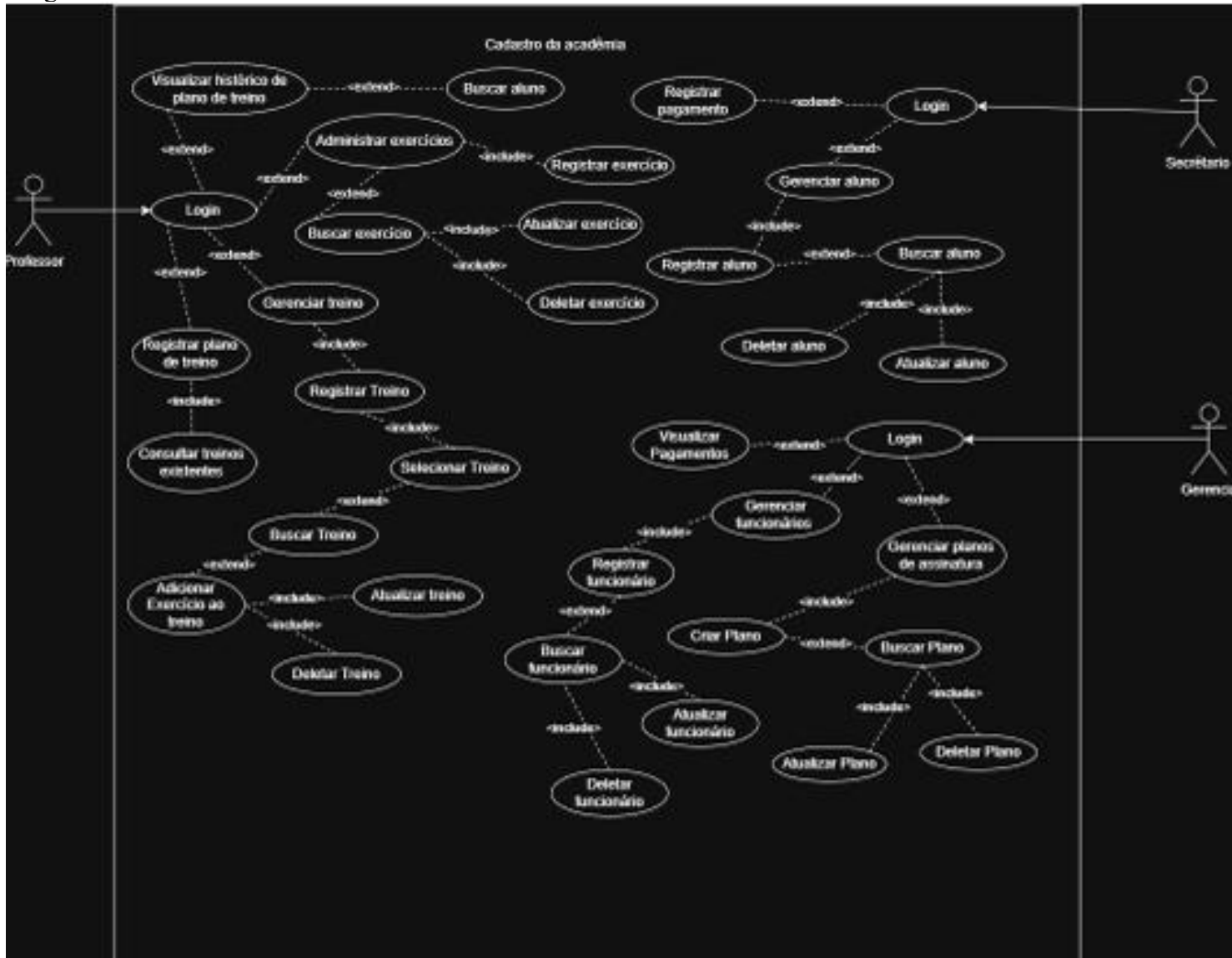
#### [NF005] Encapsulamento de informações

As informações dos Usuários e do código devem ser encapsuladas para dificultar ataques hackers.

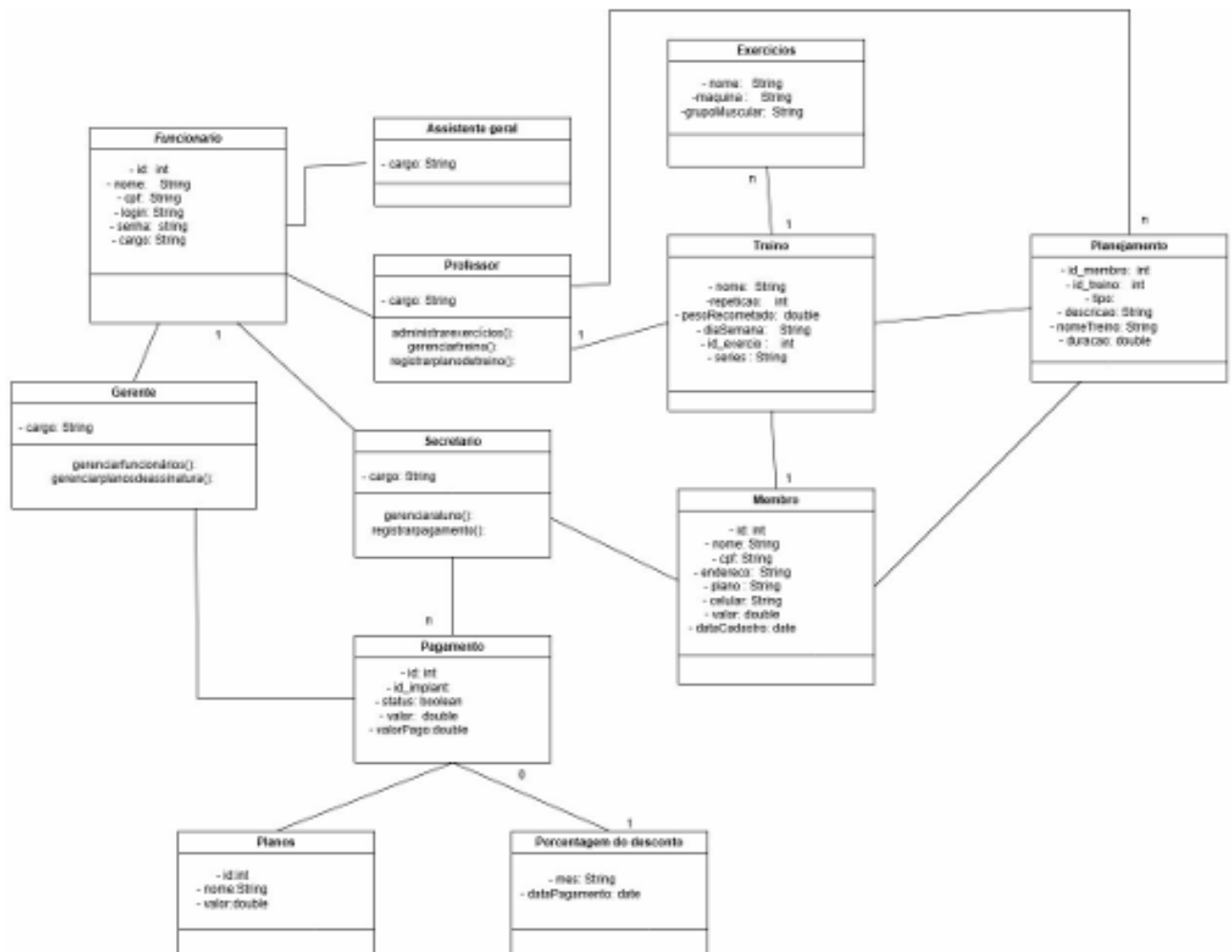
**Prioridade:** ☒ Essencial ☐ Importante ☐ Desejável

## 8) Artefatos UML

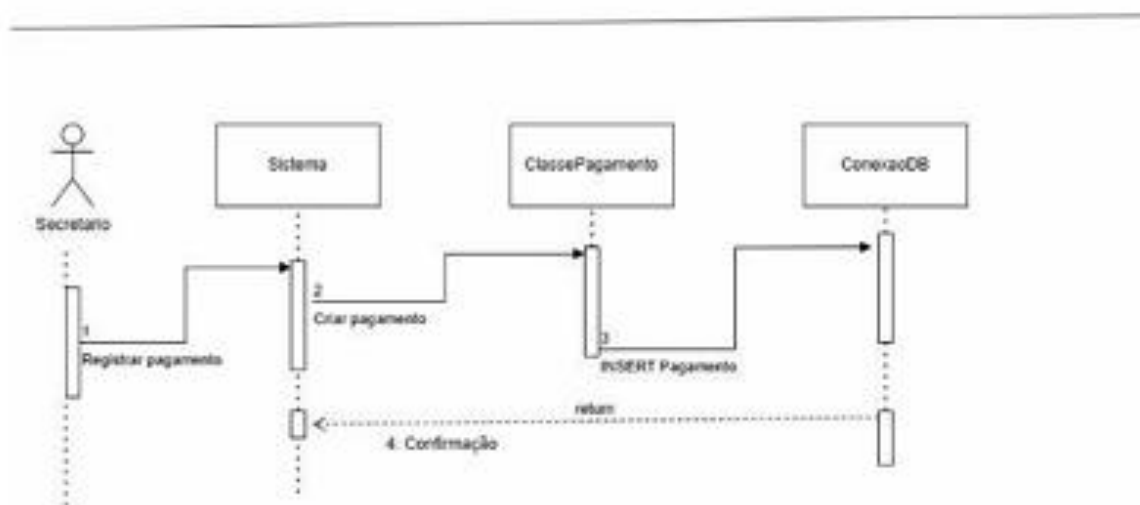
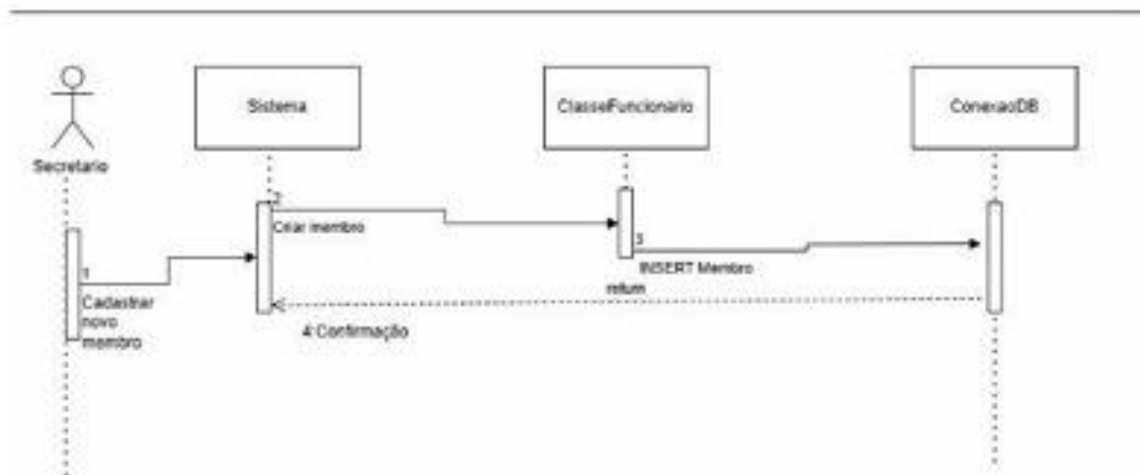
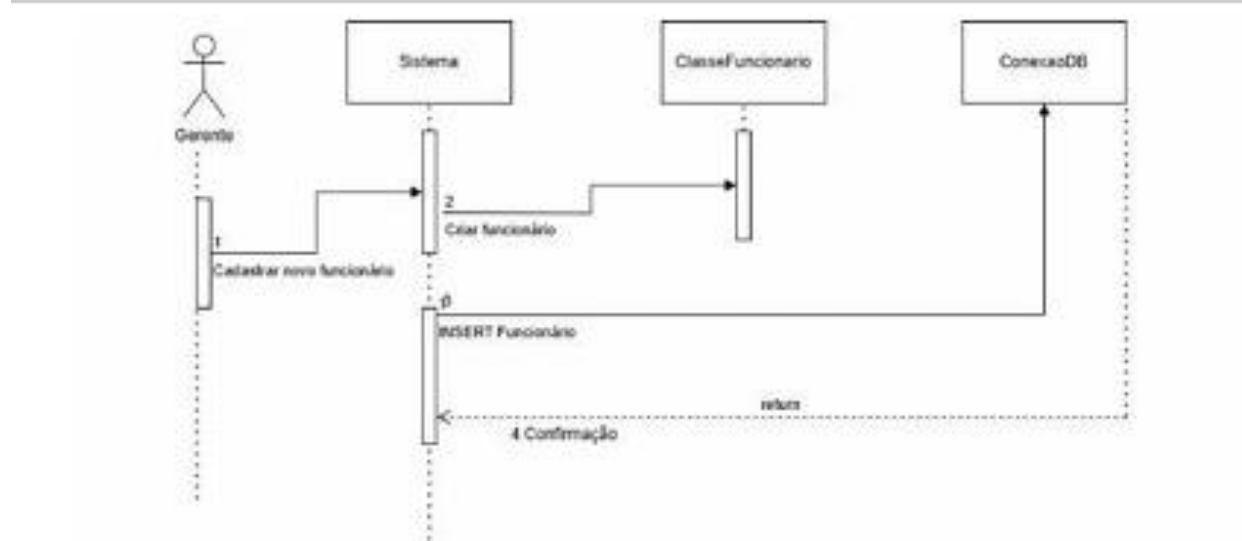
### Diagrama de Caso De Uso:



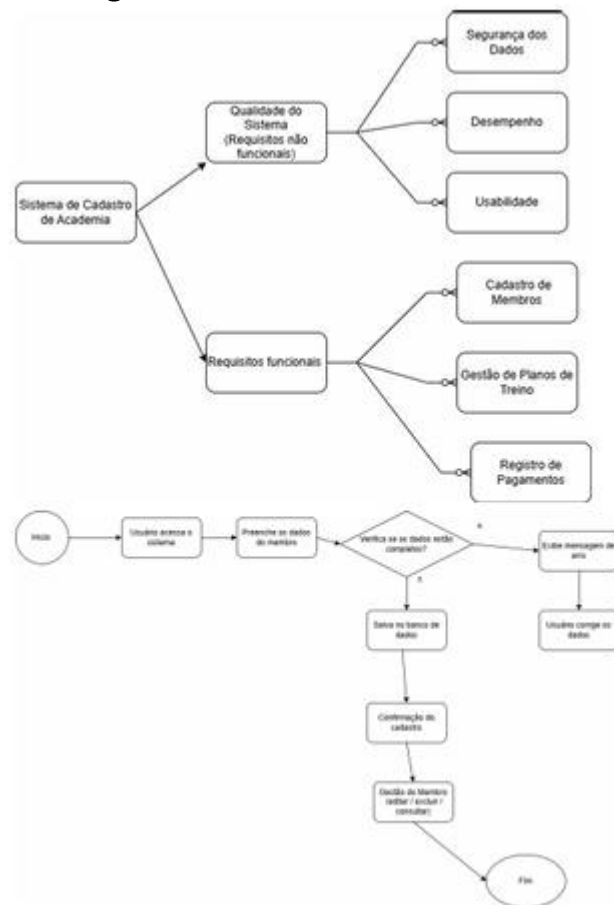
## Diagrama de Classe



## Diagrama de Sequência:



## Outros Diagramas:



## 9) Banco de Dados

No banco de dados da Academia Javasfit é utilizado 2 relacionamentos N para , abordados como ‘*membro\_planos*’ e ‘*exercício\_treinos*’. E um relacionamento 1:N, abordado como Pagamentos, em que um Membro(Aluno) pode ter vários pagamentos.

### Relacionamento de tabelas

<Apresente o diagrama lógico do banco de dados com os relacionamentos das tabelas do sistema>

## 10) Descrição da interface com o usuário

### [INT001] Interface de Login

O usuário irá poder ver uma pequena interface com o botão de Entrar no sistema, desde que tenha um Login e Senha registrado no banco de dados.

### Críticas da interface

A interface não é convidativa, tão pouco visualmente agradável, é apenas focada na funcionalidade

### [INT002] Interface para Usuário

Essa Interface é simples, com os botões referentes aos casos de Uso que o Usuário, funcionários, irá utilizar, essa interface 'mudará' dependendo do cargo do Usuário.

### Críticas da interface

A interface não é agradável visualmente, mas é rápida e completa.

### [INT003] Interface para CRUD

Essa Interface é mais completa, com auxílio de 'comboboxes' alteráveis, dependendo do cargo, trazendo uma dinâmica muito maior para o sistema, melhorando o seu fluxo e agilidade de uso.

### Críticas da interface

A interface é rápida, adaptada ao banco de dados.

## 11) CONCLUSÕES E RECOMENDAÇÕES

Assim como uma lógica complexa deve ser dividida em pequenas etapas, é extremamente necessário que uma equipe se divida em pequenas partes, com cada parte responsável por uma parte do projeto, quer seja no código, na documentação, no banco de dados, e/ ou na interface, é recomendável que todos façam suas pequenas 'tasks', mas façam isso pensando no outro, pois assim o código e o programa não se tornará um "Frankstein bizarro", mas uma arte feita por diversos programadores, e por isso é tão importante o uso dos Diagramas, para que toda a equipe esteja em sintonia e entenda o que o outro participante precisa de auxílio na sua task. Além do mais, utilizar Inteligência Artificial, de forma imprudente, pode deixar o seu código muito travado e menos responsivo, e isso não é recomendado.

## 12) REFERÊNCIAS

Oracle Java Foudations, Oracle Database Design, Oracle Database