

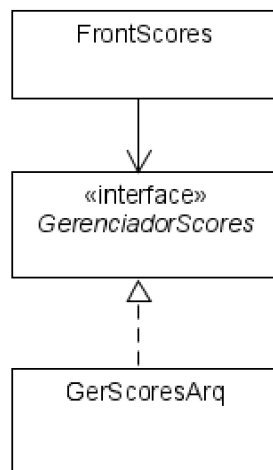
Checkpoint 5

Objetivos:

- Trabalhar com os conceitos de manipulação de dados em Java, Arquitetura em 3 camadas, Interfaces, Persistência de dados (Arquivos), Tratamento de Exceções.

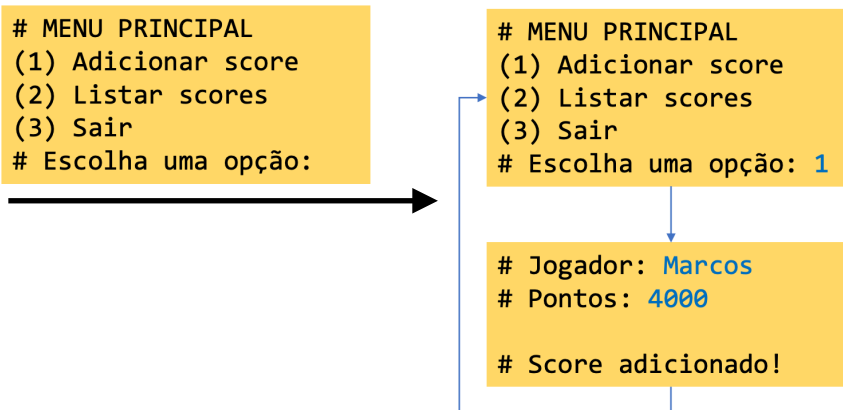
(10,0) Gerenciador de Scores

Desenvolvimento de uma aplicação em Java capaz de realizar o gerenciamento de *scores*. A equipe foi contratada para implementar uma parte de um projeto de acordo com o diagrama abaixo:



Funcionamento do sistema

- FrontScores** - Responsável pela interface com o usuário.
- Exemplo:



Checkpoint 5

- <<**interface**>> GerenciadorScores - especificação da interface.
- **GerScoresArq** - Responsável por armazenar e recuperar os scores de um arquivo

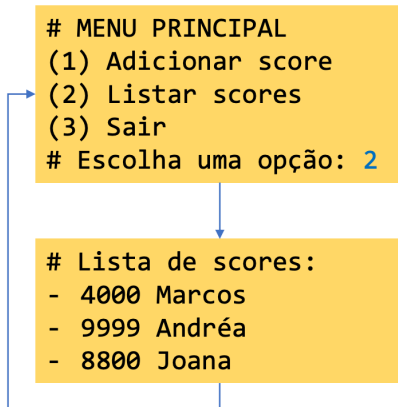
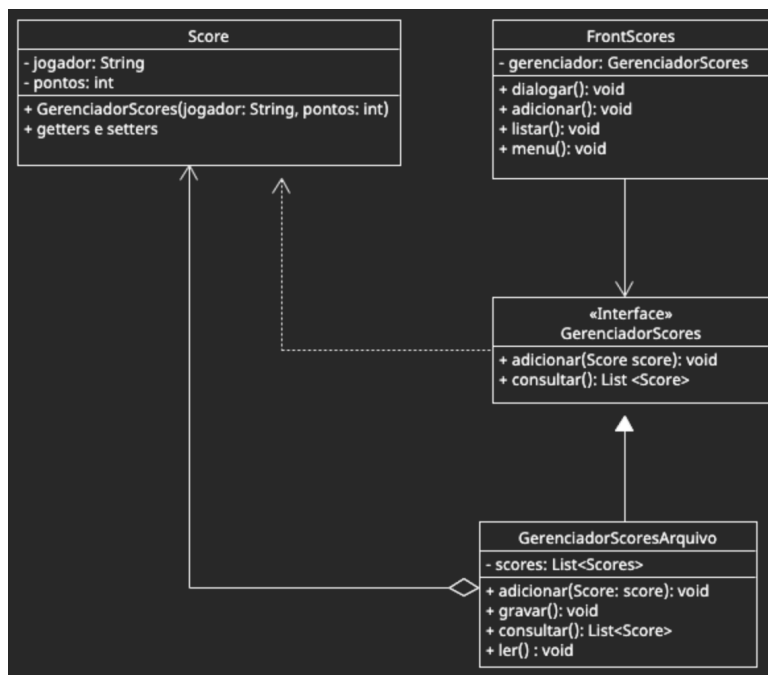


Diagrama e Descrição das Classes



- **Classe Score** - Trata-se de uma classe VO (*Value Object*) representando um objeto Score(nome e pontuação).
- **Interface GerenciadorScores** - contrato contendo as assinaturas dos métodos adicionar (adiciona um item (score) na lista) e consultar (consulta um item (score) na lista).
- **Classe GerenciadorScoresArquivo** - implementa a interface GerenciadorScores, além dos métodos gravar (grava um score no arquivo) e ler (leitura de todos os registros gravados no arquivo).



Domain Driven Design
Prof. Fernando Almeida

Checkpoint 5

- **Classe FrontScores** - Faz a interface com o usuário e possui os métodos dialogar, menu (exibe as opções disponíveis na aplicação - adicionar, listar ou sair), adicionar (adiciona um *score* na lista) e listar (lista os itens (*scores*) armazenados na lista).

Entrega

Pasta contendo o Projeto Java (código fonte) compactada com os pacotes e arquivos do projeto.

Critérios de avaliação

- Organização do Projeto
- Separação das responsabilidades (Arquitetura em camadas)
- Fidelidade a especificação do projeto
- Funcionamento do Programa