



Departamento de Informática

Escola de Engenharia

Universidade do Minho

Gestor de atividades e planos de treino

Trabalho Prático

Programação Orientada aos Objetos

Licenciatura em Engenharia Informática

2023/2024



Eduardo Faria, a104353



Nuno Silva, a104089



Ana Cerqueira, a104188

Grupo 4

11/05/2024

Índice

Índice.....	2
Introdução.....	3
Diagrama de classes	3
Estrutura do projeto	4
Testes Funcionais	4
Classes Base	4
1. Utilizador	4
1.1. Utilizador	4
1.2. TiposUtilizador	5
2. PlanoDeTreino.....	5
3. Atividade.....	5
3.1. Atividade.....	5
3.2. Distancia.....	6
3.3. Altimetria	6
3.4. Repeticoes	6
3.5. Pesos.....	6
Funcionamento.....	6
1. Início.....	6
2. Registo/Login.....	7
3. Menu principal de Manipulação de Atividades e Plano de Treino.....	7
4. Salto no Tempo.....	8
5. Gerar um Plano de Treino com Objetivos	8
6. Estatísticas gerais.....	9
Conclusão.....	9

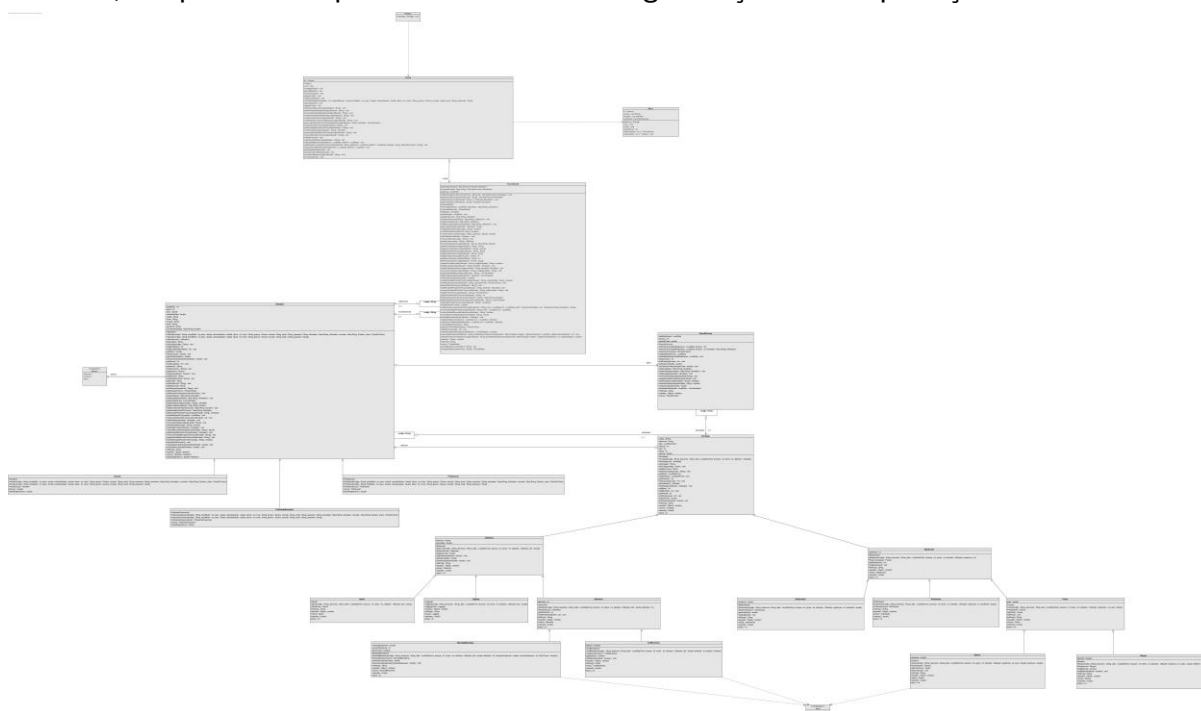
Introdução

No contexto da unidade curricular Programação Orientada aos Objetos fomos desafiados a desenvolver uma aplicação que realiza a gestão das atividades e planos de treino dos seus utilizadores.

Realizando uma breve apresentação às funcionalidades que desenvolvemos, a nossa aplicação contém um sistema de *login*, que permite que cada utilizador tenha as suas atividades físicas devidamente organizadas, juntamente com os respetivos dados. Cada utilizador consegue registar ou agendar uma qualquer atividade a qualquer momento e tem a possibilidade de criar um plano de treino automático adaptado aos seus objetivos. Um utilizador pode aceder ao painel de estatísticas, onde encontra informação útil sobre o próprio ou sobre os outros. Para além disso, é capaz de consultar os recordes pessoais ou dos diferentes utilizadores, de modo a suscitar um ambiente de competição. O utilizador pode ainda desencadear um “salto no tempo”, de modo a simular o passar do tempo, atualizando devidamente os diversos registos da aplicação.

Diagrama de classes

De forma a organizar todo o desenvolvimento do projeto, dedicamos, inicialmente, grande parte do tempo a estruturar, de forma precisa e cuidadosa, o diagrama de classes, capaz de representar toda a organização da aplicação desenvolvida.



Estrutura do projeto

No desenvolvimento deste projeto fizemos uso do padrão arquitetural MVC *delegate* de modo a assegurar uma separação clara entre os métodos de apresentação e interação com os utilizadores e os métodos que envolvem a manipulação do API da aplicação.

Para esta implementação criámos as classes *FitnessModel*, *TextUI* e *Menu*:

- ***FitnessModel***: responsável pela gestão de toda a parte computacional da aplicação e pela resposta aos pedidos de manipulação de dados. Nesta classe estão definidos os métodos que fazem a ligação entre as restantes classes e que permitem o funcionamento da aplicação.
- ***TextUI***: nesta classe está definida toda a interface não geral (que não é menu) que interage com o utilizador, assim como todos os métodos que realizam pedidos à classe *FitnessModel* através de funções lambda, proporcionadas pelas interfaces funcionais *Handler* e *PreCondicao*.
- ***Menu***: nesta classe está definida toda a interface geral, responsável por apresentar os menus base com as diferentes opções e por ler a opção escolhida pelo utilizador, executando o método correspondente à mesma. Isto é possível pois na classe *Menu* encontram-se duas listas que guardam objetos das interfaces funcionais mencionadas anteriormente. Estes objetos serão criados e adicionados nas listas na classe *TextUI*, conforme as necessidades.

Testes Funcionais

O projeto conta com diversos testes funcionais direcionados a todos os métodos da classe *PlanoDeTreino* e subclasses de *Atividade* e *Utilizador*. Estes testes são uteis para garantir o correto funcionamento dos métodos das classes em questão de uma forma simples e automática.

Classes Base

1. Utilizador

1.1. Utilizador

A classe *Utilizador* é uma classe abstrata que funciona como modelo para a definição dos diferentes utilizadores. Assim, ela apresenta um conjunto de

variáveis de instância que representam as suas principais características e um conjunto de métodos que permitem a sua manipulação.

Temos ainda a classe *enum Genero* que define os diferentes géneros que podem ser atribuídos a um utilizador.

1.2. TiposUtilizador

A nossa aplicação suporta vários tipos de utilizador, i.e., subclasses de *Utilizador*, tendo neste momento a implementação de 3 subclasses: *PraticanteOcasional*, *Amador* e *Profissional*. Esta distinção visa adequar o cálculo de certos dados, tornando-o mais preciso e correto para cada perfil. Por exemplo, o tipo de utilizador fará diferença na contabilização calórica. Nesta classificação consideramos que o *PraticanteOcasional* é um utilizador que pratica desporto pontualmente, apenas por lazer, o *Amador* já exibe uma prática desportiva mais regular e o *Profissional* apresenta uma carga de treino bastante significativa.

2. PlanoDeTreino

A classe *PlanoDeTreino* define a estrutura dos planos de treino dos utilizadores, sendo que a cada plano está associada uma data de início, a sua duração em semanas, as calorias que o mesmo totaliza e a lista de atividades a realizar inseridas no mesmo.

3. Atividade

3.1. Atividade

A classe abstrata *Atividade*, que serve como base para a definição das diferentes atividades da nossa aplicação, apresenta um conjunto de atributos que caracterizam as atividades, assim como os respetivos métodos para a sua manipulação.

No desenvolvimento desta aplicação, tivemos em conta a existência de diferentes tipos de atividades e os diferentes parâmetros a considerar em cada um. Assim, as nossas atividades dividem-se em subclasses de *Atividade* mais gerais, *Repeticoes* e *Distancia*, ambas abstratas, podendo ainda existir atividades de repetições com peso e atividades de distância e altimetria, que originam, respetivamente, as subclasses abstratas *Pesos* e *Altimetria*.

3.2. Distancia

A classe abstrata *Distancia* contém nela o parâmetro distância (km) e velocidade (km/h). Esta subclasse de *Atividade* é estendida, por exemplo, pela atividade *Sprint* e *Jogging*.

3.3. Altimetria

A classe abstrata *Altimetria* contém nela o parâmetro altimetria (m). Esta subclasse de *Distancia* é estendida, por exemplo, pela atividade *BicicletaMontanha* e *TrailMontanha*.

3.4. Repeticoes

A classe abstrata *Repeticoes* contém nela o parâmetro repetições. Esta subclasse de *Atividade* é estendida, por exemplo, pela atividade *Abdominais* e *Polichinelo*.

3.5. Pesos

A classe abstrata *Pesos* contém nela o parâmetro peso (kg). Esta subclasse de *Repeticoes* é estendida, por exemplo, pela atividade *Supino* e *Biceps*.

Para a implementação desta forma de definição de classes é fundamental recorrer ao uso da hierarquia de classes. Desta forma, torna-se simples acrescentar novas atividades sem grande alteração no código das classes existentes.

Funcionamento

1. Início

Quando no menu inicial, existe a possibilidade de carregar um ficheiro que contenha um estado já definido ou guardar o estado em que a aplicação se encontra. Caso optemos pela criação de uma nova simulação, todos os parâmetros da aplicação são redefinidos.

```
=====Menu=====
1 - Carregar Estado
2 - Nova Simulação
3 - Guardar Estado
0 - Sair

Opção: █
```

2. Registo/Login

Para a utilização da aplicação é crucial todos os dados do utilizador estarem devidamente definidos e para isso foi feito um sistema de registo/login. Caso seja a primeira utilização da aplicação será pedido que efetue o registo preenchendo alguns campos como nome, nome de utilizador, password, peso, etc. Em utilizações futuras apenas será pedido o login com o nome de utilizador e password. Com estes parâmetros a aplicação carregará todos os dados que lhe estão associados (dados pessoais, histórico de atividades, etc.) e procederá ao seu funcionamento tendo em conta o utilizador em questão. As funcionalidades de login e apagar conta apenas estão disponíveis quando existe, pelo menos, um utilizador na aplicação.

<pre>=====Menu===== 1 - Registrar 2 - Login 3 - Apagar Conta 0 - Sair Opção: █</pre>	<pre>=====Menu===== 1 - Registrar 2 - --- 3 - --- 0 - Sair Opção: █</pre>
---	--

3. Menu principal de Manipulação de Atividades e Plano de Treino

Quando neste menu, o utilizador dispõe de diversas opções, desde registar atividades realizadas ou adicionar uma atividade ao plano de treino, até consultar a informação pessoal e ver estatísticas gerais. Das diversas opções, destacam-se a opção de criar um plano de treino automático, a opção de saltar no tempo e a opção de ver as estatísticas gerais, que inclui informação relativa às atividades realizadas e aos recordes pessoais e da aplicação.

Certas opções apenas ficam disponíveis quando o utilizador realizou pelo menos uma atividade ou quando o utilizador tem um plano de treino ativo.

```
=====Menu=====
1 - Informação Pessoal
2 - Adicionar uma atividade realizada
3 - ---
4 - ---
5 - Adicionar um plano de treino
6 - Criar plano de treino Automático
7 - ---
8 - ---
9 - ---
10 - ---
11 - ---
12 - ---
0 - Sair
Opção: █

=====Menu=====
1 - Informação Pessoal
2 - Adicionar uma atividade realizada
3 - Remover uma atividade realizada
4 - Mostrar todas as atividades realizadas
5 - Adicionar um plano de treino
6 - Criar plano de treino Automático
7 - Remover o plano de treino
8 - Adicionar uma atividade ao plano de treino
9 - Remover uma atividade do plano de treino
10 - Mostrar o plano de treino
11 - Ver estatísticas gerais
12 - Saltar no tempo
0 - Sair
Opção: █
```

4. Salto no Tempo

O salto no tempo é uma funcionalidade que pressupõe que todas as atividades agendadas por um utilizador no seu plano de treino serão concretizadas. Para isso, este "realiza" todas as atividades planeadas antes da nova data e adiciona-as ao histórico de atividades realizadas do utilizador. A par disto, atualiza as calorias, o peso e os recordes pessoais e gerais. Deste modo, dá-se a transição temporal para um tempo futuro pedido pelo utilizador e indicado em dias.

5. Gerar um Plano de Treino com Objetivos

A criação automática de um plano de treino começa por registar as calorias objetivo do utilizador, assim como o número máximo de atividades por dia que pretende efetuar. De seguida, é apresentado o menu que mostra os diferentes planos de treino disponíveis, cada um com o seu conjunto de atividades pré-definidas. Depois do utilizador escolher um plano, é questionado ao mesmo quantas vezes por semana pretende realizar cada atividade contida no plano, assegurando que as restrições do planeamento de atividades estão a ser cumpridas, sendo estas:

- Não haver mais do que uma atividade hard no mesmo dia, nem em dias consecutivos (não podem existir mais do que 3 atividades hard por semana);
- O número de atividades diárias nunca ultrapassa o número máximo de atividades por dia imposto pelo utilizador (nunca se excede 3 atividades por dia) e o número máximo de atividades numa semana não é ultrapassado;
- O consumo calórico mínimo é atingido;

```
Calorias objetivo: 6000
Número máximo de atividades por dia: 3

=====Menu=====
1 - Plano de treino de distância
2 - ---
3 - Plano de treino de repetições
4 - Plano de treino de repetições com pesos
5 - Plano de treino de distância com atividades HARD
6 - Plano de treino de distância e altimetria com atividades HARD
7 - Plano de treino de repetições com atividades HARD
8 - Plano de treino de repetições com pesos com atividades HARD
0 - Sair

Opção: 8
Quantas vezes deseja repetir a atividade Supino por semana? 2
Quantas vezes deseja repetir a atividade Biceps por semana? 3

[SUCESSO] Plano de treino criado!
```


6. Estatísticas gerais

Antes de entrar no menu de estatísticas gerais, é questionado ao utilizador se pretende consultar a informação num período específico ou desde sempre. Depois disso, o mesmo dispõe de várias opções que lhe permitem visualizar os seus dados relativos a atividades realizadas, assim como ver os recordes pessoais e gerais da aplicação.

```
Pretende ver num dado período ou desde sempre? (1 - período, 2 - sempre): 2
=====Menu=====
1 - Qual é o utilizador que mais calorias dispendeu num período ou desde sempre
2 - Quantas calorias é que dispendeu num período ou desde sempre
3 - Qual é o utilizador que mais actividades realizou num período ou desde sempre
4 - Quantos kms é que realizou num período ou desde sempre
5 - Quantos metros de altimetria é que totalizou num período ou desde sempre
6 - Quantos kg de peso é que levantou num período ou desde sempre
7 - Quantas repetições é que totalizou num período ou desde sempre
8 - Qual é o tipo de actividade mais realizada
9 - Qual é o plano de treino mais exigente em função do dispêndio de calorias proposto (Média semanal)
10 - Recordes individuais
11 - Recordes da aplicação
0 - Sair
Opção: █
```

Conclusão

Após o desenvolvimento da aplicação, podemos concluir que o nosso código apresenta-se bastante robusto a nível do tratamento de erros, modularidade e encapsulamento, respeitando o paradigma da Programação Orientada aos Objetos. Estas características tornam fácil a adição de novas classes, permitindo à aplicação crescer conforme necessário.

É importante salientar que decidimos usar o mecanismo de agregação em todas as classes, uma vez que, no contexto da aplicação, queríamos assegurar que as modificações dos dados se iriam refletir em todos os objetos da aplicação de forma direta.