**Bases de Dados 2018**

**Home Gym**

A mobile application for

workout challenges

Angelo Miguel Tenreiro Teixeira , up201606516

Henrique Melo Lima , up201606525

Rui Pedro Moutinho Moreira Alves , up201606746

07 – Março - 2018

# **Index**

# What is Home gym? . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 3

# Project’s Specification . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 4

# Conceptual Model . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 6

# Relational Model . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 8

# Functional Dependencies and Normal Form Analysis . . . . . . . . . . . . . . . . . . . 10

# Restrictions . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 12

# **What is home gym?**

**Home Gym** is a mobile application for challenging other people to do workout with you in a fun and interactive way. You get rewards by completing challenges and get feedback on how well you did, to help you improve for next time!

Our idea with this project is to create a **Data Base** to manage all the user and app information, keeping track of all the user’s statistics, all on-going challenges, types of exercises, user reviews, and others (this subject will be specified in the following chapter).

# **Project’s Specification**

Any person using the application is a **User**. The user connects to the app with its facebook account, being characterized by his facebook ID, personal name and by his score, calculated from the scores of all the challenges the user has participated in.

A **User** can participate in more than one **Challenge** at a time. Each Challenge is composed by an ID, by a start and ending date and information about if the challenge is public (1) or not. In each challenge there can be 2 or more participants. When the Challenge is created, it is also specified in which **Week days** the **Exercise Plan** associated to the Challenge should be executed (e.g. , the Challenge consists in completing an exercise plan every Monday, Wednesday and Friday from 15-03-2018 to 20-04-2018).

Associated to each challenge the user is participating in, there are stored its **Participation Details**, composed by the user’s score in that challenge, the rating the user chooses for that Challenge (after challenge is completed / after the user gives up) and the various **Executions** of the exercise plan related to that Challenge. Each Execution consists in the duration and the date the user completed that execution (so that the user can keep track of improvements).

An **Exercise** **Plan** is composed by it’s unique ID, a recommended cooldown (number of days that are recommended between Exercise Plan executions) and by its difficulty (calculated by the average difficulty of the **Exercises** that compose the Exercise Plan). An **Exercise Plan** can be a **Default** **Plan** (created by the app developers) or it can be a **Custom Plan** (created by an User). A Custom Plan can be public (1) or not and saves the date of the last time it was used.

An **Exercise** is composed by its unique ID, a name, a description, an image (that visually describes the exercise), a link to a video that explains how the exercise should be executed, its difficulty (rated from 1 to 5) and its **Exercise Type**, which can be Endurance, Strength, Flexibility or Balance

An exercise can be associated to many different exercise plans, and an Exercise Plan is composed by one or more Exercises. For each Exercise in an Exercise Plan, there are **Exercise Parameters**, that is, the number of repetitions and number of sets for that Exercise execution (e.g. , 3 sets of 10 repetitions of push-ups , 3\*10 = 30 push-ups).

An **Exercise** can, simillarly to an Exercis­e Plan, be a **Default Exercise** (created by the developers) or it can be a **Custom Exercise** (created by an User). A Custom Exercise can be public (1) or not.

(1) – public : visible to other Users

# **Conceptual Model**

The conceptual model displayed above can also be found in a file attached to the submission (**Conceptual Model.png**) for an easier view.

# **RELATIONAL MODEL**

**User** (facebookID, nickname, score)

facebookID → nickname, score

**Challenge** (id, startTime, endTime, isPublic, exercisePlan→ExercisePlan)

id → startTime, endTime, isPublic, exercisePlan

**ParticipationDetails** (user→User, challenge→Challenge, score, finalPlanRating)

user, challenge → score, finalPlanRating

**Execution** (id, date, duration, user→User, challenge→Challenge)

id → date, duration, user, challenge

**WeekDay** (id, dayName)

id → dayName

**ExercisePlan** (id, recomendedCooldown, difficulty)

id → recomendedCooldown, difficulty

**CustomPlan** (id→ExcercisePlan, lastTimeUsed, isPublic, creator→User)

id → lastTimeUsed, isPublic, creator

**DefaultPlan** (id→ExercisePlan)

**Exercise** (id, name, videoLink, description, imageURL, difficulty, type→ExerciseType)

id → videoLink, description, imageURL, difficulty, type

**CustomExercise** (id→Exercise, isPublic, creator→User)

id → isPublic, creator

**DefaultExercise** (id→Exercise)

**ExerciseParameters** (exercisePlan→ExercisePlan, exercise→Exercise, numRepetitions, numSets)

exercisePlan, exercise → numRepetitions, numSets

**ExerciseType** (id, name)

**ChallengeDay** (challenge→Challenge, weekDay→WeekDay)

# **FUNCTIONAL DEPENDENCIES AND NORMAL FORM ANALYSIS**

In each of the relatitions described in the previous chapter, the left side of the functional dependencies is a key for that relation, that is, the closure of the attributes in the left side is **all** the attributes in that relation, as shown in the following paragraphs:

**User:**

{ facebookID }+ = { facebookID, nickname, score }

**Challenge:**

{ id }+ = { id, startTime, endTime, isPublic, exercisePlan }

**ParticipationDetails:**

{ user, challenge }+ = { user, challenge, score, finalPlanRating }

**Execution:**

{ id }+ = { id, date, duration, user, challenge }

**WeekDay:**

{ id }+ = { id, dayNumber }

**ExercisePlan:**

{ id }+ = { id, recomendedCooldown, difficulty }

**CustomPlan:**

{ id }+ = { id, lastTimeUsed, isPublic, creator }

**DefaultPlan:**

{ id }+ = { id }

**Exercise:**

{ id }+ = { id, name, videoLink, description, image, difficulty, type }

**CustomExercise:**

{ id }+ = { id, isPublic, creator }

**DefaultExercise:**

{ id }+ = { id }

**ExerciseParameters:**

{ exercisePlan, exercise }+ = { execisePlan, exercise, numRepetitions,

numSets }

**ExerciseType:**

{ id }+ = { id, name }

**ChallengeDay:**

{ challenge, weekDay }+ = { challenge, weekDay }

Therefore, since in each of the relatitions the left side of the functional dependencies is a key for that relation, the relational model is in the **Boyce-Codd Normal Form** (BCNF). Since the **3rd Normal Form** (3NF) is a super set of the BCNF, that is, every relation in the BCNF is also in the 3NF, the relational model is also in the **3rd Normal Form**.

# **RESTRICTIONS**

**User:**

*- score* is the sum of all the *ParticipationDetails.score* the user is in.

- default value: 0

- *nickname* must have [6..48] characters

**Challenge:**

*- startTime*  and *endTime* are both dates. *endTime*  must be greater than *startTime.*

*- IsPublic –*  boolean (true or false)

- *IsPublic* default value is **true**

**ParticipationDetails:**

*- finalPlanRating*  default value is **Null**.

- finalPlanRating ∈ [1,10]

*- score* is calculated by multiplying the difficulty of the Exercise Plan associated to the Challenge, the number of exercises associated to that plan and the number executions between the user and the challenge.

- *score* “Simplified Expression”:

score = planDifficulty \* numExercisesInPlan \* numExecutions

**WeekDay:**

*- dayNumber* can hold the following values:-Monday  
 - Tuesday  
 - Wednesday  
 - Thursday  
 - Friday  
 - Saturday

- Sunday

**ExercisePlan:**

*- difficulty* is the average of all the *Exercise.difficulty* associated to this plan.

**CustomPlan:**

*- isPublic –*  boolean (true or false)

- *isPublic* default value is **true**

**Exercise:**

*- difficulty*  can hold values in the range [1 , 5]

- difficulty default value: 3

**CustomExercise:**

*- isPublic –*  boolean (true or false).

- *isPublic* default value is **true**