

UNIVERSIDADE DO MINHO
MESTRADO INTEGRADO EM ENGENHARIA INFORMÁTICA



Relatório - Programação Orientada aos
Objetos

GRUPO 43

Trabalho realizado por:



A75119
Adriana Martins
Gonçalves



A88539
Luís Miguel Teixeira
Fernandes



A58898
Rui Pedro Gomes
Coelho

Braga, 12 de junho de 2021

Conteúdo

1	Introdução	2
2	Descrição da aplicação	3
3	Manual de utilização da aplicação	3
3.1	<i>Menu Jogadores</i>	3
3.2	<i>Menu Equipas</i>	5
3.3	<i>Menu Jogos</i>	6
4	Arquitetura da aplicação	10
4.1	<i>Jogador</i>	10
4.2	<i>Avançado</i>	11
4.3	<i>Defesa</i>	12
4.4	<i>Guarda-Redes</i>	12
4.5	<i>Lateral</i>	12
4.6	<i>Médio</i>	12
4.7	<i>Equipa</i>	12
4.8	<i>Jogo</i>	13
4.9	<i>Menu</i>	14
5	Diagrama de Classes	15
6	Conclusão	16

1 Introdução

Este projeto foi-nos solicitado pelos docentes da unidade curricular de Programação Orientada aos Objetos e tem como principal objetivo a realização de um projeto em Java sobre a simulação e gestão de equipas de futebol.

Através da definição das classes intervenientes, o programa deve permitir que em qualquer momento se possa guardar em ficheiro a informação sobre as várias entidades existentes em memória. A gravação deve ser feita de forma a permitir que o estado que foi gravado seja recuperado novamente. Neste relatório iremos ainda apresentar e fundamentar a arquitetura e desempenho do trabalho realizado, bem como eventuais limitações/superações dos objetivos propostos.

2 Descrição da aplicação

O projeto tem como principal objetivo criar a gestão e simulação de equipas de futebol, bem como todo o envolvente que um jogo deste desporto tem (desde substituições, a resultados finais, equipa principal etc).

Para isso, o nosso programa será dividido em Jogadores, Equipas e Jogos. Os Jogadores podem ser avançados, defesas, guarda-redes, laterais ou médios. Cada jogador possuirá características únicas de acordo com a posição que ocupa em campo. Tanto os jogadores como as equipas, deverão possuir um histórico com todas as informações sobre a que clube já fizeram parte (no caso dos jogadores) e a lista dos jogadores que já integraram uma equipa (no caso da equipa).

Depois de tudo isto definido, será ainda possível simular um jogo entre duas equipas, efetuar substituições e saber o resultado de um jogo.

3 Manual de utilização da aplicação

Esta é uma aplicação com uma interface para o utilizador muito simples feita por nós, de maneira a que o utilizador pudesse tirar o maior proveito da mesma, com comandos simples, tendo em conta que todos os menus funcionam à base de opções por números.

Quando um utilizador executa a aplicação, o primeiro menu que lhe será apresentado é o seguinte:

```
----- Menu -----
(1) Jogadores
(2) Equipas
(3) Jogos
(4) Carregar estado de um ficheiro
(5) Guardar estado num ficheiro
(6) Carregar logs
(0) Sair

Selecione opção:
```

Figura 2: Menu inicial

No menu inicial apresentado acima, estão presentes todos os sub-menus presentes na aplicação. Optamos por dividir os menus por Jogadores, Equipas, Jogos e os restantes três menus são responsáveis por carregar os dados para a aplicação. Sempre que necessário, pode-mos voltar atrás inserindo um '0' na linha de comandos.

3.1 Menu Jogadores

O *menu jogador* permite adicionar um novo jogador (escrever 1) ou remover do conjunto de jogadores inscritos na aplicação (escrever 2).

```
----- Menu -----
(1) Adicionar jogador
(2) Remover jogador
(3) Consultar todos os jogadores
(4) Consultar um jogador
(5) Calcular habilidade de um jogador
(0) Sair

Selecionar opção: █
```

Figura 3: Menu jogadores

Ao fazer a inscrição, será apresentado como o menu representado abaixo, onde deverá indicar a posição do jogador a inscrever. Para isso, basta escrever o nº correspondente, presente no início de cada posição.

```
----- Menu -----
(1) Avançado
(2) Defesa
(3) Guarda-Redes
(4) Lateral
(5) Médio
(0) Sair

Selecionar opção: █
```

Figura 4: Posição do jogador

Depois disso, deverá preencher todos os dados pedidos. Neste caso, conforme podemos ver no exemplo abaixo, pede o nome do jogador, o nº da camisa, a velocidade, a resistência, a destreza, o impulso, o jogo de cabeça, a capacidade de remate, de passe e de marcação de penaltis. Este último dado pertence apenas aos avançados, cada uma das restantes posições possuem o mesmos dados a preencher com uma variação numa característica. Os **avançados** possuem a característica particular 'penalti', os **defesas** 'marcação', os **guarda-redes** 'elasticidade' e 'defesa', os **laterais** 'cruzamento' e os **médios** 'recuperação'.

```
Inserir nome do jogador: Cristiano Ronaldo
Inserir número da camisa: 7
Inserir velocidade (0-100): 99
Inserir resistência (0-100): 99
Inserir destreza (0-100): 99
Inserir impulso (0-100): 99
Inserir jogo de cabeça (0-100): 99
Inserir capacidade de remate (0-100): 99
Inserir capacidade de passe (0-100): 99
Inserir capacidade de marcação de penalti (0-100): 99
```

Figura 5: Menu de registo

Para além disso, é ainda possível consultar todos os jogadores (escrever 3) onde aparece

no ecrã todos os jogadores presentes na aplicação, consultar apenas um jogador (escrever 4) e aparecerá todas as características do jogador cujo nome for escrito na consola e ainda o histórico desse mesmo jogador, ou ainda calcular a habilidade (escrever 5).

```
Inserir nome do Jogador: Daniela Eduarda Cruz Fernandes
#5 Daniela Eduarda Cruz Fernandes | Velocidade: 53 | Resistência: 71 | Destreza: 5 | Impulsão: 69 |
Cabeça: 51 | Remate: 80 | Passe: 76 | Recuperação: 61 | Posição: Medio | Histórico: [Mozart F. C.]
```

Figura 6: Menu consultar um jogador

```
Inserir nome do Jogador: Daniela Eduarda Cruz Fernandes
Habilidade: 315.4
```

Figura 7: Habilidade de um jogador

3.2 *Menu Equipas*

Quando ao *menu equipa*, permite adicionar uma nova equipa (escrever 1) ou remove-la da lista de equipas inscritas na aplicação (escrever 2).

```
----- Menu -----
(1) Adicionar equipa
(2) Adicionar jogadores a uma equipa
(3) Consultar todas as equipas
(4) Consultar uma equipa
(5) Calcular habilidade de uma equipa
(0) Sair

Selecionar opção: █
```

Figura 8: Menu Equipa

Ao fazer a inscrição da equipa, terá apenas de escrever o nome da equipa, conforme podemos verificar na imagem abaixo.

```
Inserir nome da Equipa: UMinho F. C.
```

Figura 9: Menu registar equipa

Caso pretenda adicionar um jogador a uma equipa, apenas terá de escrever o nome do jogador e a equipa para onde o quer transferir.

```
Inserir nome da Equipa: Vivaldi F. C.  
Inserir nome do Jogador: Cristiano Ronaldo
```

Figura 10: Adicionar jogador a equipa

É ainda possível consultar todas as equipas e respetivos jogadores (escrever 3) ou consultar apenas uma equipa (escrever 4). Para isso basta escrever o nome da equipa a consultar, e aparecerá todos os jogadores dessa equipa, bem como a respetiva posição em campo e habilidade.

```
Inserir nome da Equipa: Vivaldi F. C.  
Equipa: Vivaldi F. C.  
#43 | Jorge Rafael Gomes Azevedo | Medio | Habilidade: 418,4  
#3 | Beatriz da Ponte Rocha | Lateral | Habilidade: 275,1  
#20 | Silvia Maria Acosta Roca | GuardaRedes | Habilidade: 322,6  
#30 | Ricardo Manuel Almeida Vieira De Castro | Avancado | Habilidade: 344,5  
#26 | Ines Maria Fernandes Martins | Medio | Habilidade: 468,9  
#28 | Joao Miguel Amorim Alves | Avancado | Habilidade: 361,6  
#10 | Francisco Antonio Borges Paulino | Medio | Habilidade: 423,6  
#33 | Tiago Manuel Da Silva Rego Macedo Alves | Avancado | Habilidade: 439,3  
#47 | Sergio Miguel Cabral Passinhas Dos Santos Costa | Lateral | Habilidade: 344,6  
#38 | Pedro Jorge Fernandes Martins | Lateral | Habilidade: 256,2  
#41 | Carlos Filipe Almeida Dias | Avancado | Habilidade: 357,7  
#16 | Helder Cristiano Miranda Lopes | Lateral | Habilidade: 258  
#4 | Daniel Constantino Martins Faria | Defesa | Habilidade: 388,3  
#42 | Tomas Ferreira Sousa Dias | Lateral | Habilidade: 363,7  
#7 | Pedro Tavares Alves | Avancado | Habilidade: 358,4  
#18 | Hugo Joao Pereira Calheiros Brandao | GuardaRedes | Habilidade: 149,5  
#31 | Ricardo Lopes Santos Silva | Defesa | Habilidade: 290,8  
#39 | Leonor Lopes Caldas | Defesa | Habilidade: 333  
#8 | Mariana Rocha Marques | Medio | Habilidade: 229,1  
#50 | Miguel Angelo Mesquita Rego | Medio | Habilidade: 384,1
```

Figura 11: Menu consultar equipa

Caso pretenda apenas consultar a habilidade de uma equipa, tem apenas de inserir a equipa que pretende consultar e surgirá o cálculo da habilidade, conforme exemplificado na imagem abaixo.

```
Inserir nome da Equipa: Vivaldi F. C.  
Habilidade global da equipa: 9296,4
```

Figura 12: Habilidade de uma equipa

3.3 Menu Jogos

O *menu jogos*, é o menu relativo às informações de jogos e ainda onde é possível simular a realização de um jogo entre duas equipas.

```
----- Menu -----
(1) Adicionar jogo
(2) Consultar todos os jogos
(3) Consultar um jogo
(4) Determinar o resultado de um jogo
(5) Simular jogo
(0) Sair

Selecionar opção: █
```

Figura 13: Menu jogos

Para adicionar um jogo à aplicação, terá de escrever 1 na linha de comandos e, posteriormente, proceder ao preenchimento das informações pedidas. Neste caso, deve inserir-se uma data e o nome da equipa da casa e da equipa visitante.

```
Inserir data jogo segundo o formato YYYY-MM-DD: 2021-06-03
Inserir nome da equipa da casa: Mendelssohn F. C.
Inserir nome da equipa visitante: Mozart F. C.
```

Figura 14: Adiciona jogos

Pode-se ainda consultar todos os jogos realizados (escrever 2) ou consultar apenas um determinado jogo (escrever 3). Para isso basta escrever a data em que o jogo se realizou, o nome da equipa da casa e o nome da equipa visitante. Conforme podemos ver na imagem abaixo, as informações como o resultado, os titulares de cada equipa e as substituições serão apresentadas relativas ao jogo pedido.


```

Inserir data jogo segundo o formato YYYY-MM-DD: 2021-01-16
Inserir nome da equipa da casa: Stravinsky Athletic
Inserir nome da equipa visitante: Sporting Club Prokofiev

***** JOGO *****
Data: 2021-01-16
Equipas: Stravinsky Athletic vs Sporting Club Prokofiev
Resultado: 2-2

Stravinsky Athletic
Titulares:
[0, 1, 8, 13, 17, 19, 22, 29, 36, 42, 50]

Substituições:
1 --> 24
23 --> 10
36 --> 23

Sporting Club Prokofiev
Titulares:
[3, 5, 9, 11, 15, 17, 28, 38, 43, 47, 48]

Banco:
3 --> 30
38 --> 1
47 --> 35

```

Figura 15: Consulta um jogo

Para determinar o resultado de um jogo (escrever 4), os dados a inserir são semelhantes aos do caso de adicionar um jogo ou consultar um jogo.

Ao escrever 5 na linha de comandos, torna-se possível simular a realização de um jogo. Assim, será necessário adicionar um jogo primeiro, e depois disso, as regras de funcionamento de um jogo vão surgir no ecrã, conforme podemos ver na imagem abaixo.

```

Regras de simulação de jogo:
(1) Um jogo é constituído por duas partes;
(2) Em cada parte, são permitidas no máximo 10 ações;
(3) As substituições efetuadas terão de ser necessariamente válidas;
    (3.1) São permitidas, no máximo, 3 substituições no decurso de um jogo;
    (3.2) As substituições são válidas se for seleccionado um jogador presente em campo e outro no banco

```

Figura 16: Regras de um jogo

Para iniciar o jogo basta carregar numa tecla aleatória à escolha do utilizador. Ao iniciar a primeira parte, temos 4 opções disponíveis, visíveis na imagem abaixo.

```

--- PRIMEIRA PARTE: INÍCIO ---
Wagner Athletic vs Vivaldi F. C.

(1) Substituir jogador da equipa da casa
(2) Substituir jogador da equipa visitante
(3) Jogar
(0) Terminar parte
    Selecionar opção: █

```

Figura 17: Menu primeira parte jogo

As primeiras duas, dizem respeito às substituições dos jogadores, tanto da equipa da casa (escrever 1) ou da equipa visitante (escrever 2). A forma de funcionamento é semelhante em ambos os casos, tendo apenas que seleccionar os jogadores que saem e entram.

```

Jogadores em campo
#23 | Joao Pedro Martins da Silva Freitas Cardoso | Medio | Habilidade: 390,5
#28 | Raul Arieira Parente | Avancado | Habilidade: 404,5
#48 | Nelson Ramos Alves | Defesa | Habilidade: 334,8
#36 | Eduardo Costa de Magalhaes | GuardaRedes | Habilidade: 291,3
#2 | Bruno Filipe Jardim Machado | Avancado | Habilidade: 397,7
#15 | Diogo da Costa e Silva Lima Rebelo | Lateral | Habilidade: 362,8
#17 | Francisco Jose Pereira Teofilo | Medio | Habilidade: 508,1
#20 | Mariana Dinis Rodrigues | Lateral | Habilidade: 304,2
#32 | Jose Pedro Pinto Teixeira da Cunha | Avancado | Habilidade: 374,2
#30 | Pedro Nuno Fernandes Freitas | Lateral | Habilidade: 295,2
#25 | Tomas Cardoso Francisco | Lateral | Habilidade: 325,7

Jogadores disponíveis para trocar
#40 | Barbara Paulista de Faria | Lateral | Habilidade: 305,7
#21 | Goncalo Miguel Leao Barros Oliveira Pinto | Medio | Habilidade: 261,1
#8 | Fabio Fernandes Silva | Defesa | Habilidade: 374,3
#27 | Pedro Miguel Sampaio Faria | Defesa | Habilidade: 356,6
#11 | Carlos Daniel da Silva Santos | Medio | Habilidade: 459,7
#22 | Pedro Antonio Pires Correia Leite Sequeira | Avancado | Habilidade: 408,4
#12 | Joao Guilherme Rodrigues Reis | Medio | Habilidade: 391,3
#29 | Tiago Rafael Fernandes Guimaraes | Lateral | Habilidade: 320,9
#39 | Guilherme Gil Rocha Goncalves | Avancado | Habilidade: 307,6
Selecionar jogador para sair:

```

Figura 18: Substituições jogo

Caso não se pretenda fazer nenhuma substituição, pode apenas fazer o jogo arrancar ao seleccionar a opção 'Jogar' (escrever 3), e caso à da figura abaixo.

```

GOL0! Marcou a equipa visitante.

(1) Substituir jogador da equipa da casa
(2) Substituir jogador da equipa visitante
(3) Jogar
(0) Terminar parte
    Selecionar opção:

```

Figura 19: Golo equipa visitante

Para terminar uma parte, basta escrever 0 na linha de comandos. A segunda parte é iniciada e o desempenho da simulação é igual à da primeira parte.

```

--- PRIMEIRA PARTE: FIM ---
--- INTERVALO ---
--- SEGUNDA PARTE: INÍCIO ---
(1) Substituir jogador da equipa da casa
(2) Substituir jogador da equipa visitante
(3) Jogar
(0) Terminar parte
    Selecionar opção:

```

Figura 20: Fim da primeira parte/Início da segunda parte

No final do jogo, aparece uma mensagem semelhante à que temos abaixo.

```

--- SEGUNDA PARTE: FIM ---
--- FIM DO JOGO ---
Wagner Athletic vs Vivaldi F. C.

```

Figura 21: Fim do jogo

4 Arquitetura da aplicação

Neste capítulo falaremos sobre a estrutura da nossa aplicação bem como de todas as classes, atributos e funcionamento de cada um destes.

4.1 *Jogador*

A classe Jogador é importante na medida em que é uma generalização das cinco classes que a herdam e para além disso permite-nos evitar a repetição de variáveis e métodos. Esta classe servirá como modelo para as outras classes que dela dependam. Neste caso, foram criadas as classes **avançado**, **defesa**, **guarda-redes**, **lateral** e **médio** que herdam a informação que está na super classe (Jogador).

Esta classe terá como variáveis de instância o nome, o número da camisola, a velocidade, a resistência, a destreza, a impulsão, o jogo de cabeça, remate, passe e habilidade de cada um

dos jogadores. Nos métodos desta classe estão incluídos os métodos mais comuns: get's, set's e equals.

```
private String nome;  
private int nrCamisola;  
private int velocidade;  
private int resistencia;  
private int destreza;  
private int impulsao;  
private int cabeca;  
private int remate;  
private int passe;  
private double habilidade;  
private Set<String> historicoEquipas;
```

- **String nome:** nome do jogador.
- **int nrCamisola:** número da camisola do jogador.
- **int velocidade:** velocidade do jogador.
- **int resistencia:** resistência do jogador.
- **int destreza:** destreza do jogador.
- **int impulsao:** impulsão do jogador.
- **int cabeca:** jogo de cabeça do jogador.
- **int remate:** remate do jogador.
- **int passe:** habilidade geral do jogador.
- **double habilidade:** capacidade de passe do jogador.
- **Set<String> historicoEquipas:** lista com nome das equipas às quais o jogador já pertenceu.

4.2 *Avançado*

Esta classe corresponde a um avançado e é uma "especificação" de um Jogador onde estarão todos os dados herdados da super classe e mais alguns. A esta classe foi ainda acrescentada uma variável de instância correspondente à capacidade de marcação de penaltis por um avançado.

```
private int penalti;
```

- **int penalti:** capacidade de marcação de penaltis.

4.3 *Defesa*

Esta classe corresponde a um defesa e é também uma "especificação" de um Jogador. A esta classe foi acrescentada a variável de instância correspondente à capacidade de marcação de um defesa aos jogadores adversários.

```
private int marcacao;
```

- **int marcacao:** capacidade de marcar devidamente os jogadores adversários.

4.4 *Guarda-Redes*

Esta classe corresponde a um guarda-redes corresponde também a uma "especificação" de um Jogador. A esta classe foram ainda acrescentadas duas variáveis de instância correspondentes à capacidade de elasticidade e defesa de um guarda-redes.

```
private int elasticidade;  
private int defesa;
```

- **int elasticidade:** elasticidade do guarda-redes.
- **int defesa:** capacidade de defesa da baliza.

4.5 *Lateral*

Esta classe corresponde a um lateral e é também uma "especificação" de um Jogador. A esta classe foi ainda acrescentada uma variável de instância correspondente à capacidade de efetuar cruzamentos por parte de um lateral.

```
private int cruzamento;
```

- **int cruzamento:** capacidade de efetuar cruzamentos.

4.6 *Médio*

Finalmente, a classe médio e é a última "especificação" de um Jogador. A esta classe foi acrescentada a variável de instância correspondente à capacidade de recuperação da bola por um médio.

```
private int recuperacao;
```

- **int recuperacao:** capacidade de recuperar a bola.

4.7 *Equipa*

A classe Equipa é a responsável por definir tudo o que é relevante sobre as equipas. Esta classe terá como variáveis de instância o nome, uma lista com os jogadores que integram uma dada equipa, e o resultado da habilidade global da equipa. Nos métodos desta classe estão incluídos os métodos mais comuns: get's, set's e equals.

```
private String nome;
private Set<Jogador> jogadores;
private double habilidadeGlobal;
```

- **String nome:** nome da equipa.
- **Set<Jogador> jogadores:** jogadores que integram a equipa.
- **double habilidadeGlobal:** habilidade total da equipa.

4.8 *Jogo*

A classe Jogo é a responsável por definir tudo o que é relativo aos jogos. Esta classe terá como variáveis de instância o nome da equipa da casa e o da equipa visitante, os golos marcados pela equipa da casa e pela visitante, a data em que acontece o jogo e a lista dos jogadores e as substituições das duas equipas em jogo. Nos métodos desta classe estão incluídos os métodos mais comuns: get's, set's e equals.

```
private String equipaCasa;
private String equipaVisitante;
private int golosCasa;
private int golosVisitante;
private LocalDate dataJogo;
private Set<Integer> jogadoresCasa;
private Map<Integer,Integer> substituiçoesCasa;
private Set<Integer> jogadoresVisitante;
private Map<Integer,Integer> substituiçoesVisitante;
private int fimJogo;
```

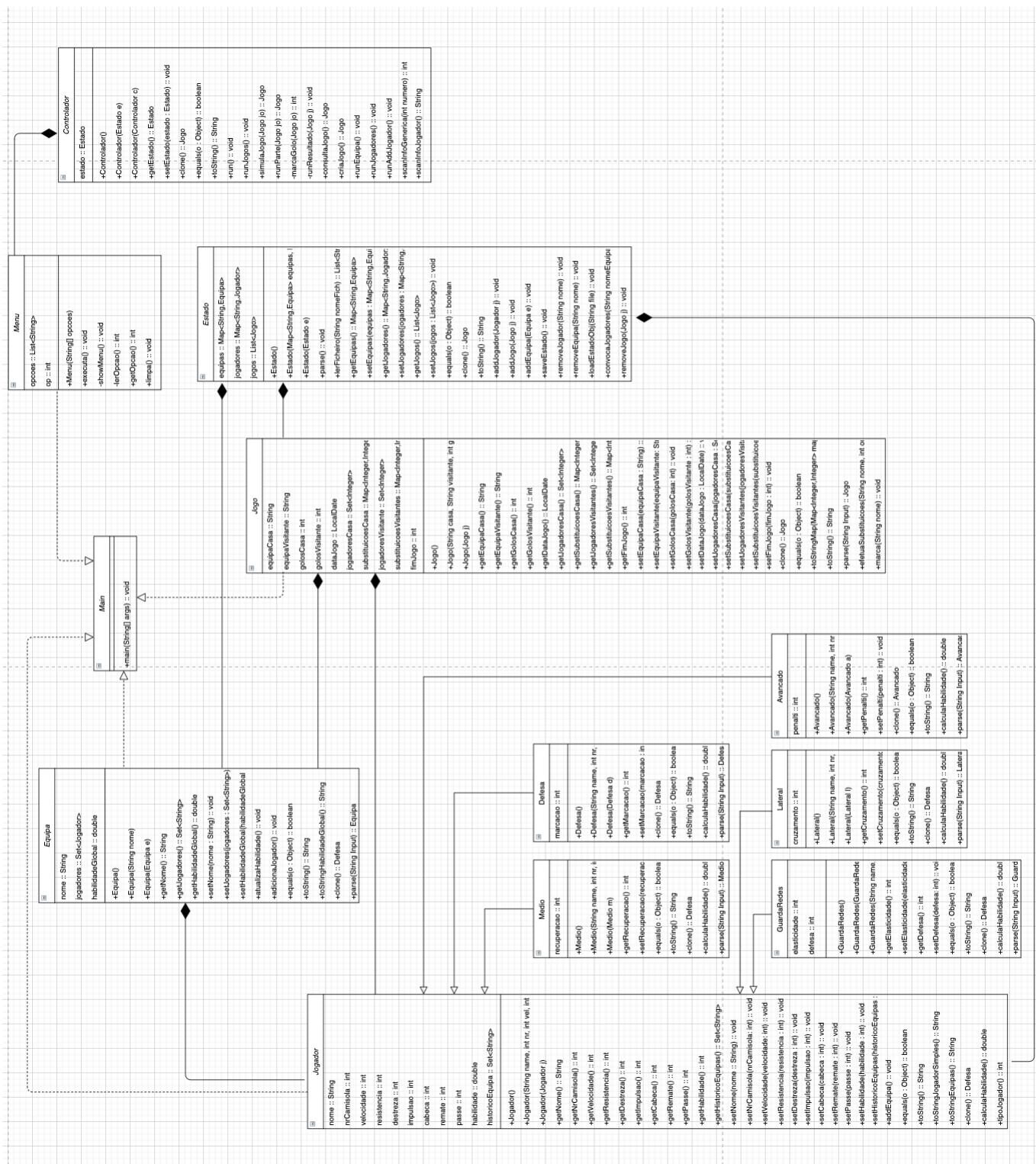
- **String equipaCasa:** nome da equipa da casa.
- **String equipaVisitante:** nome da equipa visitante.
- **int golosCasa:** golos marcados pela equipa da casa.
- **int golosVisitante:** golos marcados pela equipa de fora.
- **LocalDate dataJogo:** data do jogo.
- **Set<Integer> jogadoresCasa:** jogadores da equipa de casa.
- **Map<Integer,Integer> substituiçoesCasa:** substituições efetuadas pela equipa de casa.
- **Set<Integer> jogadoresVisitante:** lista de jogadores da equipa de fora.
- **Map<Integer,Integer> substituiçoesVisitante:** substituições efetuadas pela equipa de fora.
- **int fimJogo:** Um jogo terminado tem esta variável com o valor 1.

4.9 *Menu*

Esta classe tem como objetivo facilitar a criação de menus e a leitura da opção introduzida pelo utilizador visto que o nosso menu funciona à base de números. Contém, por exemplo, um método que nos permite formar novos menus dada uma string e um array de strings ou apenas uma string. Portanto, funções que tenham que ver com formação e interação em menus serão aqui definidas.

```
private List<String> opcoes;  
private int op;
```

5 Diagrama de Classes



6 Conclusão

Com a conclusão deste projeto foi possível perceber os diferentes aspetos no funcionamento da programação orientada aos objetos apesar de não ser uma linguagem nova para todos os elementos do grupo.

Assim, ao longo da realização deste projeto, fomos confrontados com algumas dificuldades nomeadamente no caminho a seguir para responder de forma mais eficiente ao pretendido e o tipo de estruturas a usar, tendo sido necessário um trabalho de investigação mais aprofundado de forma a dar resposta às nossas dúvidas.

Este projeto permitiu-nos ainda que o nível de conhecimentos sobre esta linguagem aumentasse bem como a experiência de trabalhar num projeto onde conseguimos aplicar grande parte dos conhecimentos que obtivemos nas aulas de POO.