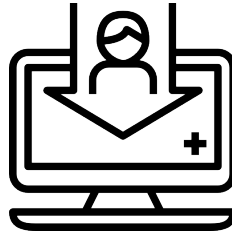




Pontifícia Universidade Católica de São Paulo
Faculdade de Ciência Exatas e Tecnologia
Ciência da Computação

Projeto: Cadastro de alunos em Java



LED: Laboratório de Estruturas Dinâmicas

Prof. Julio Arakaki

Abril 2024

1 Descrição

Este projeto tem como objetivo inicial a implementação de um cadastro de alunos utilizando a linguagem java. Como armazenador para este cadastro, deverá ser utilizado um "array" de objetos do tipo Aluno. Posteriormente será utilizada outra estrutura de dados que exemplifica a alocação dinâmica destes objetos. Por exemplo, uma lista ligada cujos elementos serão objetos da classe Aluno.

O armazenamento dessas informações dos alunos deverá ser feita utilizando-se destas estruturas de dados no qual o conteúdo de cada elemento destas estruturas dinâmicas será o endereço do objeto da classe Aluno.

Serão utilizadas classes já implementadas e outras deverão ser especificadas e implementadas. A figura 1 mostra um diagrama de classes, incluindo as dependências entre essas classes. Este diagrama é uma representação (modelagem) dos módulos de software e as respectivas dependências que devem auxiliar na implementação desta aplicação.

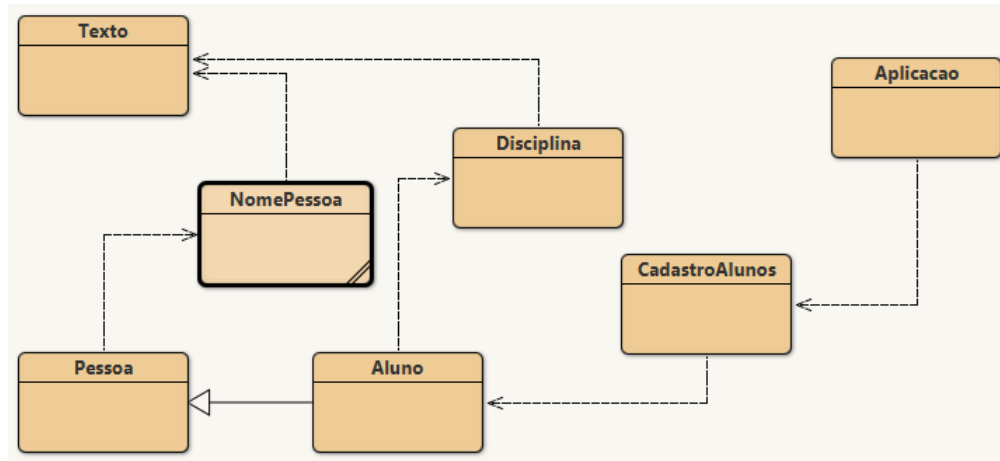


Figure 1: *Diagrama de classes da aplicação.*

2 Requisitos iniciais

A aplicação a ser desenvolvida, deverá atender aos requisitos básicos iniciais descritos neste item. Posteriormente, outros requisitos serão solicitados no decorrer do semestre e de acordo com o andamento das aulas.

2.1 Funcionais

São requisitos relacionados com a funcionalidade da aplicação.

- REQ01: Menu com as operações sobre o cadastro de alunos

Inicialmente deve ser criado um menu para a aplicação de cadastro com as seguintes opções:

```
(I)nserir
(R)emover
(L)istar
(S)air
```

Neste caso, deverá ser utilizado os métodos e as classes da linguagem java, responsáveis pela entrada e Saída (Scanner, JOptionPane, System.out.println, etc.) .

REQ01.01: Opção **inserir**, permite que o usuário da aplicação forneça os dados do aluno e em seguida, este aluno é inserido na estrutura de dados utilizada no cadastro. A aplicação deverá verificar

se já existe algum aluno com o RA fornecido. Caso já exista, não deixa inserir e emitir mensagens coerentes.

Os dados para cada aluno devem ser (por exemplo): **nome, idade, ra, curso, disciplinas**. Para cada disciplina no qual o aluno está matriculado, deverá ter: **nome da disciplina e respectiva nota**

REQ01_02: Opção **remover**, ao usuário deverá ser perguntado qual o RA do aluno ser removido. Se for encontrado, o aluno correspondente deverá ser removido do cadastro. Caso não exista o aluno no cadastro, deverá ser emitido a mensagem *"Aluno não cadastrado"*

REQ01_03: Opção **listar**, mostra todos as informações dos alunos cadastrados.

REQ01_04: Opção **Sair**, finaliza a aplicação. Futuramente, com a implementação da persistência dos dados em arquivo, deverá ser feito a verificação se os dados foram salvos ou não.

2.2 Implementação

Estes requisitos são relacionados com alguns detalhes de implementação.

- REQ02: Criar classes seguindo o padrão com construtores e, sempre que necessário, os métodos "setters e getters".
- REQ03: Utilizar classes já implementadas como: Texto, NomePessoa, entre outras.

Essa reutilização de classes já implementadas, visa possibilitar o uso das funcionalidades já contidas nestas classes como inverter nome do aluno assim como a possibilidade de escrever o nome do aluno no formato de referência bibliográfica.

- REQ04: Criar outras classes, de acordo com a necessidade

Além de utilizar as classes já implementadas nas aulas anteriores, devem ser implementadas também as seguintes classes: Pessoa, Aluno, Disciplina e Cadastro. Todas essas com suas respectivas características (Atributos) e dependências (Observadas no diagrama de classe deste projeto, conforme a figura 1).

2.3 Não-Funcionais

- REQ05: Proteger todas as entradas do usuário (entradas inválidas).

Todas as entradas fornecidas pelo usuário devem ser protegidas.

Por exemplo: quando for realizada entrada de alguma informação numérica e o usuário fornecer uma letra, informar o usuário que a entrada foi inválida. Ou seja, deve-se evitar que apareça mensagens técnicas para o usuário (erros, exceções,etc.)