



Unidade Curricular de Programação Orientada aos Objetos

Licenciatura em Engenharia Informática

Ano Letivo 2023/2024

Filipa Oliveira da Silva

(A104167)



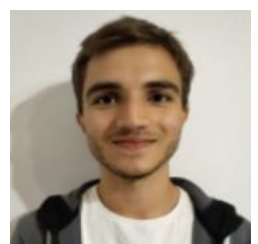
Maria Cleto Rocha

(A104441)



Mário Rafael Figueiredo da Silva

(A104182)



Índice

1. Introdução	3
2. Arquitetura de Classes	4
3. Descrição Da Aplicação Desenvolvida.....	7
4. Conclusões	8

Introdução

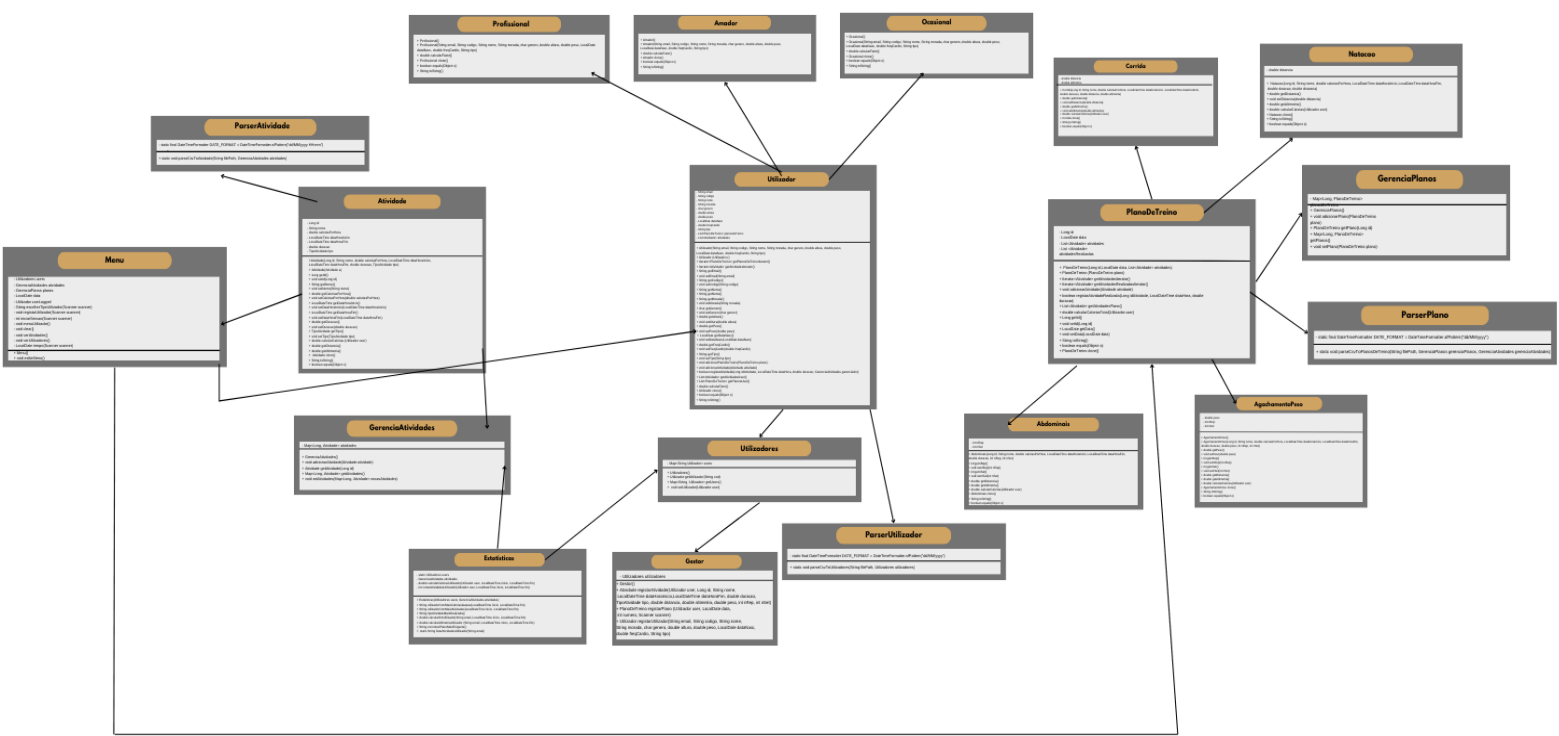
Este relatório serve de apoio ao projeto da unidade curricular de Programação Orientada aos Objetos, desenvolvido pelo nosso grupo, composto por: Filipa Oliveira da Silva (a104167), Maria Cleto Rocha (a104441) e Mário Rafael Figueiredo da Silva(a104182). Foi-nos proposta a criação de uma aplicação, em linguagem Java, que permita fazer a gestão de atividades e planos de treino, de praticantes de atividades físicas.

Desde já achamos o projeto extremamente aliciante e interessante o que motivou, e de certa forma facilitou, o seu desenvolvimento.

Neste documento, iremos expor e resumir uma boa parte do nosso projeto, explicando os passos e o raciocínio que seguimos de maneira a concluir o mesmo. Iremos também numerar tanto as nossas dificuldades e dúvidas no decorrer do trabalho como os aspetos em que poderíamos melhorar.

Arquitetura de Classes

Como já foi referido, este projeto tem como propósito a criação de uma aplicação que gere as atividades e planos de treinos de praticantes de atividades físicas. O objetivo principal da mesma é o de permitir que cada utilizador registre a realização das atividades, permitindo depois que se obtenha vários indicadores relativos ao treino de cada um.



Decidimos seguir uma abordagem mais simples e intuitiva no que toca à definição das classes utilizadas.

Primeiramente começamos por fazer um esboço com o objetivo de nos orientar e de certa forma ajudar a visualizar o projeto como um todo, simplificando assim, o desenvolvimento e progressão das classes e respectivas variáveis de instância.

A aplicação é composta por várias classes que representam diferentes aspetos de um sistema de gestão de atividades físicas, incluindo o registo de utilizadores, atividades e planos de treino.

No nosso código existem 3 classes principais que definem as características essenciais para a execução e desenvolvimento da nossa aplicação.

- **'Atividade'**: Classe abstrata base para todos os tipos de atividades físicas. Contém atributos como **'id'**, **'nome'**, **'caloriasPorHora'**, **'dataHoralInicio'**, **'dataHoraFim'**, **'duracao'** e **'tipo'** (enum **'TipoAtividade'**), além de métodos para cálculo de calorias, obtenção de distância e altimetria;
- **'Utilizador'**: Classe abstrata que define os atributos e métodos comuns a todos os tipos de utilizadores do sistema, como **'email'**, **'codigo'**, **'nome'**, **'morada'**, **'genero'**, **'altura'**, **'peso'**, **'dataNasc'**, **'freqCardio'**, **'tipo'** e listas de **'planosDeTreino'** e **'atividades'**;
- **'PlanoDeTreino'**: Representa um conjunto de atividades planejadas para serem realizadas em uma data específica. Inclui métodos para adicionar atividades e calcular o total de calorias consumidas.

Associadas às mesmas existem simultaneamente:

- **Classes Derivadas de Atividade:**
 - **'Abdominais'**, **'AgachamentoPeso'**, **'Corrida'**, **'Natacao'** que especificam diferentes tipos de atividades, adicionando atributos relevantes como **'peso'**, **'distancia'** e **'altimetria'**, e implementam de maneira específica o cálculo de calorias baseado no tipo de atividade.
- **Classes Derivadas de Utilizador:**
 - **'Amador'**, **'Profissional'**, **'Ocasional'** que são tipos específicos de utilizadores que estendem **'Utilizador'**, diferindo principalmente no método **'calcularFator()'**, que ajusta o cálculo de calorias queimadas de acordo com o tipo de usuário.
- **Classes usadas para o Gerenciamento de Dados:**
 - **'GerenciaAtividades'**, **'GerenciaPlanos'**, **'Utilizadores'** que são classes responsáveis por gerenciar coleções de atividades, planos de treino e utilizadores, respetivamente, permitindo operações como adicionar, remover e procurar por um ID ou e-mail.

- **Classes Auxiliares:**

- **‘Estatísticas’**, que fornece métodos para análise de dados, como, por exemplo, encontrar o user que mais gastou calorias ou o que realizou atividades num determinado período;
- **‘ParserAtividade’**, **‘ParserPlano’** e **‘ParserUtilizador’**, são classes para parsing de arquivos CSV usados para carregar atividades, planos e usuários no sistema;
- **‘Menu’**, responsável por exibir as opções do menu interativas ao utilizador. Inclui métodos para exibir diferentes menus de acordo com o tipo de user (amador, profissional e ocasional) e captura a entrada do utilizador para navegar pelas várias funcionalidades da aplicação;
- **‘Visual’**, que é usada para gerenciar a representação visual das informações na aplicação, como a exibição de gráficos de estatísticas ou a formatação de saída de dados em tabelas. É fundamental para garantir que a interface do utilizador seja clara e acessível.

As classes foram criadas para refletir os componentes lógicos do domínio da aplicação – utilizadores, atividades e planos de treino – e para permitir a expansão futura do sistema, como a adição de novos tipos de atividades ou métricas de desempenho.

As relações entre as classes permitem uma gestão integrada desses componentes, facilitando operações como o cálculo de estatísticas e a geração de relatórios detalhadas.

Descrição da Aplicação Desenvolvida

A aplicação oferece, como um todo, funcionalidades para:

- **Registo e gestão de usuários e atividades:** Os utilizadores podem ser de diferentes tipos (amador, profissional, ocasional) e cada um pode registar múltiplas atividades físicas;
- **Criação e gestão de planos de treino:** Os utilizadores podem criar planos de treino, que são coleções de atividades planeadas para datas específicas;
- **Cálculo de calorias queimadas:** A aplicação calcula as calorias queimadas por atividade, ajustadas pelo tipo de utilizador e especificidades da atividade;
- **Relatórios e Estatísticas:** Fornece relatórios detalhados sobre as atividades e sobre o desempenho dos utilizadores, como, por exemplo, o utilizador que mais queimou calorias ou o que realizou atividades e progressos em planos de treino, oferecendo deste modo insights importantes sobre a eficiência dos treinos.

Descrição dos Menus da Aplicação

- **Menu Principal**

O Menu Principal serve como o ponto de entrada para a aplicação, oferecendo opções para registo e login de utilizadores, bem como acesso aos menus específicos, dependendo do tipo de utilizador que tem acesso ao sistema.

O método '**mostrarMenuPrincipal**' exhibe as opções básicas para o usuário, permitindo registo, login ou saída do sistema. Cada opção chama um método correspondente que manipula a funcionalidade solicitada, facilitando a navegação inicial no sistema.

- **Menu do Utilizador**

O Menu do Utilizador aparece após o login bem-sucedido. Ele permite com que o utilizador gerencie as suas atividades e planos de treino, visualizar relatórios de progresso e atualizar seus dados pessoais.

O método utilizado personaliza o menu com base no usuário logado, mostrando as opções disponíveis como adicionar atividades ou visualizar e atualizar planos de treino. A estrutura facilita a interação do usuário com o sistema, promovendo uma experiência personalizada.

- **Menu de Estatísticas**

O Menu de Estatísticas é destinado à visualização de relatórios detalhados sobre as atividades e o desempenho do utilizador, como calorias queimadas, distâncias percorridas e comparações de desempenho ao longo do tempo. Este menu oferece uma análise aprofundada que pode ser crucial para o planeamento e ajuste de metas de fitness, bem como para manter os usuários motivados e informados sobre seu progresso. Sendo este menu uma ferramenta vital na aplicação, proporcionando aos usuários um feedback detalhado e motivador sobre suas atividades e progresso, e incentivando a continuidade e o ajuste de seus esforços de fitness.

A modularidade do sistema permite a fácil adição de novas funcionalidades ou tipos de atividades, enquanto o encapsulamento garante a segurança dos dados e a robustez da aplicação. O design intuitivo e a interface amigável facilitam a interação dos utilizadores com o sistema, tornando a gestão de atividades físicas uma tarefa simples e compensadora.

Conclusões

Este projeto envolveu o desenvolvimento de uma aplicação de software para a gestão de atividades e planos de treino de praticantes de atividades físicas. O sistema foi estruturado em torno de uma arquitetura de classes robusta, incluindo a definição e implementação de classes para utilizadores **'Utilizador'**, atividades **'Atividades'**, planos de treino **'PlanosDeTreino'**, e gerenciamento **'GerenciaAtividades'**, **'GerenciaPlanos'** e **'Utilizadores'**, bem como uma classe para análises estatísticas **'Estatisticas'**, e classes responsáveis pela parte interativa **'Menu'** e **'Visual'**.

Durante o desenvolvimento desta aplicação, deparemos-mos com diversas dificuldades, incluindo o gerenciamento complexo de múltiplas entidades relacionadas, o que exigiu uma implementação cuidadosa de classes mais importantes, de forma a manter a integridade dos dados. A utilização de herança e classes abstratas para comportamentos comuns e especializados apresentou desafios na implementação de métodos contextuais como o cálculo de calorias, que varia entre tipos de atividades e usuários.

Em suma, o desenvolvimento desta aplicação foi uma excelente oportunidade para aplicar conceitos avançados de programação orientada a objetos.

A aplicação final conseguiu atender os requisitos que estabelecemos previamente, oferecendo uma solução eficaz para a gestão de atividades físicas e planos de treino. As experiências adquiridas e as soluções implementadas para superar as dificuldades encontradas serão de grande importância para projetos futuros, permitindo uma melhor preparação e abordagem para desafios semelhantes.