

Trabalho AEDS III - CRUD Clubes de Futebol

Davidson Marra¹, Pedro Henrique Teixeira²

¹PUC Minas – Pontifícia Universidade Católica de Minas Gerais (PUCMG)
30.535-901 – Belo Horizonte – MG – Brazil

Abstract. *This work aims to show the application of an algorithm that controls data through secondary memory (text file). As an application model a CRUD scheme was used for football clubs, whereby they can add points and play matches against each other.*

Resumo. *Esse trabalho têm como objetivo mostrar a aplicação de um algoritmo que controla os dados por meio da memória secundária (arquivo de texto). Como modelo de aplicação foi utilizado um esquema de CRUD para clubes de futebol, pelo qual eles podem somar pontos e disputar partidas entre si.*

1. Introdução

Com o desenvolvimento dessa aplicação, foi possível ver como é na prática o funcionamento de um algoritmo que trabalhe com arquivos de texto. Para a elaboração do CRUD dos clubes e leitura dos arquivos, foi utilizado a linguagem Java.

2. Desenvolvimento

Para o desenvolvimento da aplicação, foi separado em arquivos para melhor organização e aproveitamento do código. Esses arquivos foram: CRUD, Clube, Interface Entidade e Menu.

2.1. Clube

O arquivo Clube, representa a classe que será a entidade/registo do projeto. Nela estão as propriedades de um Clube, pelo qual são: Byte id, pois por se tratar de uma simulação de um campeonato brasileiro o número de times não pode passar de 20. String nome. String cnpj. String cidade. Byte partidasJogadas, pois o número de partidas não pode passar de 38. Byte pontos, pois um time não pode passar de 114 pontos. Além das propriedades, ele contém 4 métodos. O construtor, pelo qual inicializa as propriedades do objeto. O método toString, onde printa as informações de um clube. O procedimento toByteArray, pelo qual retorna um array de bytes preenchidos para escrever no arquivo. E o método fromByteArray, pelo qual lê o arquivo e salva as informações em um array de Bytes.

2.2. CRUD

O arquivo CRUD, armazena todos os métodos que vão gerenciar as operações feitas no arquivo. Elas são: Create, Read, Delete e Update.

2.3. Interface Entidade

A interface Entidade tem como objetivo determinar os métodos que obrigatoriamente devem ser implementados na classe Clube.

2.4. Menu

A classe Menu, foi criada para desenvolver a interface gráfica do aplicativo e fazer as chamadas dos métodos do CRUD. Além de armazenar o método para jogar uma partida, onde é informado o id dos dois times que iram se enfrentar e o placar da partida. Após isso é atualizado o número de pontos e a quantidade de partidas jogadas dos times.

3. Testes e Resultados

Os testes foram feitos em cada procedimento do CRUD, pelo qual é verificado se está certo conforme o resultado final do arquivo de texto que armazena os dados.

3.1. Método Criar

Para utilizar o método criar é necessário ter a pasta "dados" criada na raiz do projeto. Ao selecionar a opção 1 no menu, o próximo passo é preencher as informações do clube, da seguinte maneira:

```
Opcao:
1
Você entrou no método criar.
Entre com o nome do Clube:
Atlético
Entre com o cnpj do Clube (sem máscara):
1234567
Entre com o cidade do Clube (sem máscara):
Belo Horizonte
TIME CRIADO COM SUCESSO!
```

Figure 1. Time criado com sucesso

Após o time ser criado de maneira correta, o arquivo "Clube.db" será possível de se acessar dentro da pasta "dados", e nele estão armazenados as informações dos times da seguinte maneira:

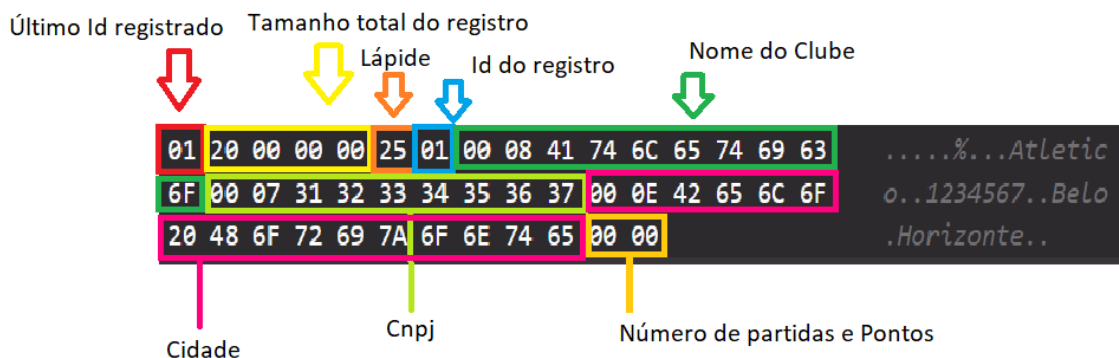


Figure 2. Organização arquivo de dados

3.2. Método Listar

Para fazer a listagem das informações de um clube, é necessário passar a opção 2 no menu. E depois informar o id do clube que deseja ser achado, o retorno deve ser:

```

2
Você entrou no método pesquisar.
Entre com o id do Clube:
1
TIME ENCONTRADO:

Nome: Atletico
CNPJ: 1234567
Cidade: Belo Horizonte
Partidas Jogadas: 0
Pontos: 0

```

Figure 3. Retorno método Listar

3.3. Método Atualizar

Para atualizar os dados de um clube, no menu é selecionado a opção 3 e após isso é indicado o id do clube e as suas novas informações. Ao completar as informações, caso tudo dê certo é printado os novos dados do clube, da seguinte forma:

```

Opcao:
3
Você entrou no método atualizar.
Entre com o id do Clube:
1
Entre com o nome do Clube:
Vila Nova
Entre com o cnpj do Clube (sem máscara):
1234
Entre com o cidade do Clube (sem máscara):
Nova Lima

Nome: Vila Nova
CNPJ: 1234
Cidade: Nova Lima
Partidas Jogadas: 0
Pontos: 0
TIME ALTERADO

```

Figure 4. Método Atualizar

Caso as informações do novo clube sejam maiores que a do clube antigo, o arquivo primeiro vai marcar o antigo como excluído e após isso escrever as informações do novo com o mesmo id, da seguinte forma:

```

01 2a 00 00 00 11 01 00 04 67 61 6c 6f 00 03 31 .*.....galo..1
32 33 00 02 42 48 00 00 20 00 00 00 2b 01 00 0d 23..BH.. ...+...
56 61 73 63 6f 20 64 61 20 47 61 6d 61 00 08 31 Vasco da Gama..1
32 33 34 35 36 37 38 00 0e 52 69 6f 20 64 65 20 2345678..Rio de
4a 61 6e 65 69 72 6f 00 00 Janeiro..

```

Figure 5. Método Atualizar com infos maiores

3.4. Método Excluir

O método excluir pode ser utilizado selecionando a opção 4 no menu e passando o id do clube que deseja ser excluído, dessa forma:

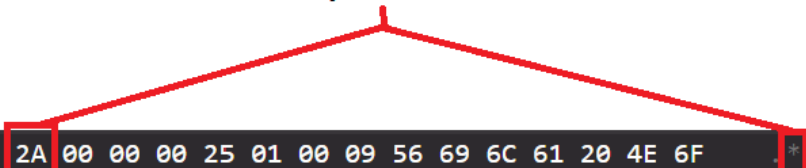
```
Opcao:
4
Você entrou no método Exclui.
Entre com o id do Clube:
1
TIME EXCLUIDO:

Nome: Vila Nova
CNPJ: 1234
Cidade: Nova Lima
Partidas Jogadas: 0
Pontos: 0
```

Figure 6. Método Excluir

No registro do clube excluído é trocado o tamanho do arquivo por um "*" para indicar que ele não existe mais, ficando da seguinte forma:

Indicação de clube excluído



```
01 2A 00 00 00 25 01 00 09 56 69 6C 61 20 4E 6F  * ..%...Vila.No
76 61 00 04 31 32 33 34 00 09 4E 6F 76 61 20 4C  va..1234..Nova.L
69 6D 61 00 00 7A 6F 6E 74 65 00 00             ima..zonte..
```

Figure 7. Método Excluir

3.5. Método Jogar Partida

Para utilizar o método Jogar Partida, é necessário escolher a opção 5 no menu e já ter no banco 2 clubes criados. Após isso, é informado para o programa quais os times irão se enfrentar e o placar do jogo, da seguinte forma:

```
Opcao:
5
Aqui realizamos partidas entre dois times.
Entre com o id do primeiro clube:
1
Entre com o id do segundo clube:
2
PLACAR:
Vasco da Gama:
1
Atlético:
4
O clube Atlético ganhou! Ele faz 3 pontos e o clube Vasco da Gama 0 pontos!
```

Figure 8. Informações completas para método Jogar Partida

E para completar, é feita uma atualização com os novos valores de pontos e partidas jogadas de cada clube.

```
Nome: Vasco da Gama
CNPJ: 12345678
Cidade: Rio de Janeiro
Partidas Jogadas: 1
Pontos: 0
TIME ALTERADO

Nome: Atlético
CNPJ: 12345
Cidade: Belo Horizonte
Partidas Jogadas: 1
Pontos: 3
TIME ALTERADO
```

Figure 9. Informações atualizadas pós partida jogada

4. Conclusão

Podemos concluir que com esse trabalho, que desenvolver aplicações com memória secundária é muito interessante e necessário, pois garantimos consistência e persistência dos

dados. Fizemos um sistema de CRUD simples que poderia muito bem ser modificado para atender necessidades pessoais e no mercado de trabalho, em que uma empresa poderia ter por exemplo um sistema para gerenciamento de equipamentos, em que um certo funcionário pode realizar ações de criação, atualização, pesquisa e deletar certo equipamento. Então, por meio desse trabalho aprendemos as operações básicas em um sistema que vai armazenando cada dado em memória secundária. Outra vantagem de se trabalhar em memória secundária é que se por algum motivo ocorresse falta de energia durante a execução do programa nenhum dado seria perdido, já que quando for reiniciado será necessário apenas recarregar o arquivo. Por fim, ainda aprendemos muito sobre a linguagem Java em si, que nunca tínhamos usado antes. Java é uma linguagem que lida muito bem com o fluxo de arquivos, e é umas das linguagens mais usadas no mercado de trabalho atualmente, por isso é tão importante saber sua sintaxe e ter projetos com ela.