

Documento de Especificação de Projeto

Projeto de Software

*Disciplina Análise e Projeto Orientado à Objetos – Terceiro Semestre
Bacharelado em Ciência da Computação
Professor Lúcio Geronimo Valentin
UTFPR – Universidade Tecnológica Federal do Paraná – Campus Campo Mourão*

Projeto: *GymAdmin*

Equipe:

Diogo Rodrigues dos Santos <diogo.2003@alunos.utfpr.edu.br>

Vitor Hugo Melo Ribeiro <vitorhugoribeiro@alunos.utfpr.edu.br>

1. [Introdução](#)
2. [Descrição do sistema](#)
 - a. [Funções do produto.](#)
 - b. [Características do Usuário](#)
 - c. [Suposições e Dependências](#)
3. [Objetivo do Sistema](#)
4. [Convenções, termos e abreviações](#)
 - a. [Identificação dos casos de uso](#)
 - b. [Prioridades dos casos de uso](#)
5. [Lista de Requisitos](#)
6. [Requisitos Funcionais.](#)
 - a. [_____ CADASTRO DE ALUNO](#)
 - b. [MATRÍCULA](#)
 - c. [TURMAS](#)
 - d. [AVISOS](#)
 - e. [PLANOS](#)
7. [Requisitos Não Funcionais](#)
8. [Atores](#)
9. [Casos de Uso](#)

- a. [\[UC01\] Criar Turma](#)
 - b. [\[UC02\] Fazer Contrato](#)
 - c. [\[UC03\] Matricular](#)
 - d. [\[UC04\] Desmatricular](#)
 - e. [\[UC05\] Cancelar Turma](#)
 - f. [\[UC06\] Editar Turma](#)
 - g. [\[UC07\] Gerenciar Aluno](#)
 - h. [\[UC08\] Gerenciar Professor](#)
10. [Análise Textual](#)
 11. [Modelo Conceitual](#)
 12. [Sequência](#)
 13. [Contrato de operações](#)
 14. [Diagrama de comunicação](#)
 15. [Diagrama de classe](#)
 16. [Referências](#)

Introdução

Este documento tem como objetivo especificar os requisitos e funções do sistema GymAdmin. Este documento contém: Documento de Requisitos, Diagrama de Casos de Uso, Descrição Textual dos Casos de Uso, Modelo Conceitual, Diagrama de Sequência do Sistema, Diagrama de Comunicação e Diagrama de Classe.

Descrição do sistema

Neste projeto é apresentado um sistema de gerenciamento de turmas, alunos e funcionários de uma academia. Conectando as turmas que podem ter diversas modalidades (musculação, zumba, crossfit e outras) com os alunos que desejam se matricular nelas.

Também é possível fazer o cadastro de alunos e funcionários, criação de turmas e organização.

Funções do produto.

- Exibir lista de alunos em determinada turma.
- Exibir informações dos alunos.
- Permitir consulta de turmas abertas.
- Permitir visualizar alunos na academia em determinado horário.
- Permitir matrícula e cancelamento de matrícula para alunos.
- Permitir a visualização do histórico de presença de cada aluno.
- Permitir imprimir e reimprimir comprovante de pagamento.
- Visualizar histórico de pagamento da mensalidade.

- Permitir a substituição de professores.
- Permitir fazer contratos com professores.
- Exibir lista de professores e quais turmas ele está dando aula.

Características do Usuário

O sistema terá dois tipos de usuários, sendo eles, alunos e funcionários administrativos. Os funcionários administrativos terão acesso através de uma versão web. Os alunos terão acesso via app.

Suposições e Dependências

O sistema deverá funcionar em linux e windows.

O aluno acessar o sistema através de um app(android e ios)

Os demais usuários acessarão o sistema através de um navegador web.

Objetivo do Sistema

O objetivo do sistema é facilitar e automatizar o processo de gerenciamento e controle da academia.(adicionar o que se perdeu)

Convenções, termos e abreviações

A correta interpretação deste documento exige o conhecimento de algumas convenções, termos específicos e abreviações, que são descritos a seguir.

Identificação dos casos de uso

Os casos de uso devem ser identificados com um identificador único. A numeração inicia com o identificador [UC01] e prossegue sendo incrementada à medida que forem surgindo novos casos de uso.

A nomenclatura dos fluxos secundários dos casos de uso é dada por uma sigla e por um número. A sigla deve ser FA para fluxos alternativos e FE para fluxos de erro. O número é uma sequência que inicia de 01. Um exemplo de fluxo alternativo é [FA01] e um exemplo de fluxo de erro é [FE01]. O número do identificador reinicia a cada caso de uso.

Para referenciar casos de uso em qualquer local do documento, o identificador do caso de uso é utilizado. Por exemplo, a pré-condição de um caso de uso poderia conter o seguinte texto: “Este caso de uso demanda que o usuário da aplicação esteja logado, como descrito em [UC12]”. Para referenciar um fluxo secundário fora do caso de uso que o define, é necessário utilizar o identificador do caso de uso concatenado com um ponto e com o identificador do fluxo. Por exemplo, a descrição de um caso de uso poderia conter o seguinte texto: “Este caso de uso permite que o usuário edite informações avançadas do seu perfil e é disparado quando o usuário clica no botão **Informações Avançadas** durante seu cadastro, conforme descrito no fluxo alternativo [UC01].[FA02]”.

Prioridades dos casos de uso

A prioridade de cada caso de uso, informada nas seções 3 e 4, pode ser classificada como “essencial”, “importante” e “desejável”, de acordo com a descrição abaixo:

- Um caso de uso **essencial**, se não for atendido, impede que a aplicação entre em funcionamento. Casos de uso essenciais são imprescindíveis, isto é, têm de ser implementados impreterivelmente.
- Se um caso de uso **importante** não for atendido, a aplicação pode até entrar em funcionamento, mas de forma não-satisfatória. Casos de uso importantes deveriam ser implementados, mas, se não forem, não impedirão a implantação e utilização da aplicação.
- Um caso de **desejável**, por fim, é aquele cuja ausência de implementação não compromete a operacionalização da aplicação, isto é, a aplicação pode funcionar de forma satisfatória mesmo sem sua implementação. Esses casos de uso podem ser deixados para versões posteriores da solução, caso não haja tempo hábil para implementá-los na versão que está sendo especificada.

Lista de Requisitos

Requisitos Funcionais.

CADASTRO DE ALUNO

RF1. O sistema deve conseguir inserir e remover alunos, cada aluno possui: nome, CPF, endereço, ficha de avaliação médica

MATRÍCULA

RF2. O sistema deve evitar conflito de horários no ato de matrícula de um aluno.

RF3. O sistema deve bloquear o acesso do aluno caso suas mensalidades não estejam em dia.

RF4. O sistema deve impedir a matrícula de alunos que não possuam cadastro completo.

TURMAS

RF5. Uma turma não poderá exceder o número de 20 alunos.

RF6. Cada turma possuía no máximo 2 professores.

RF7. Só poderão ser criadas turmas com modalidades que possuam professores disponíveis.

RF8. Uma turma deve possuir uma data de abertura e uma data de término.

AVISOS

RF9. Os alunos deverão receber uma notificação avisando que não haverá aula. Isso ocorrerá quando o professor registrar no sistema que não irá comparecer na aula.

RF10. O sistema deverá notificar os alunos quando suas mensalidades estiverem próximas de expirar.

PLANOS

RF11. O sistema deve ser capaz de alterar os valores dos planos quando solicitado, mas os alunos que entraram antes dessa alteração deverão continuar com o valor do plano antigo.

Requisitos Não Funcionais

RNF1. O sistema deve possuir cores que também são usadas pela academia.

RNF2. O sistema deve se integrar com leitores de biometria.

Atores

A tabela abaixo descreve brevemente cada ator da aplicação.

Ator	Descrição
Gerente	O gerente faz o cadastro de funcionários, gerência modalidade e realiza os contratos.
Atendente	O atendente é quem faz o cadastro de alunos, de turmas, modalidades e normalmente também acessa o sistema para pegar informações para o cliente.
Professor	O professor poderá fazer o registro de alunos nas turmas e a criação de cronograma de treinos.
Aluno	O aluno poderá acessar seu cronograma, ver horários e informações sobre matrículas.

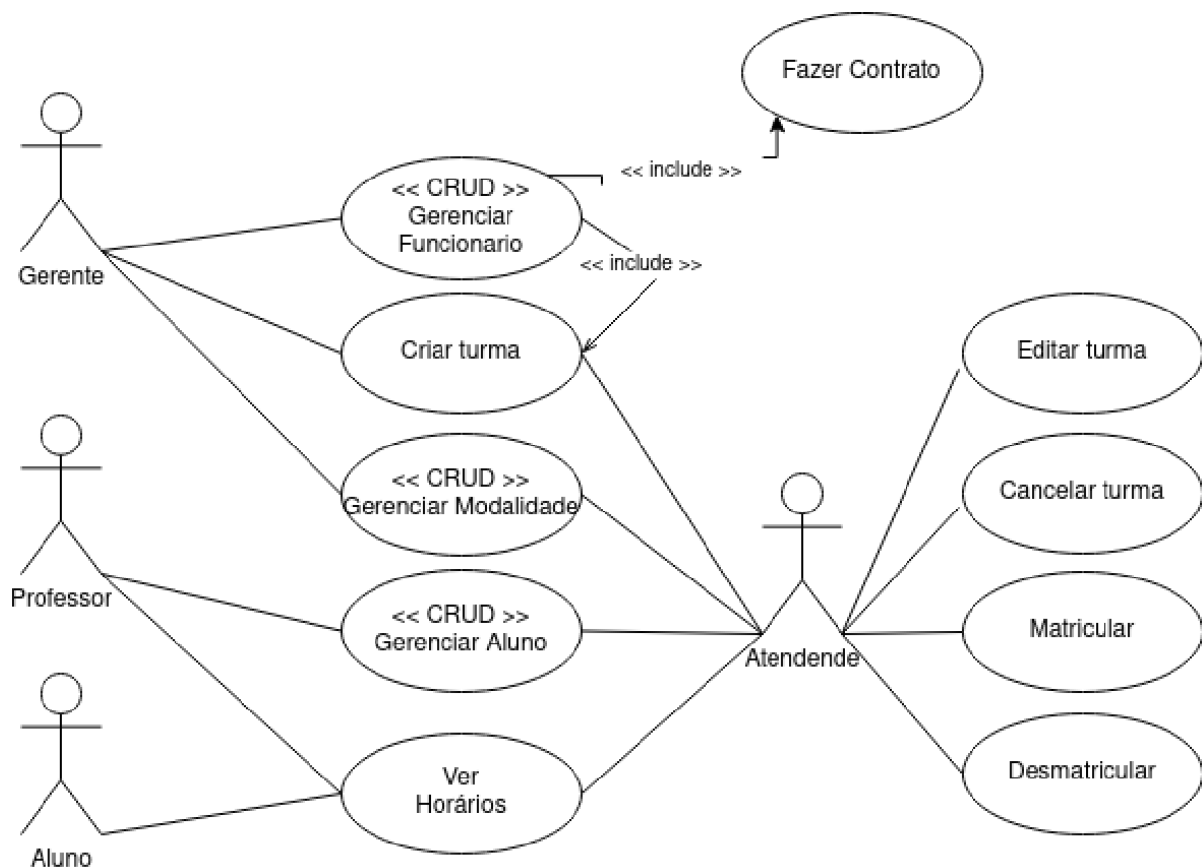
Casos de Uso

Na linguagem de modelagem unificada (UML), o diagrama de caso de uso resume os detalhes dos usuários do seu sistema (também conhecidos como atores) e as interações deles com o sistema. Para criar um, use um conjunto de símbolos e conectores especializados.

Referencia: https://www.lucidchart.com/pages/pt/diagrama-de-caso-de-uso-uml#section_0

Diagrama de Caso de Uso - Gym Admin

Essa seção apresenta todos os requisitos funcionais da aplicação, especificados como casos de uso.



[UC01] Criar Turma

Prioridade:	<input type="checkbox"/> Essencial	<input checked="" type="checkbox"/> Importante	<input type="checkbox"/> Desejável
Ator(es):	Gerente, Atendente		

Descrição: A criação de turma é feita quando as turmas de determinada modalidade estão cheias ou por decisão administrativas.

Pré-condições: É necessário que ao menos uma modalidade esteja disponível.

Pós-condições: O número de alunos matriculados na turma deve ser atualizado sempre que houver uma nova matrícula.

Fluxo principal

1. O gerente ou atendente clica em abrir turma.
2. O sistema exibe as modalidades disponíveis.
3. A modalidade desejada é selecionada.
4. O sistema pede que seja informado os horários.
5. O sistema exibe os professores disponíveis para aquela turma.
6. Um contrato é iniciado com o professor escolhido.
7. O sistema solicita a data de início e fim da turma.
8. O sistema adiciona o id da turma na lista de turma abertas.
9. O sistema informa que a criação foi concluída com sucesso.

Fluxos Alternativos

[FA 001]

2.1 Não existe modalidades disponíveis

2.2 O sistema é encerrado.

[UC02] Fazer Contrato

Prioridade:	<input checked="" type="checkbox"/>	Essencial	<input type="checkbox"/>	Importante	<input type="checkbox"/>	Desejável
Ator(es):	Gerente, Atendente					

Descrição: O contrato é realizado quando um professor deseja assumir uma turma.

Pré-condições: O cadastro do professor já deve existir.

Pós-condições: O professor deve aparecer como disponível na modalidade do contrato

Fluxo Principal

1. O atendente ou gerente clica em iniciar uma contratação.
2. O sistema pede para que seja informado o CPF do professor a ser contratado.
3. O sistema verifica se existe contrato aberto para aquele cpf e retorna eles caso encontre.
4. Ao iniciar um novo contrato o sistema pede que seja informado o período deste contrato.
5. O atendente ou gerente informa quais modalidades o novo professor pode ministrar.
6. O sistema atualiza a lista de professores da modalidade.
7. O processo de contratação é finalizado.

Fluxos Alternativos

[FA 001]

2.1 Caso o CPF seja invalido, o sistema pede para que seja informado novamente.

2.2 O atendente inserir novamente.

2.3 O sistema continua no caso de uso 3.

[UC03] Matricular

Prioridade:	<input checked="" type="checkbox"/>	Essencial	<input type="checkbox"/>	Importante	<input type="checkbox"/>	Desejável
Ator(es):	Atendente					

Descrição: A matrícula é feita para controlar quais atividades o aluno participa na academia e, para controle das turmas.

Pré-condições: O cadastro do aluno já deve existir. Ao menos uma turma deve estar aberta

Pós-condições: A turma deve aparecer na lista de matrículas do aluno.

Fluxo Principal

1. O atendente clica em abrir uma turma.
2. O sistema exibe uma lista de modalidades disponíveis.
3. O atendente seleciona a modalidade desejada.
4. O sistema verifica as turmas abertas para a modalidade escolhida.

5. O atendente seleciona a turma desejada.
6. O sistema insere o aluno na turma e retorna o id da turma para inserir na matrícula.
7. O atendente clicar em confirmar a matrícula.
8. O sistema exibe uma mensagem confirmando que a matrícula foi realizada com sucesso.

Fluxos Alternativos

[FA 001]

2.1 Modalidades indisponíveis.

2.1 Encerra o sistema e avisa que não existe modalidades disponíveis

[UC04] Desmatricular

Prioridade:	<input checked="" type="checkbox"/>	Essencial	<input type="checkbox"/>	Importante	<input type="checkbox"/>	Desejável
Ator(es):	Atendente					

Descrição: O atendente recebe a solicitação de desmatricular um aluno, seja pelo próprio aluno ou por algum motivo maior. No sistema, o atendente clica no botão “Desmatricular”, logo, o sistema retorna um input label para que seja colocado o cpf do aluno, se o aluno estiver matriculado em alguma turma, retornará uma lista com essas turmas, e então o atendente clica na turma que deseja executar a desmatrícula, aparecendo um modal de confirmação.

Pré-condições: O aluno precisa estar matriculado.

Pós-condições: O aluno estará elegível a se matricular na mesma turma futuramente.

Fluxo Principal:

1. No painel de controle, o atendente clica em ‘alunos’.
2. É exibido uma lista com os alunos ativos.
3. O atendente pode procurar o aluno pelo cpf.
4. Clicando no nome do aluno, abrirá uma tela com as informações do aluno.
5. Após isso, haverá um botão desmatricular.
6. Clicando no botão, vai aparecer uma lista de turmas que o aluno está matriculado.
7. Clica na turma e confirma.

[UC05] Cancelar Turma

Prioridade:	<input checked="" type="checkbox"/>	Essencial	<input type="checkbox"/>	Importante	<input type="checkbox"/>	Desejável
Ator(es):	Atendente					

Descrição: Se por algum motivo externo a turma precisar ser cancelada por falta de professor ou alunos.

Pré-condições: A turma já existe.

Pós-condições: O aluno estará elegível a se matricular na mesma turma futuramente.

Fluxo Principal:

1. No painel de controle, o atendente clica em cancelar a turma.
2. É exibido uma lista com as turmas existentes.
3. O atendente procura a turma digitando o código da turma.
4. Clicando na turma um modal será aberto pedindo a confirmação.
5. Após confirmar, a turma será 'movida' para o estado de cancelada.

[UC06] Editar Turma

Prioridade:	<input checked="" type="checkbox"/>	Essencial	<input type="checkbox"/>	Importante	<input type="checkbox"/>	Desejável
Ator(es):	Atendente					

Descrição: Quando uma turma precisar ser editada, como: horários, professor, local, etc... Esta função será utilizada.

Pré-condições: A turma deve existir.

Pós-condições: O a turma será editada e as informações devem ser disponíveis no próprio painel

Fluxo Principal:

1. No painel de controle, o atendente clica em editar a turma.
2. É exibido uma lista com as turmas existentes.
3. O atendente procura a turma digitando o código da turma.
4. Clicando na turma, aparecerá uma tela com as informações da turma e seus campos de edição.
5. Basta editar o que for necessário.

[UC07] Gerenciar Aluno

Prioridade:	<input checked="" type="checkbox"/>	Essencial	<input type="checkbox"/>	Importante	<input type="checkbox"/>	Desejável
Ator(es):	Atendente					

Descrição: Caso de uso utilizado para criação, remoção, e atualização dos cadastros de aluno.

Pré-condições: CPF deve ser válido

Pós-condições: O aluno deve estar apto para matrícula.

Fluxo Principal:

1. O atendente informa o CPF do aluno.
2. O sistema verifica se existe algum cadastro para o CPF e, caso não exista, pede para preencher os campos de informações.
3. O atendente informa os dados do aluno (nome, endereço, plano escolhido).
4. O sistema valida os campos e informa sucesso.

Fluxos Alternativos

[FA 001]

- 1.1 Caso o CPF seja invalido, o sistema pede para que seja informado novamente.
- 1.2 O atendente inserir novamente.
- 1.3 O sistema continua no caso de uso 3.

[UC08] Gerenciar Professor

Prioridade:	<input checked="" type="checkbox"/>	Essencial	<input type="checkbox"/>	Importante	<input type="checkbox"/>	Desejável
Ator(es):	Atendente					

Descrição: Caso de uso utilizado para criação, remoção, e atualização dos cadastros de professores.

Pré-condições: CPF deve ser válido

Pós-condições: O professor deve estar disponível para inserção nas turmas.

Fluxo Principal:

1. O atendente informa o cpf do professor.
2. O sistema valida e pede para informar os dados(nome, formações, .
- 3.

Análise Textual

Identificador do Caso de Uso	Substantivos encontrados	Verbos Encontrados
[UC01]	Gerente, atendente, turma, sistema, modalidades, professores.	solicita, adiciona, informa.
[UC02]	atendente, gerente, contrato, sistema, professor, cadastro	iniciar, preencher
[UC03]	Atendente, sistema, modalidade.	Abrir, exibir, clicar, informar.
[UC04]	selecionado, valor, campo	Iniciar, editar, confirmar
[UC05]	desmatricula	iniciar, desmatricular
[UC06]	turma	iniciar, cancelar

Modelo Conceitual

Nesta seção, será apresentado o diagrama de classes da aplicação. A visão escolhida para o diagrama de classes é focada na persistência dos objetos, primando por um projeto flexível e de fácil extensão/manutenção.

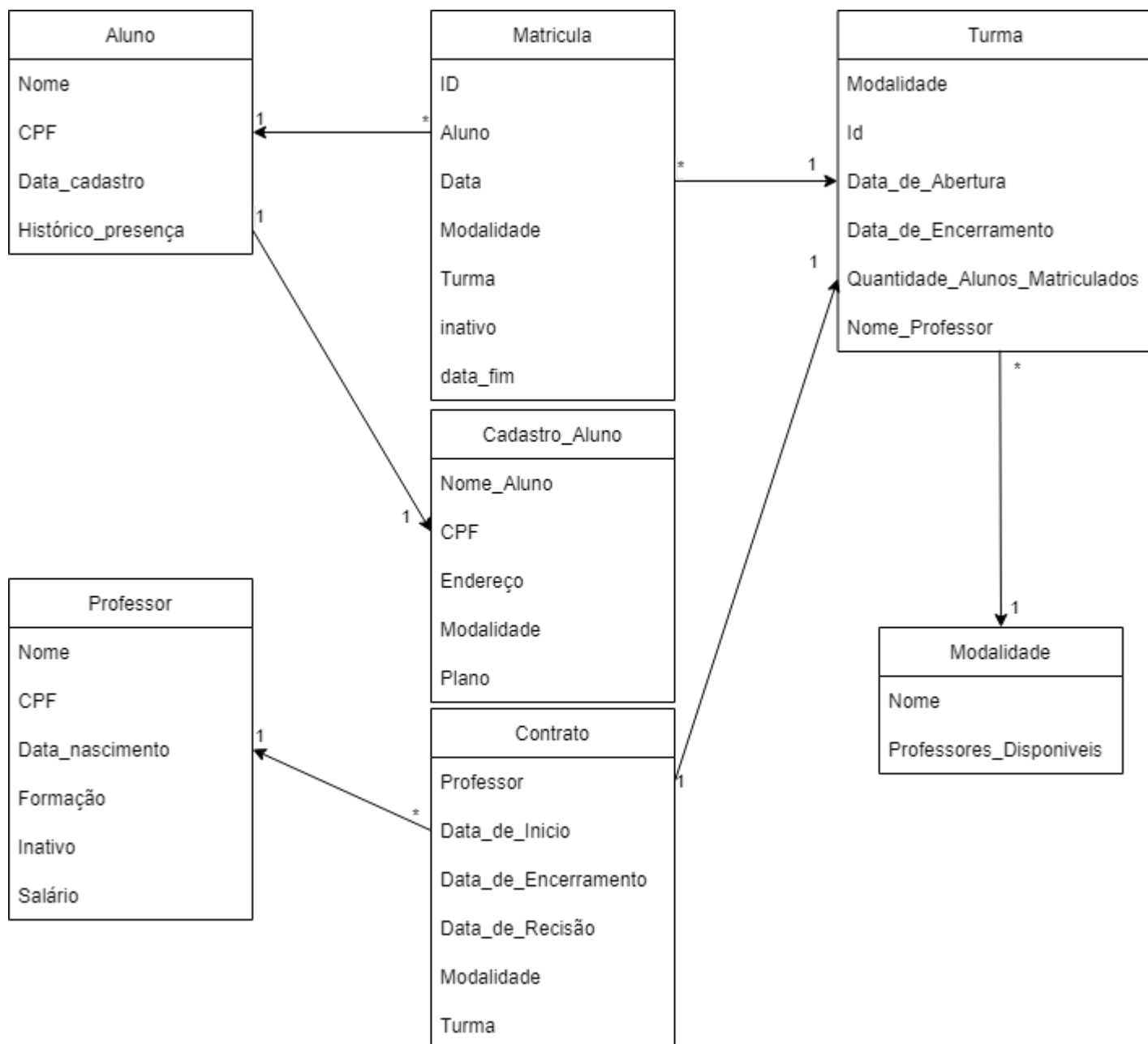
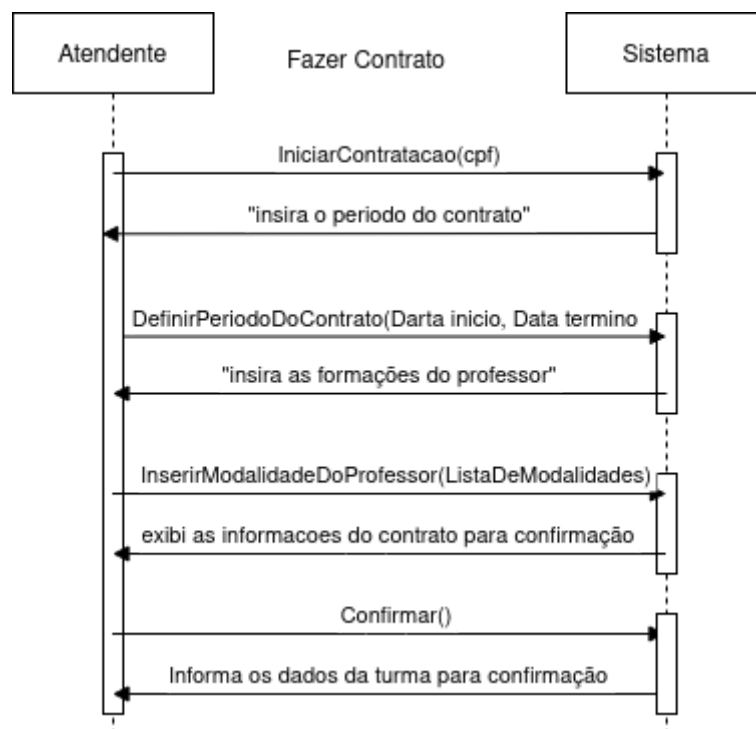
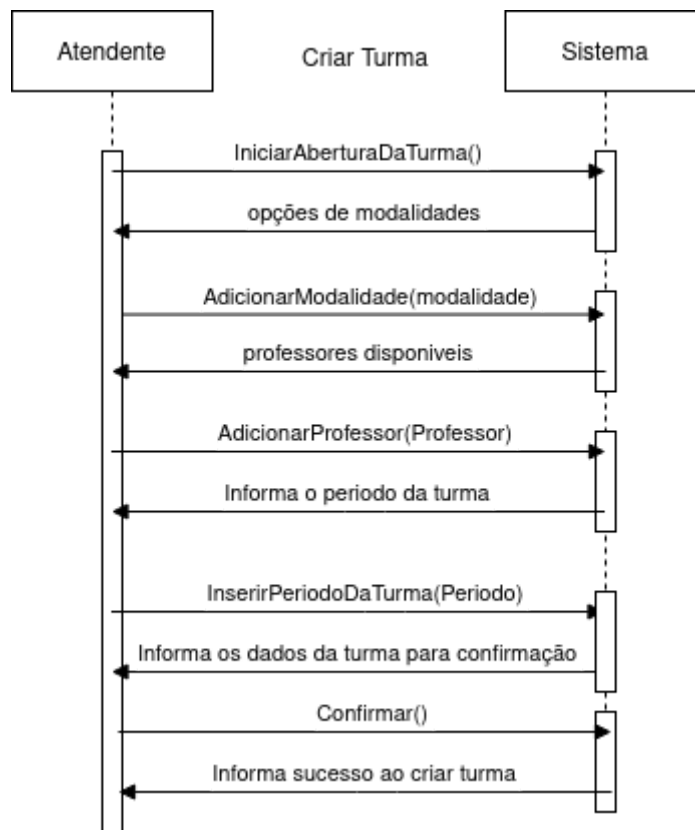
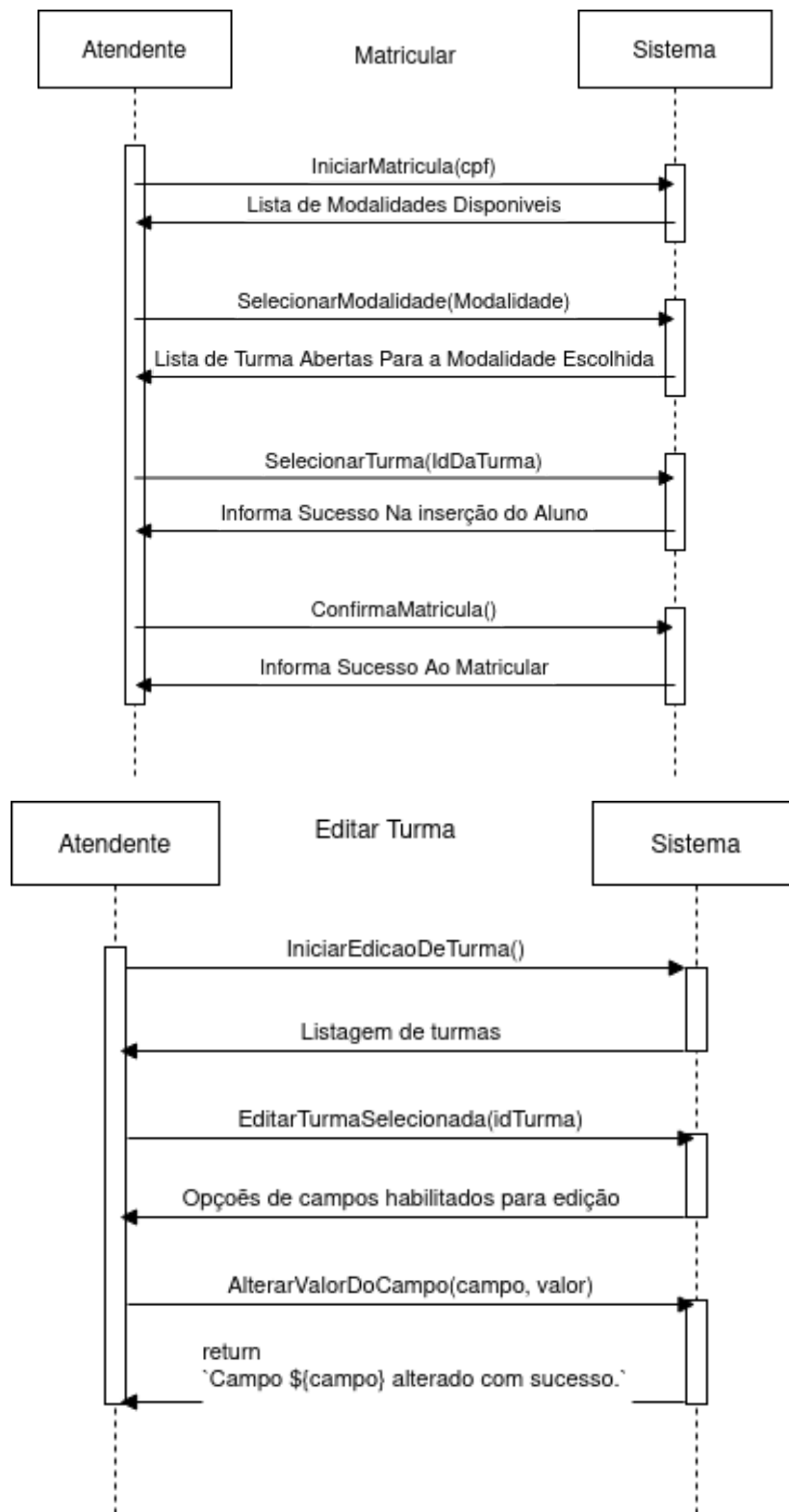
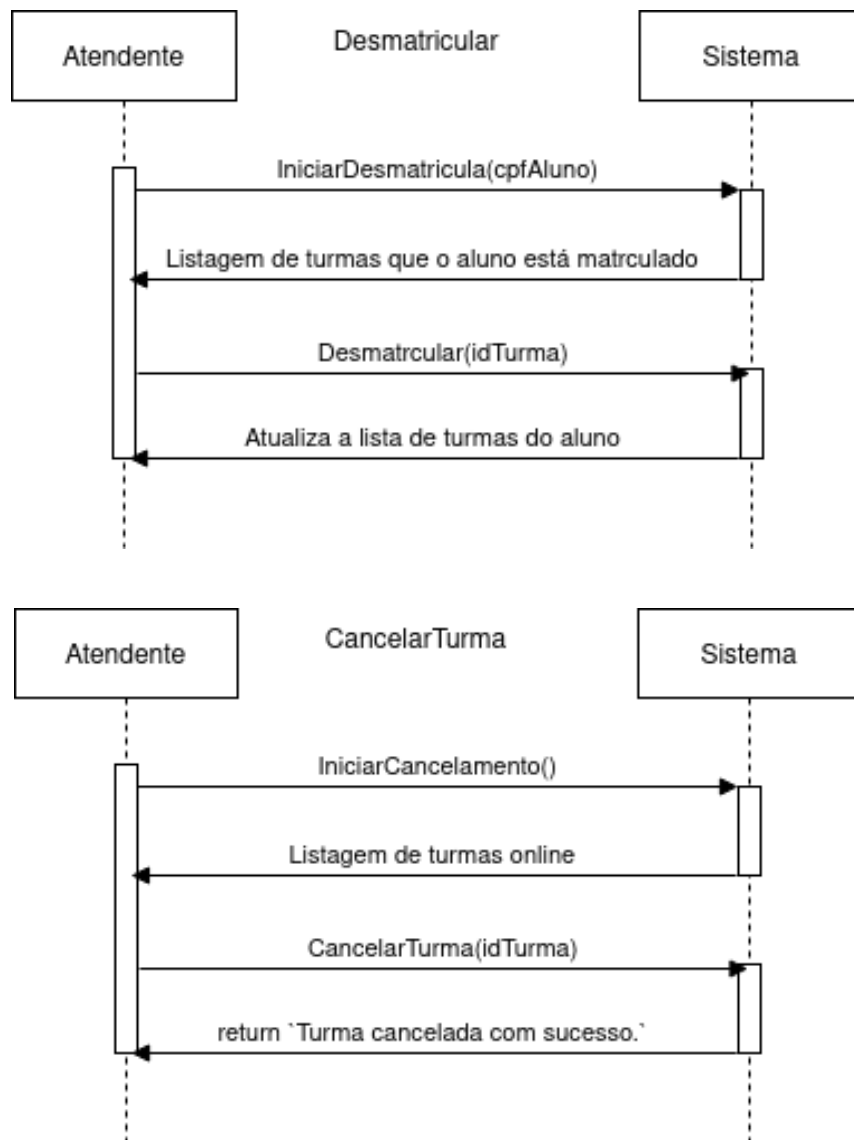


Diagrama de Sequência







Contrato de Operações

Operação: Matrícula Iniciar.

Parâmetros: CPF do Aluno

Referências cruzadas: Matricular.

Pré Condições: Aluno deve estar cadastrado no sistema.

Pós Condições: A matrícula com o CPF indicado deve estar disponível.

Operação: Iniciar Abertura Da Turma.

Parâmetros:

Referências cruzadas: Criar Turma.

Pré Condições: Devem existir Modalidades Disponíveis.

Pós Condições: A lista de turmas abertas deve estar atualizada e os dados das novas turmas devem estar guardados.

Operação: Adicionar Modalidade.

Parâmetros: Modalidade.

Referências cruzadas: Criar Turma.

Pré Condições: Devem existir modalidade disponíveis

Pós Condições: O sistema deve verificar se a modalidade escolhida ainda estará disponível.

Operação: Iniciar Contratação.

Parâmetros: cpf do professor

Referências cruzadas: Fazer Contrato, Criar Turma

Pré Condições: o cpf deve ser válido.

Pós Condições: Um novo contrato deve existir no sistema.

Operação: Inserir Período Turma.

Parâmetros: Data de início e fim da turma.

Referências cruzadas: Criar Turma.

Pré Condições: Data inicial deve ser superior a atual.

Pós Condições: Os períodos da turma devem estar validados.

Operação: Inserir Período Do Contrato.

Parâmetros: Data de início e fim do contrato

Referências cruzadas: Fazer Contrato.

Pré Condições: As datas devem ser superiores a atual.

Pós Condições: Os períodos inseridos devem estar validados.

Operação: Inserir Modalidade Professor.

Parâmetros: Lista de Modalidade.

Referências cruzadas: Fazer Contrato.

Pré Condições: as modalidades devem ser válidas.

Pós Condições: A lista com modalidade disponível devem ser atualizadas caso seja inserido novas modalidades.

Operação: Iniciar Matrícula.

Parâmetros: CPF.

Referências cruzadas: Matricular.

Pré Condições: CPF deve ser válido.

Pós Condições:

Operação: Selecionar Modalidade.

Parâmetros: Modalidade.

Referências cruzadas: Matricular.

Pré Condições:

Pós Condições:

Operação: Selecionar Turma.

Parâmetros: Id da Turma.

Referências cruzadas: Matricular.

Pré Condições:

Pós Condições:

Operação: Confirmar Matrícula.

Parâmetros:

Referências cruzadas: Matricular.

Pré Condições:

Pós Condições:

Operação: Iniciar Desmatrícula.

Parâmetros:

Referências cruzadas: desmatricular.

Pré Condições: O aluno precisa estar matriculado em pelo menos uma turma.

Pós Condições:

Operação: Remove aluno.

Parâmetros: cpf, id da turma.

Referências cruzadas: desmatricular.

Pré Condições: O aluno precisa estar matriculado em pelo menos uma turma.

Pós Condições:

Operação: Iniciar edição.

Parâmetros:

Referências cruzadas: Editar turma.

Pré Condições: A turma precisa estar criada.

Pós Condições:

Operação: Editar.

Parâmetros: Id da turma.

Referências cruzadas: Editar turma.

Pré Condições: A turma precisa existir.

Pós Condições:

Operação: Definir novo valor.

Parâmetros: Campo, valor.

Referências cruzadas: Editar turma.

Pré Condições: A turma precisa existir.

Pós Condições:

Operação: Iniciar cancelamento.

Parâmetros:

Referências cruzadas: Cancelar turma.

Pré Condições: A turma precisa existir.

Pós Condições:

Operação: Cancelar.

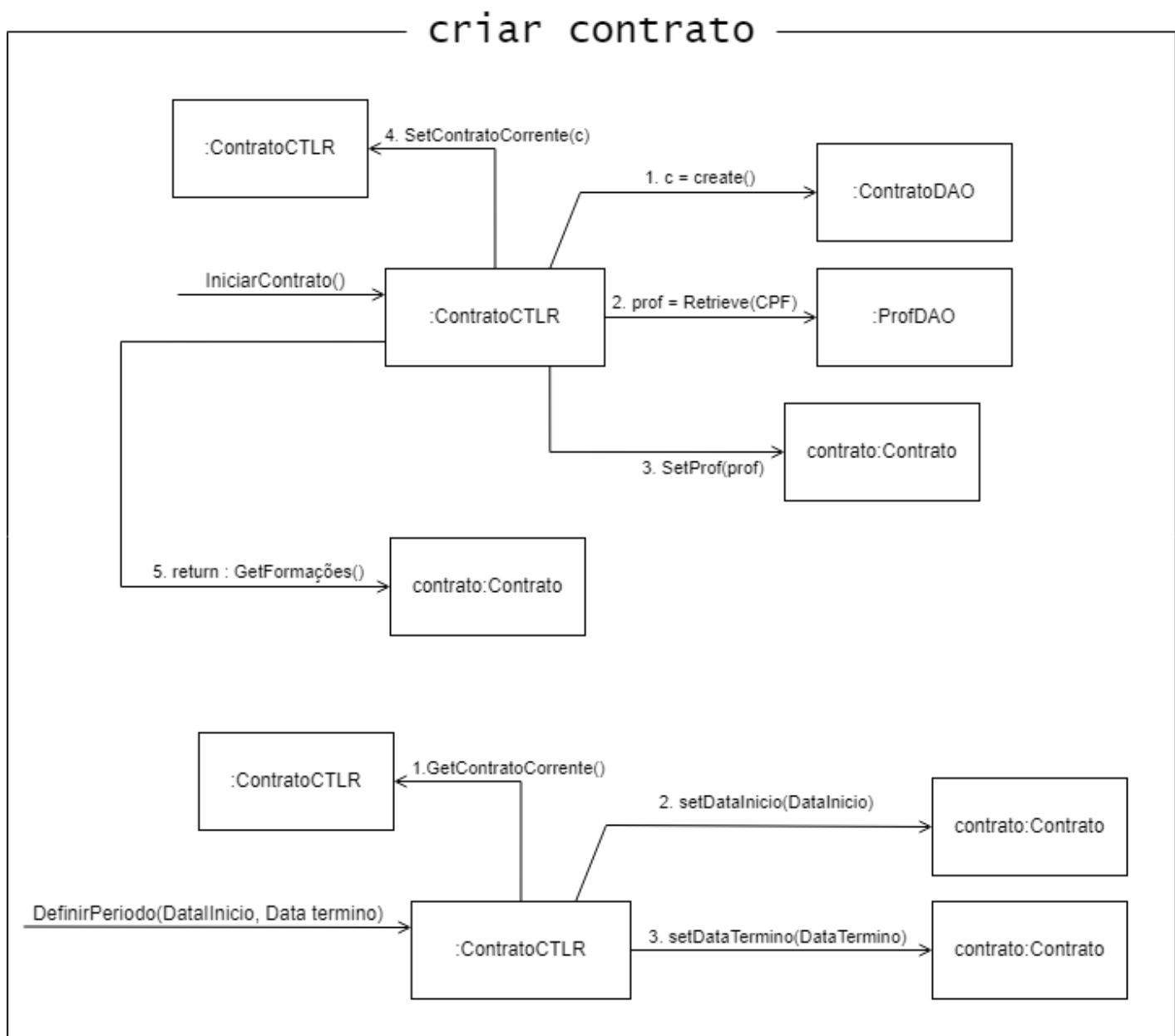
Parâmetros: Id da turma.

Referências cruzadas: Cancelar turma.

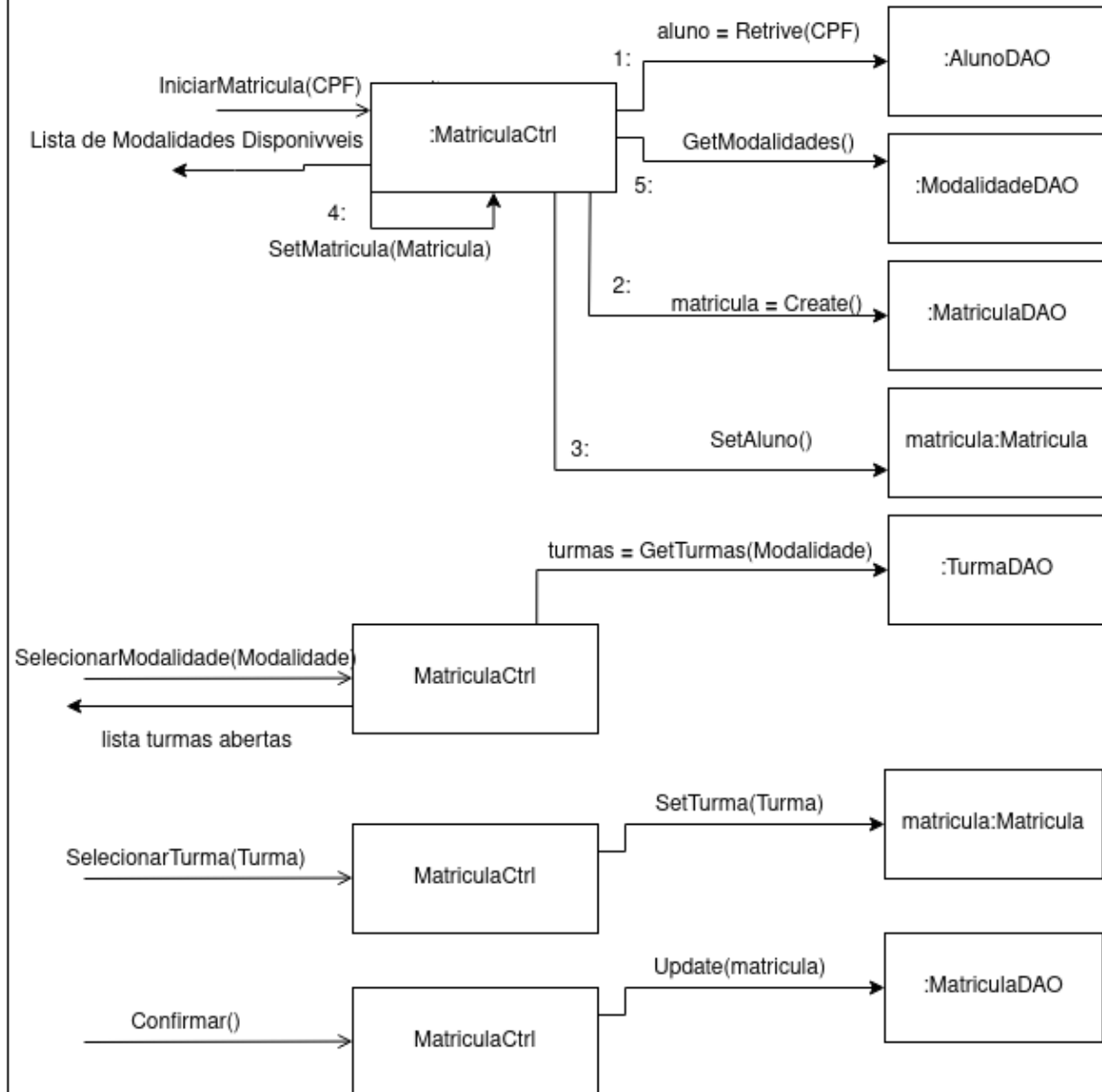
Pré Condições: A turma precisa existir.

Pós Condições:

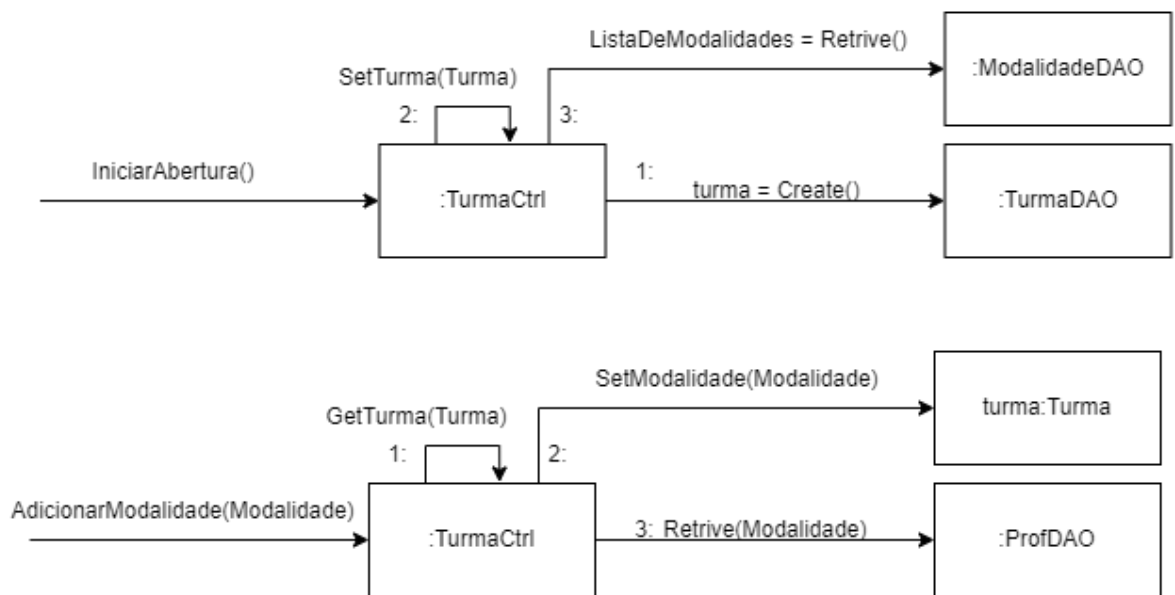
Diagrama de Comunicação



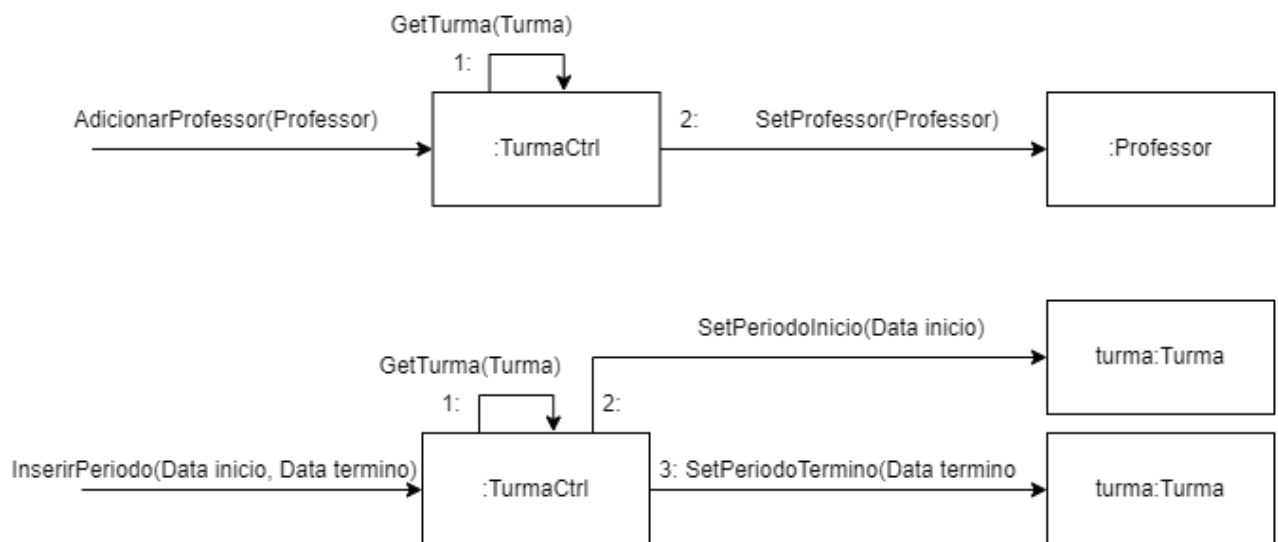
Matricular



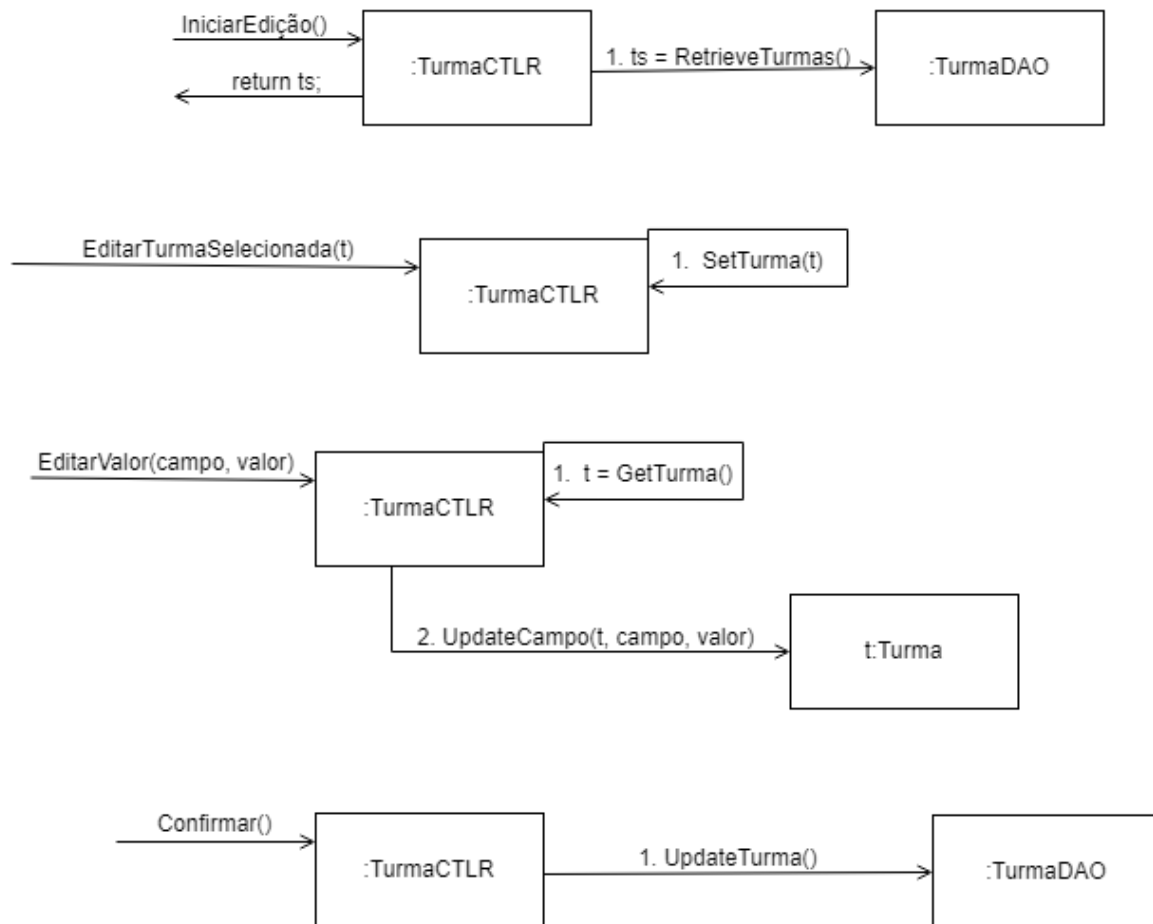
criar Turma



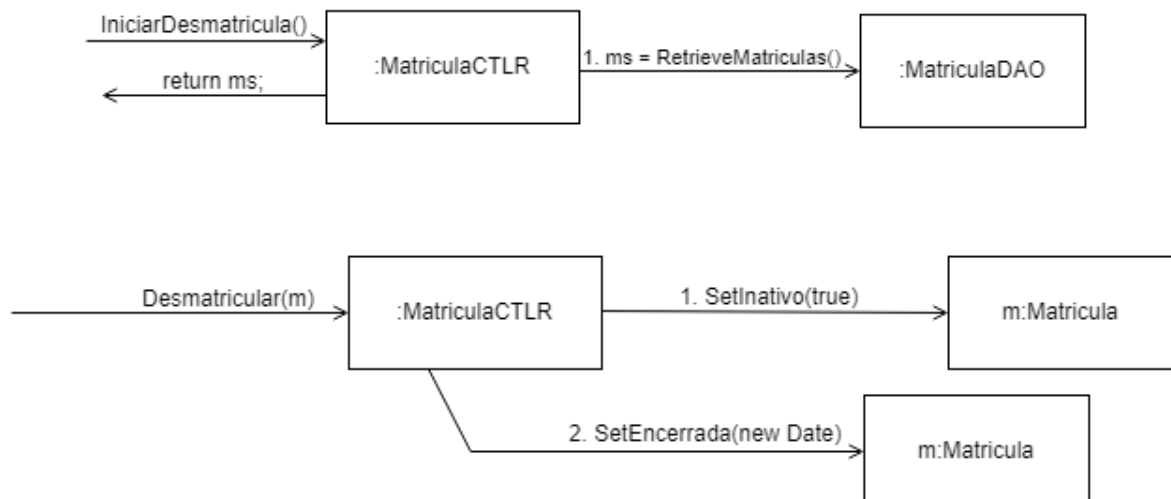
criar Turma



editar turma



desmatricular



cancelar turma

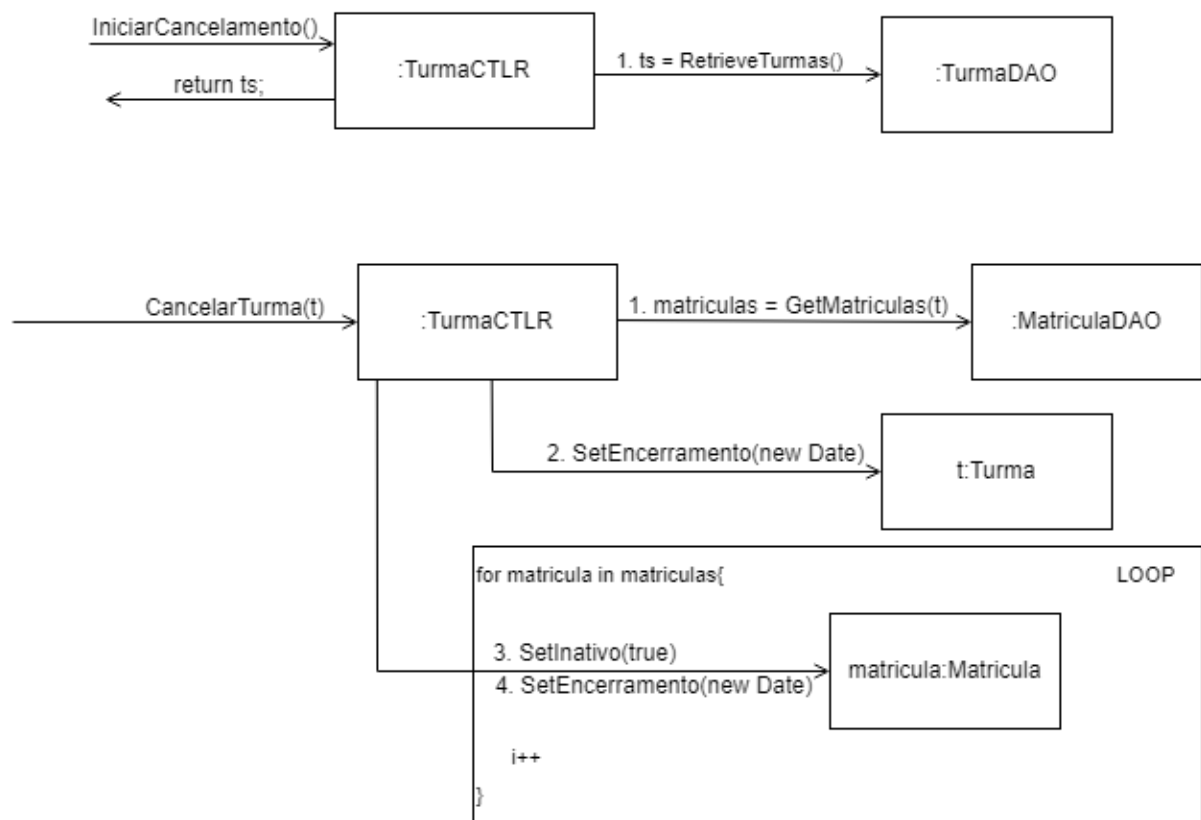
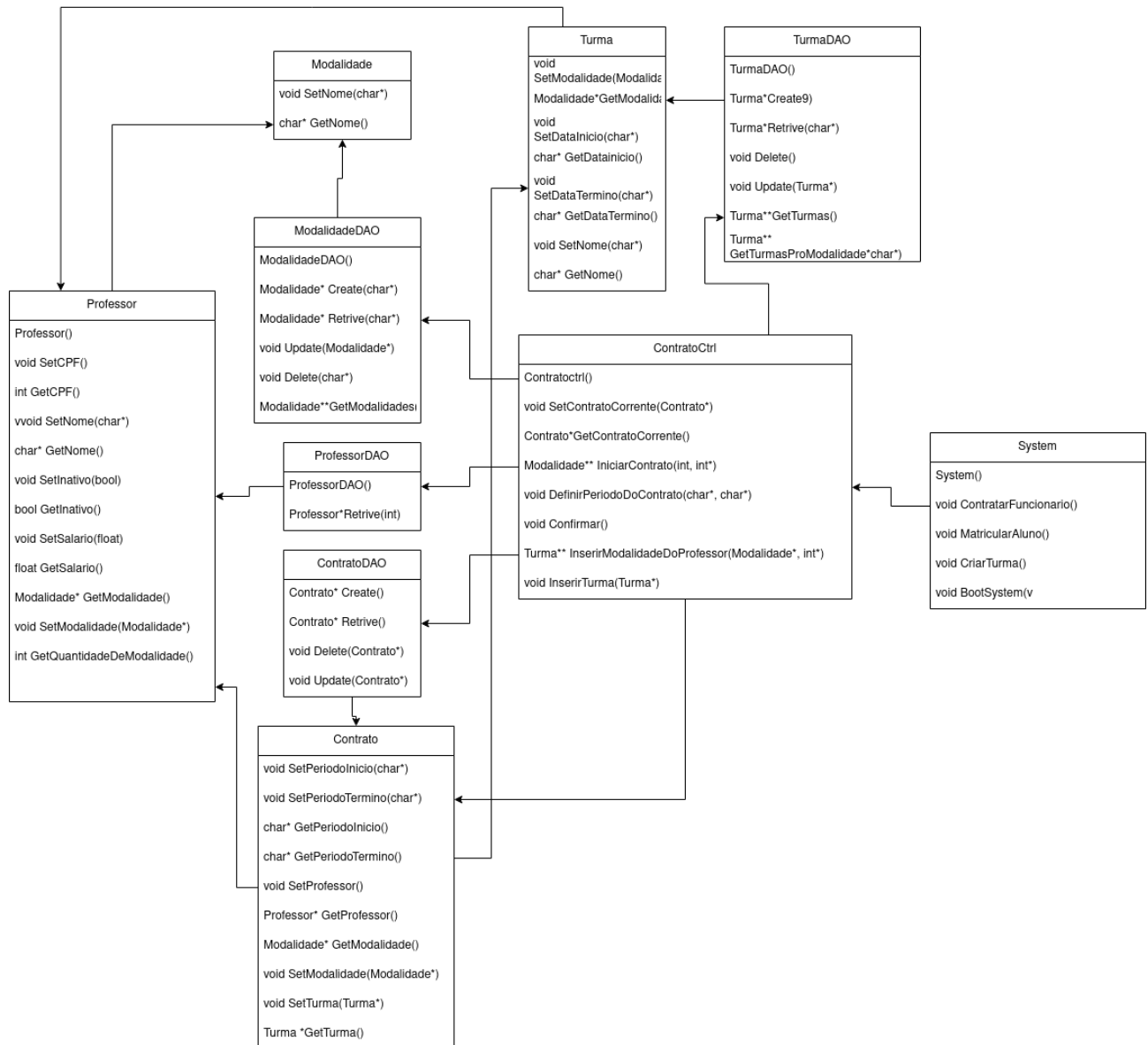


Diagrama de classe



Referências

Nesta seção, são apresentadas as referências utilizadas para a elaboração deste documento.

Site: Draw.io