First Delivery

Group Nº 9 | **Class -** 4MIEEC_T2_RC

Name: Daniela Filipa Oliveira Faria || up201604500@fe.up.pt

Nome: Sónia Inês Moreira da Silva || up201606449@fe.up.pt

Topic: Gym Management Database System.

Description of the topic:

The associated member has his/hers own, id, associated number, username and password to enter in the site or the building, sex, weight, height, muscle mass, and fat mass indices, birthday, age, health insurance number. Also is stored the information about intolerances and allergies. There are clients which can be only for one class, so they only need associated number.

Each associated member can have her training plan which have the number of workouts per week, start date, end date and the exercises that are planned.

Each exercise has the weight, number of repetitions per serie, number of series, break between series and the equipment needed.

Each associated member can have a training goal, with the information of intended weight - fat mass, muscle_mass, number of calories per train intended to lose, distance intended to achieve, number of workouts, number of classes , calories to eat.

When the associates enter the gym they receives a free nutrition appointment and physical assessment. They can request another when they want. Which appointment need an ID, a date, a hour, the weight and height of the associate. For the physical assessment it's analysed the eight, weight, body fat, muscle mass, bone weight, body mass, basal metabolism, metabolic age, water percentage, and visceral fat level. For the nutrition one, it's in a specific room and also measured the diabetes and cholesterol levels. There is build the nutrition plan which has an unique ID, a start date, end date and the schedule for each food with information of the day and also the hour. For the food each recommendation has a name, a quantity, a type and a preparation method.

An employee has a name, a username and password to enter in the site or the building, id, sex, birthday, age, health insurance number, start date, type of contract, degree,

experience, languages which he/her is able to speak, and main function in the gym (P.T, cleaning, receptionist, teacher, …). An employee can also be an associated member.

One registe is the registration of each entrance in the gym for employees and associated members. It is registered the date, the hour of the entrance, the hour when the person goes out and a unique ID for that regist.

Besides normal gym environment, there are also classes and they have to have at least one student and one employee who is going to delivery the class, an unique ID, name, type, day, duration, hour which the class starts and ends, number of vacancies and the equipment needed. Is also needed the classroom which the class is going to happen.

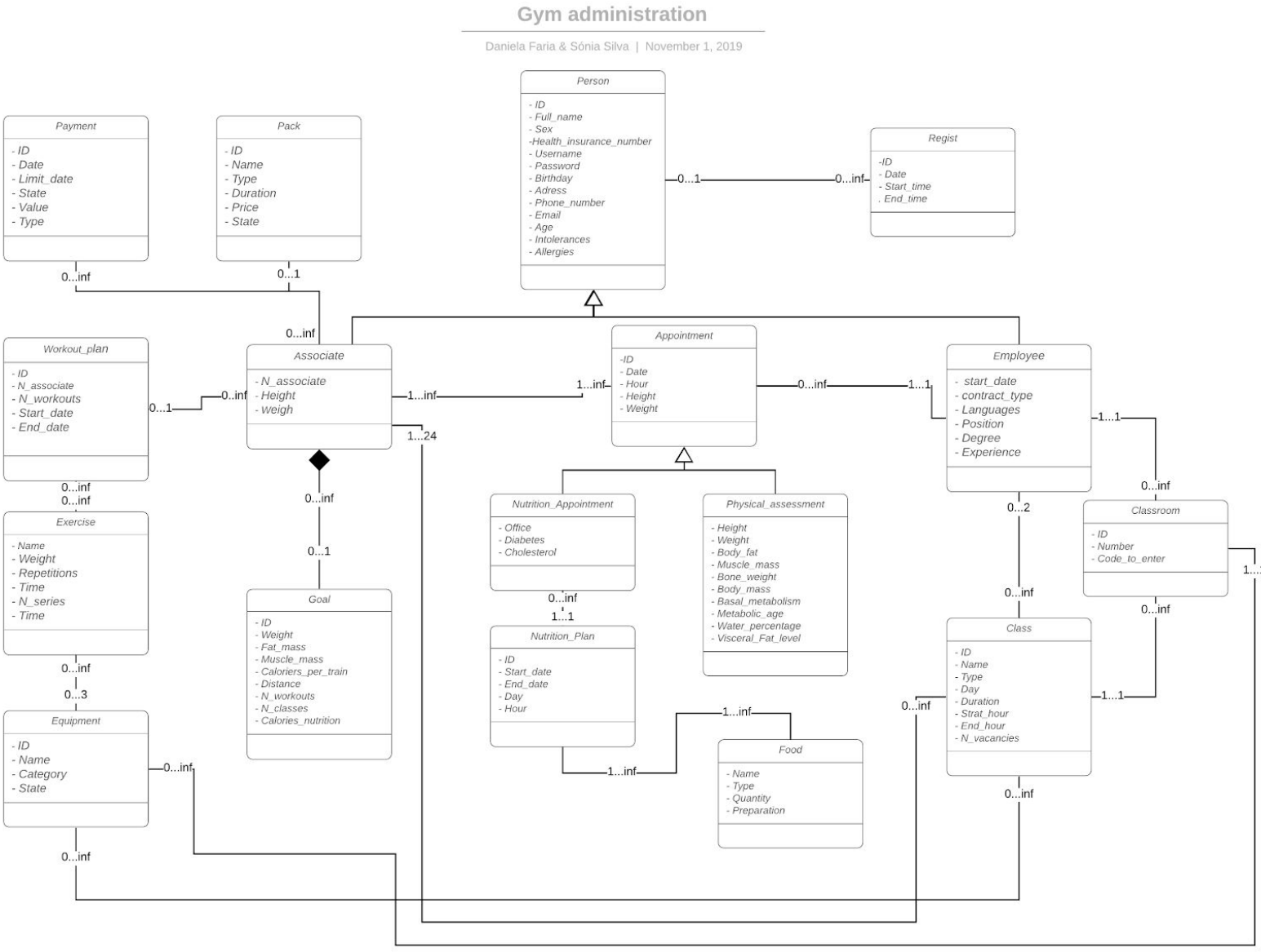
A classroom has an ID, a number, an employee that is responsible for the cleaning, and a pin that allows to open the door of the classroom. Also has the information of the equipment in that classroom.

For the equipment each of one has an unique ID, a name, a category and a state.

For payments, there are plans, each associated member can have a pack, with an unique ID, a name, a type, a duration period, a prace and a state. For the payments, they need anique ID as well, a date, a limit date, a state, a value and a type.

UML Diagram

Link to a better image:
https://drive.google.com/open?id=1U8eZtD8cesRmBUwIYTdqAPsW8BE7xLQM



Relational Model

Person(ID, Full_name, health_insurance_number, username, password, birthday, adress, phone_number, email, age, intolerances, allergies,)

NOT NULL(Full_name)

NOT NULL(health_insurance_number)

Associate(N_associate, height, weight, ID->Person)

Employees (ID->Person, start_date, contract_type, languages, position, degree, experience)

NOT NULL(start_date)

NOT NULL(position)

Goal(ID->Associate, weight, fat_mass, muscle_mass, calories_per_train, distance, n_workouts, n_classes, calories_nutrion)

Pack(ID, name, type, duration, price, state, pack_of_associate->Associate)

NOT NULL(Price)

Payment(ID->Associate, date, limit_date, state, value, type)

NOT NULL(Value)

Appointment(ID, date, hour, height, weight, given_to->Associate, given_by->employee)

NOT NULL(date)

NOT NULL(hour)

Nutrition_Appointment(ID->Appointment, office, diabetes_index, cholesterol_index)

NOT NULL(office)

Nutrition_Plan(ID, start_date, end_date, day, hour, given_in->Nutrition_Appointment)

Food(ID, Name, Type, Quantity, Preparation, food->Nutrition_Plan)

NOT NULL(Name)

NOT NULL(Type)

Physical_assessment(ID->Appointment, height, weight, body_fat, bone_weight, body_mass, basal_metabolism, metabolic_age, water_percentage, visceral_fat_level)

NOT NULL(height)

NOT NULL(weight)

Workout_plan(ID, n_workouts, start_date, end_date, from->associate)

Exercise(Name, weight, repetitions, time, n_series, time_break, with->Equipement)

Equipment(ID, Name, category, state, used->Exercise)

Class(ID, name, type, day, duration, start_hour, end_hour, n_attendees, given_by->Employee, given_to->Associate, given_with->equipment)

NOT NULL(Name)

NOT NULL(Duration)

CHECK(N_attendees>0)

CHECK(given_by != NULL)

Classroom(ID, number, code_to_enter, with_this->Equipment, cleen_by->Employee)

NOT NULL(number)

Regist(ID, date, start_time, end_time, who_entered->person)

NOT NULL(date)

NOT NULL(start_time)

NOT NULL(end_time)