Dragon Force

Sports academies solutions

 |Mestrado Integrado Engenharia Informática e Computação | 19 de novembro de 2016

João Furriel Pinheiro - up201104913@fe.up.pt

Leonardo Teixeira - up201502848@fe.up.pt

Luís Correia - up201503342@fe.up.pt

Índice

[1. Descrição Sucinta 2](#_Toc467445632)

[2. Solução/Casos de Uso 2](#_Toc467445633)

[3. Diagramas UML 3](#_Toc467445634)

[4. Principais Dificuldades 3](#_Toc467445635)

[5. Esforço Individual 4](#_Toc467445636)

# Descrição Sucinta

O Sistema desenvolvido permite a gestão de de atletas um clube desportivo de futebol. A flexibilidade do mesmo faz com que seja fácil gerir vários escalões para várias épocas.

Em linhas gerais a ferramenta permite:

* Gestão da Informação Completa de Atletas
* Gestão da Informação Completa de Treinadores
* Gestão da Informação da ficha de ECG dos Atletas

Para cada época, em cada escalão é possível:

* Registo de Treinos
* Registo de Jogos Amigáveis
* Registo de Torneios (em cada torneio o registo dos diferentes jogos)

O sistema guarda a informação do programa em ficheiros de texto para posterior carregamento em execuções futuras.

# Solução/Casos de Uso

A solução implementada é baseada num sistema de navegação por menus. Os menus, por usa vez, chamam os objetos gerados pelas classes bases do programa, sendo elas:

* Worker: classe usada para registar todos os trabalhadores do clube
  + Athlete: subclasse de Worker usada para gerir atletas
    - Goalkeeper
    - Defender
    - Midfielder
    - Forward
  + Coach: subclasse de Worker usada para gerir treinadores
* Club: classe usada para gerir o clube
* Season: classe usada para gerir informação das épocas desportivas
* Level: classe usada para gerir a informação de cada escalão
* Match: classe usada para gerir informação relativa a jogos
* Training: classe usada para gerir a informação relativa a treinos
* Tournament: classe usada para gerir informação relativa a torneios (esta classe faz uso de binary trees para gerir a árvore do torneio)

Outras classes adicionais:

* Date
* Table
* Diferentes tipos de exceções
* Fraction
* InfoAthlete (ficha de atleta com informação relativa a desempenho do mesmo)

Em termos de organização de ficheiros/pastas:

* Nome Do Clube(Pasta)
  + Ficheiro de Atletas
  + Ficheiro de Treinadores
  + Ficheiro de informação específica do clube
  + Pasta relativa a cada época desportiva
    - Pasta relativa a cada escalão
      * Ficheiro de atletas
      * Ficheiro de treinadores
      * Ficheiro de jogos amigáveis
      * Ficheiro de torneios
      * Ficheiro de treinos
      * Pasta relativa a jogos
        + Um ficheiro por cada jogo com informação específica do mesmo
      * Pasta relativa a torneios
        + Um ficheiro por cada torneio com informação específica do mesmo (essencialmente a árvore do torneio)

# Diagramas UML



# Principais Dificuldades

Parte 1:

A única dificuldade e que se revelou uma razão fundamental para não termos conseguido desenvolver tudo o que queríamos prendeu-se com o tempo. Temos planeado terminar todas as funcionalidades que nos propusemos desenvolver na segunda parte do trabalho, sobretudo a parte de interface com utilizador e alguns pequenos erros na parte gráfica do sistema e no lançamento de exceções mais específicas para cada erro. No entanto, o essencial pedido no enunciado foi desenvolvido.

Parte 2:

Em comparação à primeira parte do trabalho, o tempo disponível para fazer o que foi pedido foi mais razoável, tendo em conta a carga de trabalho. No entanto, o fato de termos vários projetos de outras cadeiras, ambos para entregar na mesma data, limitou um pouco os nossos esforços para este projeto.

# Esforço Individual

Parte 1:

* João Furriel Pinheiro
  + Desenvolvimento de classes base do sistema e os seus métodos principais
    - Club
    - Level
    - Match
    - Training
    - Tournament
    - Algumas classes e funções utilitárias (ex.:exceções)
* Leonardo Teixeira
  + Desenvolvimento da interface gráfica, alguns métodos auxiliares de classes base e desenvolvimento das seguintes classes
    - Worker
    - Coach
    - Athlete
    - Goalkeeper
    - Defender
    - Info
    - Algumas classes e funções utilitárias (ex.: excepções)
* Luis Correia
  + Desenvolvimento da documentação
    - Doxygen
    - Diagrama de classes
    - Midfielder
    - Forward

Parte 2:

* João Furriel Pinheiro
  + Novos membros-dado nas classes necessárias para registar a performance dos jogadores
  + BST para ranking de performance de jogadores:
    - Estruturação e inicialização da BST
    - Funcionalidades
  + Hash Table para criação de postais de aniversário de ex-jogadores e treinadores
* Leonardo Teixeira
  + BST para ranking de performance de jogadores:
    - Conceção do algoritmo de cálculo da performance dos jogadores.
  + Fila de prioridade para notificação de atraso dos ECG’s dos jogadores
  + Implementação das novas funcionalidades nos Menus de interação com o Utilizador
* Luis Correia
  + Desenvolvimento da documentação
    - Doxygen
    - Diagrama de classes