

1º Trabalho Prático

## Sistemas de informação II

Licenciatura Engenharia Informática e de Computadores

**Docente:** Afonso Remédios

**Grupo 7:**

Pedro Miranda - 43584

Ricardo Rodrigues – 43594

Índice

[Introdução 3](#_Toc89640557)

[Modelo Entidade-Associação 4](#_Toc89640558)

[Modelo Relacional 5](#_Toc89640559)

[Avaliação Experimental 6](#_Toc89640560)

[Exercício A 6](#_Toc89640561)

[Exercício B 6](#_Toc89640562)

[Exercício C 6](#_Toc89640563)

[Conclusão 7](#_Toc89640564)

[WebGrafia 8](#_Toc89640565)

# Introdução

No enunciado deste trabalho é proposta a resolução de um problema através da manipulação de uma base de dados. Para tal, será necessária a concretização dos modelos relacional e entidade-associação, bem como a criação e inserção de dados nas respetivas tabelas e a utilização do mecanismo transacional de bases de dados.

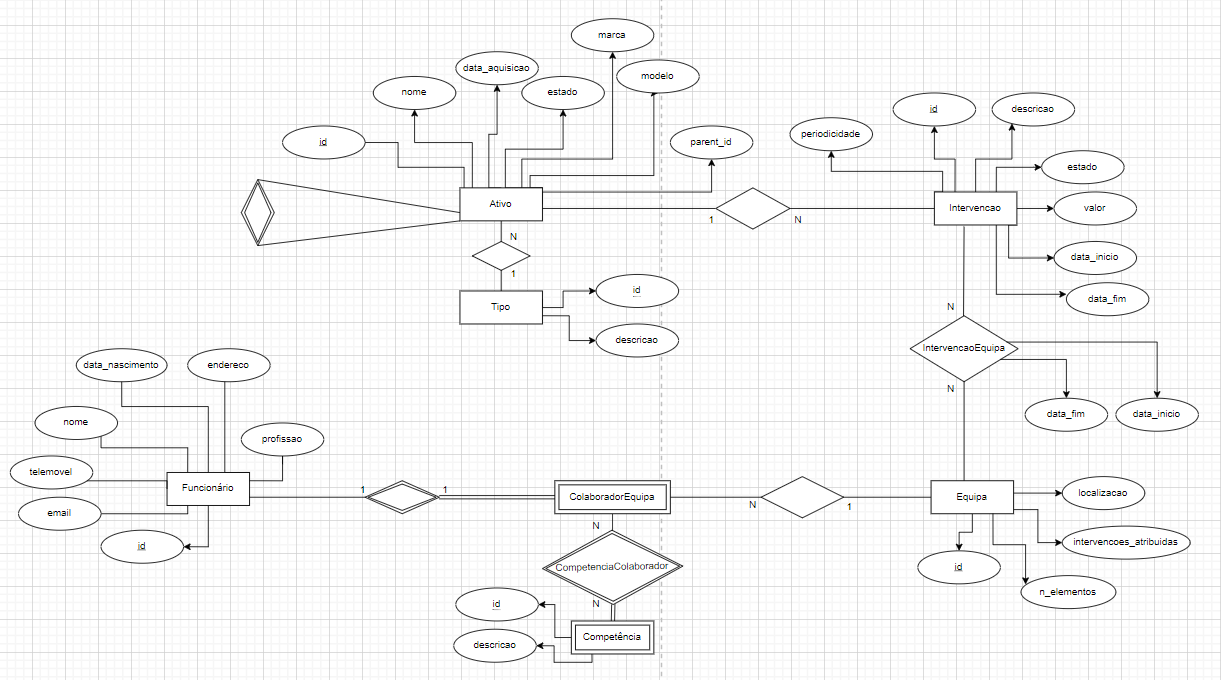
O mecanismo transacional tem um controlo que permite executar várias interações com a base de dados ao mesmo tempo, utilizando diferentes transações. Uma transação é um conjunto de instruções que irão ser usadas numa base de dados. Para que estas sejam compatíveis, de forma a poderem ser executadas em simultâneo, terão de ser atómicas. Para não corromperem a base de dados, estas também têm de ser consistentes. Isto é, isoladas uma das outras de forma a garantir resultados perseverantes, tanto no caso de um *commit* como de um *rollback*.

As transações têm como objetivo preservar a integridade e a consistência de uma base de dados, sendo estas mecanismos de recuperação em caso de falhas do sistema. Os escalonamentos permitem-nos utilizar várias transações e executá-las na mesma ordem. Um conflito num escalonamento acontece quando as operações pertencem a transações diferentes. Estes conflitos geram anomalias, ocorrendo quando uma transação consegue ler dados resultantes de escritas que ainda não sofreram um *commit* por outra transação. A leitura não repetitiva (*Non Repeatable Read)*, isto é, quando uma leitura demonstra resultados diferentes, após ser executada novamente, mais tarde na mesma transação. Por último temos a “leitura fantasma” (Phantom Read), que ocorre quando em duas leituras diferentes de uma transação, há uma alteração nos registos.

De modo a evitar estas anomalias são usados níveis de isolamento. Começando pelo nível de isolamento mais “baixo”, isto é, o *Read Uncommitted,* que permite que uma transação manipule valores que não são sujeitos a *commit* por outras transações. O nível seguinte é o *Read Committed*, que deixa que uma transação leia e manipule dados que já sofreram *commit* por outras transações. De seguida temos o *Repeatable Read*, que garante que a mesma leitura de um dado se repita, originando resultados iguais para diferentes execuções na mesma transação. Para finalizar, o nível mais alto e seguro é o *Serializable* que, apesar de ser semelhante ao anterior, este garante que as linhas selecionadas por uma transação não podem ser alteradas/lidas por outra transação, até que a primeira transação seja concluída.

# Modelo Entidade-Associação

Figura - Modelo Entidade-Associação



# Modelo Relacional

**CE:** Chave estrangeira

**RI:** Restrições de Integridade

**Ativo** (id, parent\_id, nome, data\_aquisicao, marca, modelo, localizacao, estado, id\_tipo)

**CE:** id\_tipo referencia Tipo(id)

**RI1:** todos os campos, com exceção de marca e modelo, são obrigatórios

**RI2:** O estado só pode tomar 2 valores: ‘0’ (desativado) ou ‘1’ (operacional)

**Funcionario** (id, email, telemovel, nome, data\_nascimento, endereco, profissao)

**Tipo** (id, descricao)

**Equipa** (id, localizacao, n\_elementos, intervencoes\_atribuidas)

**RI3:** intervencoes\_atribuidas pode ter, no máximo, o valor 3

**Intervencao** (id, descricao, estado, valor, data\_inicio, data\_fim, periodicidade, ativo\_id)

**CE:** ativo\_id referencia Ativo(id)

**RI4:** descrição pode tomar como valores “avaria”, “rutura” ou “inspeção”

**RI5:** estado pode tomar valores como “por atribuir”, “em análise”, “em execução” ou “concluído”

**RI6:** data\_inicio deverá ser superior a Ativo(data\_aquisicao)

**IntervencaoEquipa** (id\_equipa, id\_intervencao, data\_inicio, data\_fim)

**CE**: id\_equipa referencia Equipa(id)

id\_intervencao referencia Intervencao(id)

**Competencia** (id, descricao)

**ColaboradorEquipa** (id, id\_equipa)

**CE**: id referencia Funcionario(id)

id\_equipa referencia Equipa(id)

**CompetenciaColaborador** (id\_competencia, id\_colaborador, id\_equipa)

**CE**: id\_competencia referencia Competencia(id)

id\_colaborador, id\_equipa referenciam ColaboradorEquipa(id, id\_equipa)

# Avaliação Experimental

A normalização foi feita até à 3ª forma normal, de forma a evitar problemas na gestão da base de dados.

## Exercício A

Para criar o modelo físico foi criado o ficheiro create\_tables.sql, que realiza diversos “CREATE TABLE” de forma a criar as tabelas tendo por base o modelo EA e o relacional.

## Exercício B

Para apagar o modelo físico, apenas se recorreu à criação de um ficheiro drop\_tables.sql, onde são realizados os diversos DROP TABLESdas várias tabelas.

## Exercício C

Neste exercício criámos um ficheiro populate\_tables.sql, onde foram inseridos dados nas tabelas, de forma a servirem como alvo de teste de alguns dos exercícios.

# Conclusão

Considerando o problema exposto, foi-nos dado o conhecimento de como criar, manipular utilizar a gestão transacional numa base de dados. Recorremos a escalonamentos teóricos, de forma a conseguirmos prever que anomalias teríamos de evitar em cada caso. Para evitar a existência das mesmas, foi aprendida a utilidade de recorrer a níveis de isolamento e os seus impactos positivos e negativos a nível computacional, na gestão de uma base de dados.

# WebGrafia

* Relatório do trabalho
* Slides da disciplina do moodle
* [Transact-SQL Reference (Database Engine) - SQL Server | Microsoft Docs](https://docs.microsoft.com/en-us/sql/t-sql/language-reference?view=sql-server-ver15)
* [Microsoft PowerPoint - bd2\_processamento-de-transacoes.ppt [Modo de Compatibilidade] (ufop.br)](http://www.decom.ufop.br/guilherme/BCC441/geral/bd2_processamento-de-transacoes.pdf)
* [bd\_transactions (up.pt)](https://www.dcc.fc.up.pt/~edrdo/aulas/bd/teoricas/bd_transactions.pdf)