

**2ºano – MIEIC – 2016/2017**

**Gestão de um Ginásio**

**Bases de Dados**

**Turma 4 - Grupo 07**



**João Miguel Matos Monteiro - up201506130**

**Maria Eduarda Santos Cunha – up201506524**

**Mário Rui Macedo Flores dos Santos - up201503406**

Índice

[1. Contexto 3](#_Toc483496730)

[1.1. Restrições e Multiplicidades 4](#_Toc483496731)

[2. Diagrama UML 6](#_Toc483496732)

[3. Modelo Relacional 7](#_Toc483496733)

[4. Análise de Dependências Funcionais e Formas Normais 8](#_Toc483496734)

[4.1. Análise de Formas Normais 9](#_Toc483496735)

[5. Interrogações 10](#_Toc483496736)

# Contexto

A nossa base de dados consiste na gestão de toda a informação inerente a um ***Ginásio***.

Tal como o próprio tema indica, temos a classe ***Ginásio***, que possui a informação **nome**, **morada** e **número de telefone fixo**.

Relativamente a instalações, o ***Ginásio*** está associado às classes ***Balneário*** e ***Sala***.

O ***Balneário*** possui o atributo **número**, **género**, **lotação máxima** e **atual**. Sendo que, o atributo **número** refere-se ao identificador da sala no edifício, o atributo **género** permite saber se é um ***Balneário*** destinado a indivíduos do sexo masculino ou feminino e os atributos **lotação máxima** e **lotação atual** dão a conhecer a capacidade máxima do ***Balneário*** e a quantidade de pessoas presentemente lá, respetivamente. Consequentemente, estes dois últimos atributos permitem também saber a quantidade de pessoas que ainda podem ocupar um ***Balneário***, através da sua diferença.

A ***Sala***, por sua vez, é detentora dos atributos **número**, **dimensão**, **lotação máxima** e **atual**. Assim, o **número** constitui um identificador próprio que distingue as várias salas existentes no ***Ginásio***, a **dimensão** representa o espaço da sala (que pode ser mais ou menos adequado à prática de uma certa modalidade) e, por fim, a **lotação máxima** e **atual** funcionam de forma semelhante aos mesmos atributos na classe ***Balneário***.

Quanto à classe ***Pessoa***, que reúne todas as informações comuns tanto a funcionários como a clientes da cadeia, temos o **id**, **nome**, **idade**, **Bilhete de Identidade**, **número de contribuinte**, **morada** e **número de telemóvel**. Da classe ***Pessoa*** derivam as classes ***Membro*** e ***Funcionário***, que separam os indivíduos que frequentam um ***Ginásio*** ou que trabalham nele.

A classe ***Funcionário*** divide-se nas classes ***Professor***, ***Treinador***, ***Rececionista***, ***Contínuo*** e ***Gerente***, que herdam os seus atributos: derivados de ***Pessoa***, **salário** e **habilitações académicas**.

A classe ***Membro*** tem um **identificador** próprio atribuído no ***Ginásio***, necessário para se deslocar nas instalações, derivado, à semelhança do ***Funcionário***, da classe ***Pessoa***. O ***Membro*** pode ainda estar ou não **ativo**, dado que qualquer indivíduo que alguma vez tenha sido ***Membro***, ainda que já não o seja, deve permanecer registado. Isto não só por uma questão de consulta de registos antigos, mas também, por exemplo, para o caso de querer renovar o ***Contrato***.

O ***Contrato*** dispõe dos atributos **pagamento,** que corresponde ao montante pago pelo ***Membro***, e **regime**, relativo ao “tipo” de contrato.

Para estipular o **dia da semana**, **hora de início** e **hora de fim**, existe a classe ***Horário***. Possui também um **id**, que permite saber exatamente a que ***Horário*** cada ***Contrato*** ou ***Modalidade*** se está a referir.

A ***Modalidade*** é, essencialmente, uma aula dada por um ou vários professores e distingue-se pelo seu **nome**. Pode ser dada em várias salas.

Por fim, uma ***Sala*** encontra-se associada ao ***Equipamento*** nela contido. Esse ***Equipamento*** distingue-se pelo seu **id**, já que cada máquina é seriada com um número que a distingue de todas as outras. Possui ainda os atributos **nome** e **funcionalidade** e pode ou não estar disponível, segundo dita o atributo **disponível**.

## 1.1. Restrições e Multiplicidades

Algumas restrições não evidentes através do diagrama UML, a seguir representado, são, por exemplo:

* A **idade**:
  + - Um ***Funcionário*** só pode ser empregue no ***Ginásio*** caso seja maior de **idade**;
    - Um ***Membro***, cuja **idade** seja inferior a maioridade, só se pode registar com autorização parental.
* O **género** do ***Balneário***:
  + - Um ***Balneário*** só pode ser do **género** masculino ou feminino.
* As horas:
  + - A **hora de início** tem de ser inferior à **hora de fim**.
* Os **nº de telemóvel/telefone**:
  + - O **nº de telemóvel** tem de ter 9 dígitos, sendo que o primeiro será sempre 9 e o segundo respeitante à rede;
    - O **nº de telefone fixo** também deve ser constituído por 9 dígitos e os primeiros 3 são respeitantes à zona.
* O **nº de contribuinte**:
  + - Tem de ter 9 dígitos.
* O **regime**:
  + - Pode ser anual ou mensal.

Um ***Membro*** só possui um tipo de ***Contrato***. E um ***Contrato*** só pode estar associado a um ***Horário***.

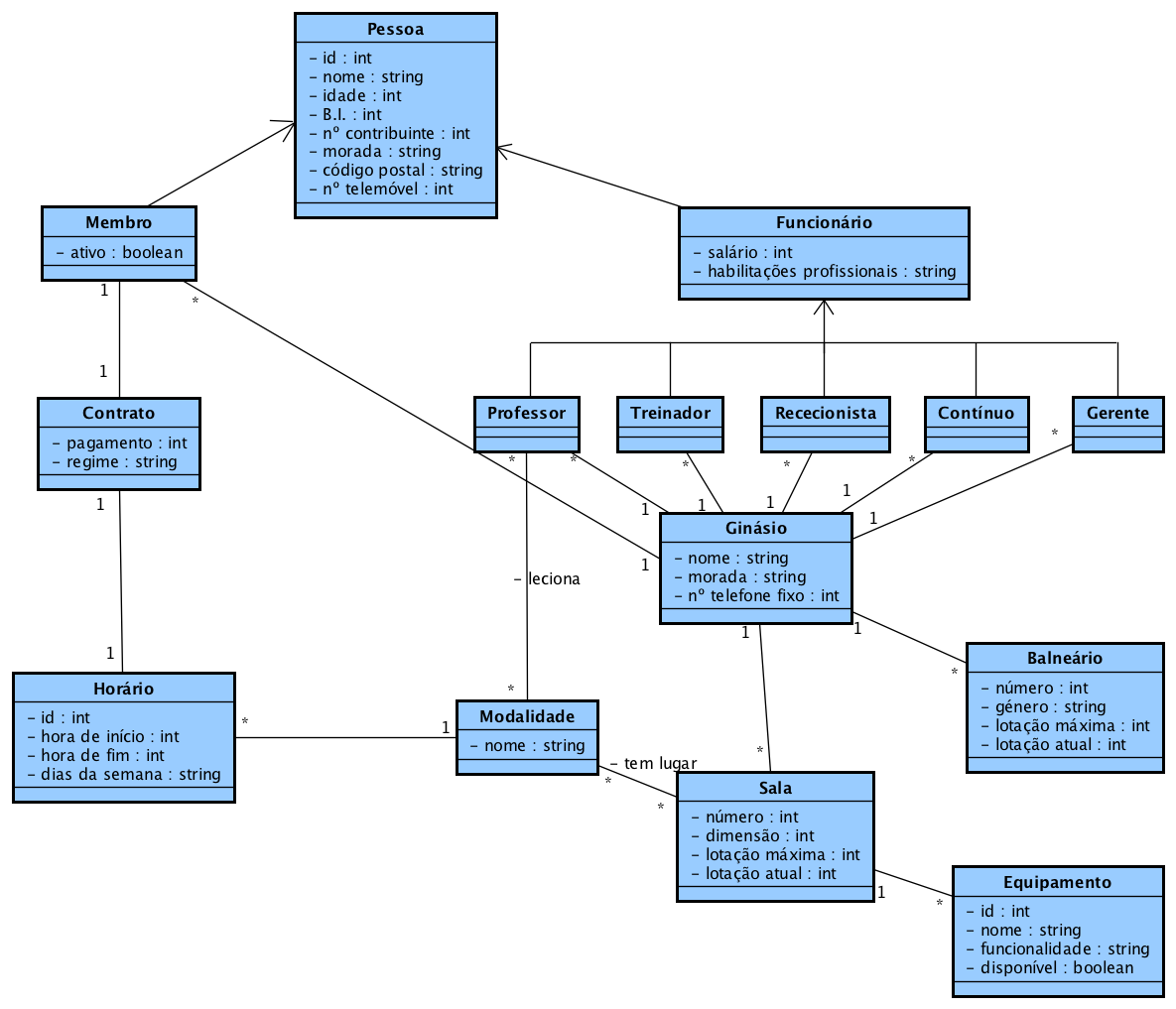
Um ***Ginásio*** pode ter vários professores, treinadores, rececionistas, contínuos e gerentes.

Um ***Ginásio*** vem associado a vários balneários e salas.

Uma ***Sala*** pode albergar vários tipos de ***Equipamento***. Ainda, cada ***Sala*** encontra-se associada a diversas modalidades, lecionadas por vários professores.

Cada ***Modalidade*** possui vários horários.

# Diagrama UML



# Modelo Relacional

***Pessoa*** (**id**, **nome**, **idade**, **B.I**., **nº contribuinte**, **morada**, **nº telemóvel**)

***Funcionário*** (idFuncionário->Pessoa, **salário**, **habilitações profissionais**)

***Professor*** (idProfessor->Funcionário)

***Treinador*** (idTreinador->Funcionário)

***Rececionista*** (idRececionista->Funcionário)

***Contínuo*** (idContínuo->Funcionário)

***Gerente*** (idGerente->Funcionário)

***Membro*** (idMembro->Pessoa, **ativo**)

***Contrato*** (idMembro->Membro, **pagamento**, **regime**)

***Horário*** (**id**, **hora de início**, **hora de fim**, **dias da semana**)

***Ginásio*** (**nome**, **morada**, **nº telefone fixo**)

***Balneário*** (**número**, **género**, **lotação máxima**, **lotação atual**)

***Sala*** (**número**, **dimensão**, **lotação máxima**, **lotação atual**)

***Equipamento*** (**id**, **nome**, **funcionalidade**, **disponível**)

***Modalidade*** (**nome**)

***Leciona*** (nomeModalidade->Modalidade, idProfessor->Professor)

***Tem Lugar*** (nomeModalidade->Modalidade, númeroSala->Sala)

# Análise de Dependências Funcionais e Formas Normais

***Pessoa*** (**id**, **nome**, **idade**, **B.I**., **nº contribuinte**, **morada**, **nº telemóvel**)

**id**->**nome**, **idade**, **B.I.**, **nºcontribuinte**, **morada**, **nºtelemóvel**

**B.I.**-> **id**, **nome**, **idade**, **nºcontribuinte**, **morada**, **nºtelemóvel**

**nº contribuinte**-> **id**, **nome**, **idade**, **B.I.**, **morada**, **nºtelemóvel**

***Funcionário*** (idFuncionário->Pessoa, **salário**, **habilitações profissionais**)

idFuncionário->**salário**, **habilitações** **profissionais**

***Professor*** (idProfessor->Funcionário)

***Treinador*** (idTreinador->Funcionário)

***Rececionista*** (idRececionista->Funcionário)

***Contínuo*** (idContínuo->Funcionário)

***Gerente*** (idGerente->Funcionário)

***Membro*** (idMembro->Pessoa, **ativo**)

idMembro-> **ativo**

***Contrato*** (idMembro->Membro, **pagamento**, **regime**)

idMembro->**pagamento, regime**

***Horário*** (**id**, **hora de início**, **hora de fim**, **dias da semana**)

**id**->**hora de início**, **hora** **de** **fim**, **dias** **da** **semana**

***Ginásio*** (**nome**, **morada**, **nº telefone fixo**)

**nome**->**morada**, **nºtelefone fixo**

**nºtelefone** **fixo**->**nome**, **morada**

**morada**->**nome**, **nºtelefone fixo**

***Balneário*** (**número**, **género**, **lotação máxima**, **lotação atual**)

**número**->**género**, **lotação** **máxima**, **lotação atual**

***Sala*** (**número**, **dimensão**, **lotação máxima**, **lotação atual**)

**número**->**dimensão**, **lotação** **máxima**, **lotação atual**

***Equipamento*** (**id**, **nome**, **funcionalidade**, **disponível**)

**id**->**nome**, **funcionalidade**, **disponível**

**nome**->**funcionalidade**

***Modalidade*** (**nome**)

## 4.1. Análise de Formas Normais

Todos os Modelos Relacionais estão na **Forma Normal de Boyce-Codd**, pois, através de um único atributo chave, é possível aceder a todos os outros atributos da classe.

Todo o modelo que esteja na **Forma Normal de Boyce-Codd** encontra-se obrigatoriamente na**3ª Forma Normal**, de modo que todos estes Modelos Relacionais estão na **3ª Forma Normal**.

# Interrogações

1. **Quantos membros existem?**

**SELECT COUNT**(\*) **FROM** Pessoa

**WHERE** id **IN** (SELECT id FROM

(**SELECT** \* **FROM** Membro));

1. **Quais membros com idade inferior a 20?(ordenados por idade e ordem alfabetica)**

**SELECT** id **AS** [ID], nome **AS** [Nome], idade **AS** [Idade] **FROM** Pessoa

**WHERE** idade <20 **AND** id IN (**SELECT** id **FROM**

(**SELECT** idMembro **FROM** Membro))

**ORDER BY**

idade **DESC**,

nome **ASC**;

1. **Quais são os 10 contratos anuais com maior pagamento?**

**SELECT TOP** 10 **WITH TIES** pagamento

**FROM** Contrato

**WHER**E

Contrato.regime=[anual]

**ORDER BY** pagamento **DESC;**

1. **Qual é a idade média dos treinadores que teem salario superior a 1000?**

**SELECT AVG**(idade) **FROM** Pessoa

**WHERE** id **IN** (**SELECT** id **FROM**

(**SELECT** \* **FROM** Treinador)) **AND** Funcionario.salario > 500;

1. **Quais são todos os membros e treinadores?**

**SELECT** nome **FROM** Pessoa,Membro

**WHERE** Pessoa.id = Membro.idMembro

**UNION**

**SELECT** nome **FROM** Pessoa,Treinador

**WHERE** Pessoa.idPessoa = Treinador.idTreinador;

1. **Qual é o ID do gerente Bruno Florenço?**

**CREATE TEMP TABLE IF NOT EXISTS** Gerente **AS**

**SELECT** Pessoa.id, Pessoa.nome

**FROM** Pessoa, Gerente

**WHERE** Gerente.id = Pessoa.idPessoa

**ORDER BY** nome **ASC**;

**SELECT DISTINCT** Pessoa.id from Pessoa, Gerente

**WHERE** Pessoa.nome **IN** (**SELECT** nome **FROM** Gerente)

**AND** Pessoa.nome = 'Bruno Florencio';

1. **Quais são as salas quem atingiram lotaçao maxima? (ordenadas por numero)**

**SELECT** numero **FROM** Sala

**WHERE** Sala.lotacaomax = Sala.lotacaoatual

**ORDER BY** numero **ASC**;

1. **Qual é o equipamento disponivel, cuja funcionionalidade contenha “esculpir”?**

**SELECT** id **AS** [ID], Equipamento.nome **AS**[Equipamento], Ginasio.nome **AS** [Ginasio] **FROM** Equipamento, Ginasio

**WHERE** Equipamento.funcionalidade **LIKE** '%esculpir%' **AND** Equipamento.disponibilidade = 1

**ORDER BY** id **DESC**;

1. **Quais são os funcionários cujo nome começa por M ou N? (ordenados por ordem alfabetica)**

**SELECT** id **AS** [ID], nome **AS** [Nome] **FROM** Funcionario

**WHERE** Funcionario.nome **LIKE** 'M%' **OR** Funcionario.nome **LIKE** 'D%'

**ORDER BY** Funcionario.nome **ASC**;

1. **Quais são os horários que não são à sexta-feira nem à segunda-feira? (ordenados por ID)**

**SELECT** id **AS** [ID], dia **AS** [Dia**] FROM** Horario

**WHERE** dia <> '%Sexta%' **AND** dia <> '%Segunda%'

**ORDER BY** id **ASC**;

# Triggers

Para evitar a circulação de informação inválida na base de dados criamos 3 triggers.

1. **Trigger para quando a funcionalidade de um equipamento for NULL, substituir pelo default “aumento de resistencia”**

**CREATE TRIGGER** NullFunc

**AFTER INSERT ON** Equipamento

**FOR EACH ROW**

**WHEN** (**NEW**.funcionalidade **ISNULL**)

**BEGIN**

**UPDATE** Equipamento **SET** nome = 'aumento de resistencia';

**END**;

1. **Trigger para apagar um contrato após este ser criado se** **ao ser inserido tiver um valor de pagamento inferior a 50**

**CREATE TRIGGER** AnualContract

**AFTER INSERT ON** Contrato

**FOR EACH ROW**

**WHEN (NEW.**pagamento < 50**)**

**BEGIN**

**DELETE FROM** Contrato **WHERE (SELECT** idMembro **FROM** Contrato

**WHERE** idMembro**=NEW.**idMembro**)**;

**END**;

1. **Trigger para fazer ABORT após a criação de um funcionário cujo salário seja inferior ao salário mínimo**

**CREATE TRIGGER** MinSalary

**BEFORE INSERT ON** Funcionario

**FOR EACH ROW**

**BEGIN**

**SELECT CASE**

**WHEN NEW.**salario <= 557

**THEN RAISE(ABORT,** 'Abaixo do salario minimo!!!'**)**;

**END**;