



Escola de Engenharia  
Departamento de Informática

Licenciatura em Engenharia Informática

# Projecto Java - FitnessUM

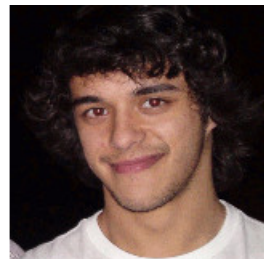
Programação Orientada aos Objectos



69854  
João Mano



66822  
Miguel Guimarães



69303  
Bruno Torres

Braga, Junho de 2014

# Conteúdo

<b>1</b>	<b>Estrutura da aplicação</b>	<b>2</b>
1.1	Actividades . . . . .	2
1.1.1	Classe abstracta Activity . . . . .	3
1.1.2	Indoor,Outdoor e actividades desportivas . . . . .	3
1.1.3	Comparadores . . . . .	3
1.2	Utilizadores . . . . .	3
1.3	Classe Abstracta Person . . . . .	3
1.4	Eventos . . . . .	4
1.5	Classe abstracta Event . . . . .	4
1.6	. . . . .	4

# 1 Estrutura da aplicação

## 1.1 Actividades

Foram definidas as seguintes actividades desportivas para a nossa aplicação:

- Yoga
- Aerobics
- Swimming
- IndoorCycling
- Handball
- Basketball
- TableTennis
- Boxing
- Badminton
- VolleyBallIndoor
- Football
- VolleyBallBeach
- Running
- Skating
- Sailing
- Walking
- Tennis
- Skiing
- Cycling
- MountainBiking
- Orienteering
- Snowboarding
- Polo

Para a implementação destas actividades foi usada a seguinte estrutura:

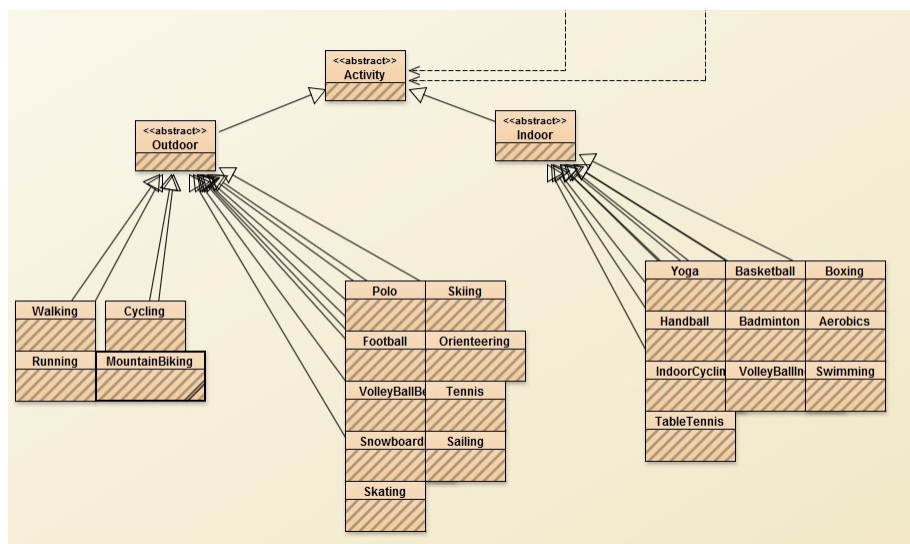


Figura 1: Estrutura das actividades

### 1.1.1 Classe abstracta Activity

Esta é a classe mais abstrata, que tem a fundação de todas as actividades desta aplicação. Contém variáveis comuns a todas as actividades:

*name*, *date*, *timeSpent* e *calories* tal como os construtores, *getters* e *setters*.

### 1.1.2 Indoor,Outdoor e actividades desportivas

Todas as actividades desportivas tem um aspecto importante,o clima caso sejam praticadas ao ar livre. Devido a este aspecto foram criadas duas classes abstractas,subclasses de *Activity*,para essa distinção.

- Outdoor,contém a variável: *String weather*
- Indoor

Todas as actividades desportivas são subclasses de *Indoor* ou *Outdoor* como exemplificado na figura 1.

### 1.1.3 Comparadores

Para organizar as actividades criaram-se dois tipo de comparadores:

- CompareActivity- Compara a actividade pela data da realização da mesma.
- CompareActivityByTime- Compara a actividade pelo tempo gasto na realização desta.

## 1.2 Utilizadores

### 1.3 Classe Abstracta Person

Aqui usámos mais uma classe abstracta para definir todas as pessoas que usam a aplicação (Administradores ou utilizadores normais). As variáveis comuns aos todos os tipos de pessoas são *email*, *password*, *name*, *gender*, e *dateOfBirth*. Para além de construtores, *getters* e *setters* esta classe não tem métodos auxiliares.

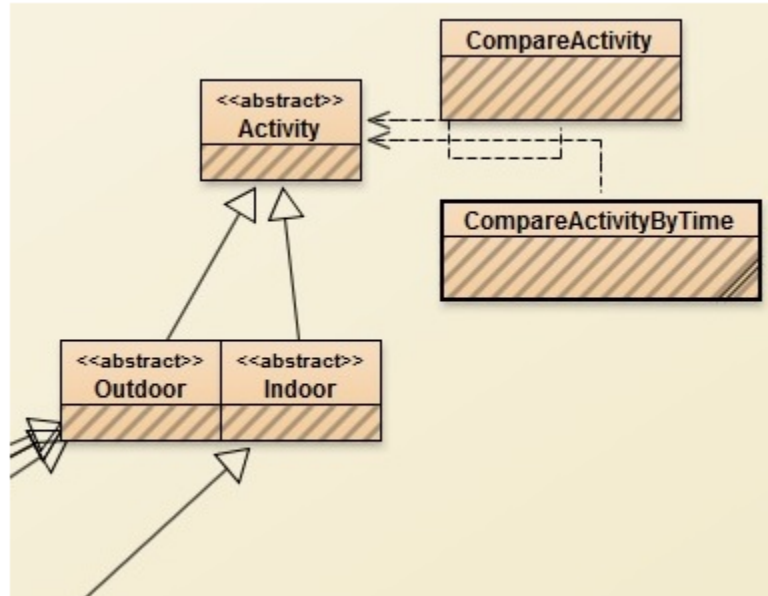


Figura 2: Comparador Activity

## 1.4 Eventos

### 1.5 Classe abstracta Event

Classe com o conceito mais abstracto de Evento, contém as variáveis *name*, *tipoActivity*, *location*, *maxParticipants*, *participants*, *deadline*, *date*, *duration*, *participantsList*, *ranking*, *desistentes* e *simula*, respetivos *getters* e *setters* e os vários contrutores. Ainda tem métodos auxiliares para adicionar um *User*, *ranking*,

### 1.6

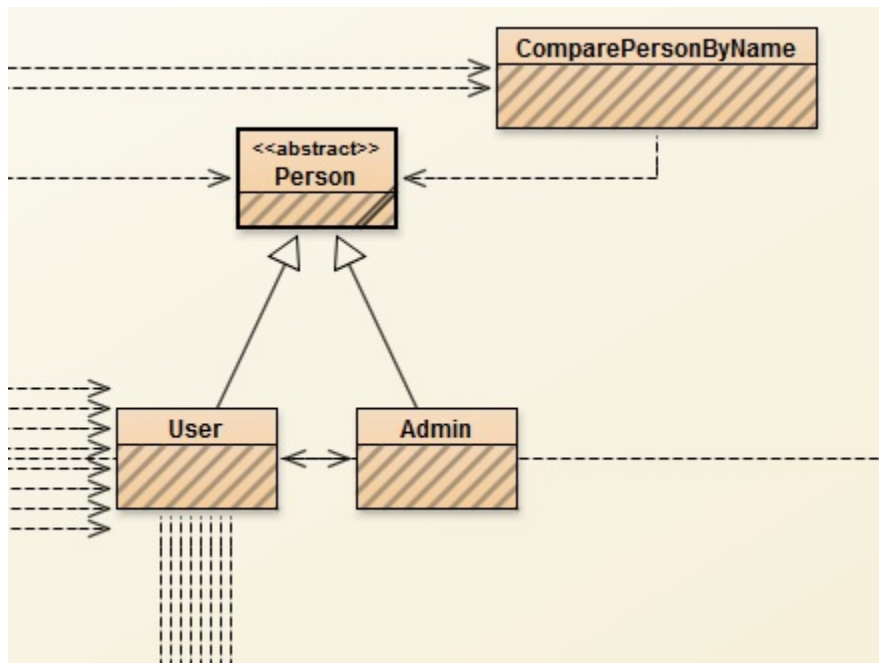


Figura 3: Estrutura das classes User e Admin