

Universidade do Minho
Escola de Engenharia
Departamento de Informática

Programação orientada aos objetos

Trabalho prático

11 de maio de 2024

Desenvolvido por:

Diogo Esteves - a104004
Rodrigo Ferreira - a104531
Rodrigo Fernandes - a104175

Conteúdo

1 Introdução

2 Arquitetura da aplicação

- 2.1 A classe "Main"
- 2.2 A classe "Menu"
- 2.3 A classe "McFit"
- 2.4 A classe "Utilizador"
 - 2.4.1 As sub-classes da classe "Utilizador"
- 2.5 A classe "Atividade"
 - 2.5.1 A classe "Corrida"
 - 2.5.2 A classe "Ciclismo"
 - 2.5.3 A classe "LevantamentoPesos"
 - 2.5.4 A classe "Pilates"
- 2.6 A classe "Plano Treino"
- 2.7 A interface "Hard"

3 Aplicação

- 3.1 Gestão de utilizadores
- 3.2 Gestão de atividades
- 3.3 Gestão de planos de treino
- 3.4 Estatísticas
- 3.5 Gestão de datas

4 Conclusão

1 Introdução

O objetivo deste relatório é fazer uma descrição do trabalho realizado, com o objetivo de facilitar a compreensão do código por detrás do trabalho desenvolvido. Para isso vamos fazer uma breve descrição da arquitetura de classes utilizada, e faremos uma breve descrição acerca das mesmas. No final vamos fazer uma pequena reflexão acerca de alguns testes por nós realizados.

2 Arquitetura da aplicação

A arquitetura de classes da nossa aplicação pode ser resumida no diagrama de classes disponível no GitHub onde o restante do projeto se encontra. Quanto à escolha das classes necessárias, eis uma breve explicação acerca das mesmas:

2.1 A classe "Main"

A classe main é a classe que inicia o programa. Esta classe começa por iniciar o construtor da classe menu e inicia um scanner. De seguida inicia o menu iniciar, para disponibilizar as opções de login e sign-in. Após o login, é chamado o método que desenha o menu principal e é a partir deste que o resto do programa de desenrola.

2.2 A classe "Menu"

A classe menu é a classe responsável por duas funcionalidades cruciais: a interação com o utilizador e o controlo do fluxo do programa. É através dos métodos desta classe que os diferentes menus da aplicação são desenhados no ecrã. Além disso, é nesta classe que o utilizador decide o que o programa deve fazer.

2.3 A classe "McFit"

A classe McFit é a classe que faz a ligação entre a classe menu e as classes utilizador, atividade e plano treino. É aqui que são guardados os mapas dos utilizadores da aplicação e a lista de atividades suportadas.

Além do referido anteriormente, a classe Mcfit é responsável pelo processamento dos outputs solicitados pelo utilizador na classe menu, ou seja, é esta classe que, por exemplo, processa as estatísticas do sub-menu de estatísticas da aplicação.

Por fim, esta é a classe responsável pela salvaguarda do estado da aplicação, ou seja, é a classe que guarda e lê do ficheiro a informação acerca dos utilizadores e atividades.

2.4 A classe "Utilizador"

A classe utilizador é uma super-classe das classes utilizador ocasional, utilizador amador e utilizador profissional. Nesta estão definidas algumas características que são comuns a todas as sub-classes referidas.

A classe utilizador é ainda uma classe abstrata, uma vez que nesta estão definidos dois métodos abstratos: o método clone e o método de fatorMultiplicativo, que são definidos nas respetivas sub-classes. O método fatorMultiplicativo define um fator multiplicativo que depende da preparação de cada utilizador e que é utilizado no cálculo das calorias que cada utilizador queimou na realização de uma atividade/plano de treino.

2.4.1 As sub-classes da classe "Utilizador"

As sub-classe utilizador ocasional, utilizador amador e utilizador profissional especializam a super-classe utilizador. Esta especialização é dada através do fator multiplicativo que varia de classe para classe

2.5 A classe "Atividade"

À semelhança da super-classe utilizador, a super-classe atividade é uma classe onde estão definidas características comuns às suas sub-classes corrida, ciclismo, LevantamentoPesos e pilates.

Esta é uma classe abstrata uma vez que também contém métodos abstratos: o método de clone e o método de calculaCaloriasGastas. Estes estão definidos nas sub-classes da classe atividade.

2.5.1 A classe "Corrida"

A classe corrida especializa a classe atividade no que toca a características de uma atividade de corrida: nela estão definidas a distância percorrida e o ritmo médio de corrida. É através destas variáveis de instância que é feito o cálculo das calorias gastas por este tipo de atividade.

Esta subclasse implementa ainda a interface "Hard" para que se possam tirar conclusões acerca da dificuldade da atividade.

2.5.2 A classe "Ciclismo"

Assim como a classe corrida, a classe ciclismo especializa a classe atividade. Nesta estão definidas variáveis de instância que caracterizam uma atividade relacionada com o ciclismo: a distância percorrida e a diferença de altimetria realizada, usadas no cálculo das calorias.

Tal como qualquer sub-classe de atividade, esta implementa a interface "Hard"

2.5.3 A classe "LevantamentoPesos"

A classe LevantamentoPesos estende a classe atividade, no que toca a atividades de ginásio que envolvam levantamento de cargas, através de séries e repetições. Para isso, são definidas as seguintes variáveis de instância: a carga, o número de repetições e de séries, usadas no cálculo das calorias.

À semelhança das demais sub-classes de atividade, esta classe implementa a interface "Hard"

2.5.4 A classe "Pilates"

A última sub-classe da classe atividade é a classe pilates. Nela estão definidas variáveis relativas a uma atividade que envolva repetições e séries, mas sem levantamento de carga: número de séries e repetições, que são usados para cálculo das calorias.

Tal como qualquer sub-classe de atividade, esta classe também implementa a interface "Hard".

2.6 A classe "Plano Treino"

A classe plano treino é a classe responsável pelo processamento de todos os planos de treino que um utilizador realiza. Na classe utilizador está definido dois

2.7 A interface "Hard"

As classes que implementam esta interface são obrigadas a implementar o método isHard. Este determina se uma atividade é ou não do tipo hard, a partir de um certo conjunto de características específicas à atividade

3 Aplicação

A aplicação "McFit" é um sistema de gestão de atividades físicas e planos de treino, projetado para um ambiente de academia. Ele permite aos usuários registrarem suas atividades físicas, visualizarem atividades disponíveis, gerarem planos de treino personalizados e monitorarem seu progresso ao longo do tempo. Para isto foi muito importante o uso dos métodos `guardar()` e `carregar()` (`Serializable`), ou seja, todas as informações dos utilizadores, das atividades e dos planos de treino em uma sequência de bytes que pode ser posteriormente armazenada em um arquivo.

3.1 Gestão de utilizadores

A nossa aplicação permite a adição de novos utilizadores ao sistema, seguido de um login seguro de utilizadores existentes com validação de senha.



Figura 1: Mcfit Menu Inicial

3.2 Gestão de atividades

A nossa aplicação permite, a partir de um login, a adição de novas atividades físicas ao sistema, bem como o registro de atividades realizadas por utilizadores e para além disso, a visualização das atividades disponíveis.

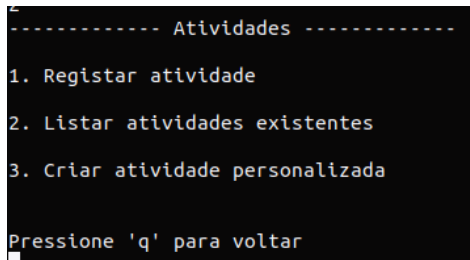


Figura 2: Gestão de atividades

3.3 Gestão de planos de treino

A nossa aplicação permite gerar automaticamente planos de treino com base nas preferências e nos objetivos do utilizador. Se preferir, a criação manual de planos de treino com especificações personalizadas e ainda consegue tirar partido do seu desenvolvimento pessoal ao consultar as suas informações em "Conta".

```
----- Planos Treino -----  
1.Gerar um novo plano de treino  
2.Criar um plano de treino customizado  
3.Pressione 'q' para voltar
```

Figura 3: Gestão de planos de treino

3.4 Estatísticas

Na análise de dados do nosso sistema, é permitido identificar o tipo de atividade mais realizada por um utilizador em específico, identificar o recorde do utilizador com mais calorias queimadas, identificar o utilizador com mais atividades realizadas, o cálculo da distância percorrida, altimetria total e ainda saber qual o plano mais exigente em função do dispêndio de calorias proposto.

```
----- Estatísticas -----  
  
Record de calorias: rodrigo(340.3259737672474kcal)  
Record de atividades realizadas: rodrigo(1 atividade(s))  
Atividade mais realizada: Corrida  
Distância total percorrida: 7.0km  
Altimetria total: 0.0m  
  
Plano de treino mais exigente: ->Lista atividades: [  
->Nome: BTT  
-Tipo: DistanciaAltimetria  
-Tempo Gasto: 44  
-Data: 2024-09-09 (Monday)  
-Distancia: 0.7131919779270091  
-Altimetria: 1.0  
-Dificuldade: Normal,  
->Nome: BTT  
-Tipo: DistanciaAltimetria  
-Tempo Gasto: 30  
-Data: 2024-09-09 (Monday)  
-Distancia: 1.6819199573800714  
-Altimetria: 1.0  
-Dificuldade: Normal,  
->Nome: BTT  
-Tipo: DistanciaAltimetria  
-Tempo Gasto: 25  
-Data: 2024-09-09 (Monday)  
-Distancia: 2.515727994333462  
-Altimetria: 3.0  
-Dificuldade: Normal]  
  
->Interações: 3  
->data: 2024-09-09  
->Veze por semana: 3
```

Figura 4: Estatísticas (exemplo)

3.5 Gestão de datas

A nossa aplicação permite o avanço da data atual para permitir o processamento de atividades e planos futuros.

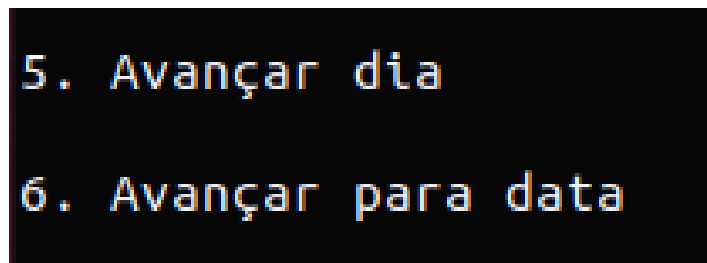


Figura 5: Gestão das datas

4 Conclusão

Para concluir este trabalho, é importante ressaltar a implementação bem-sucedida de um sistema abrangente da gestão de atividades físicas e planos de treino. Através da aplicação "McFit", desenvolvida pelos autores deste relatório, os utilizadores têm acesso a uma plataforma que não só permite o registo e acompanhamento das suas atividades físicas, mas também facilita a criação e gestão de planos de treino personalizados.

Ao longo deste documento, foram apresentadas as diferentes classes e a arquitetura subjacente ao sistema, evidenciando a sua organização e estruturação lógica. Além disso, foram destacadas as principais funcionalidades da aplicação, incluindo a gestão de utilizadores, atividades e planos de treino, bem como a análise estatística dos dados recolhidos.

É importante salientar que a implementação de funcionalidades como a gestão de datas e a persistência de dados através do uso do método `guardar()` e `carregar()` (`Serializable`) contribuem significativamente para a usabilidade e robustez do sistema.

Em suma, o trabalho demonstra não só um conhecimento sólido dos princípios de programação orientada a objetos, mas também uma capacidade eficaz de aplicar esses conceitos na resolução de problemas do mundo real. A aplicação "McFit" representa um recurso valioso para indivíduos que procuram melhorar a sua condição física e acompanhar o seu progresso ao longo do tempo.