

Projecto de Bases de Dados, Parte 1

Bruno Cardoso (72619), Lúdia Freitas (78559) e Rodrigo Bernardo (78942)

Instituto Superior Técnico

16 de Outubro de 2015



Grupo 17

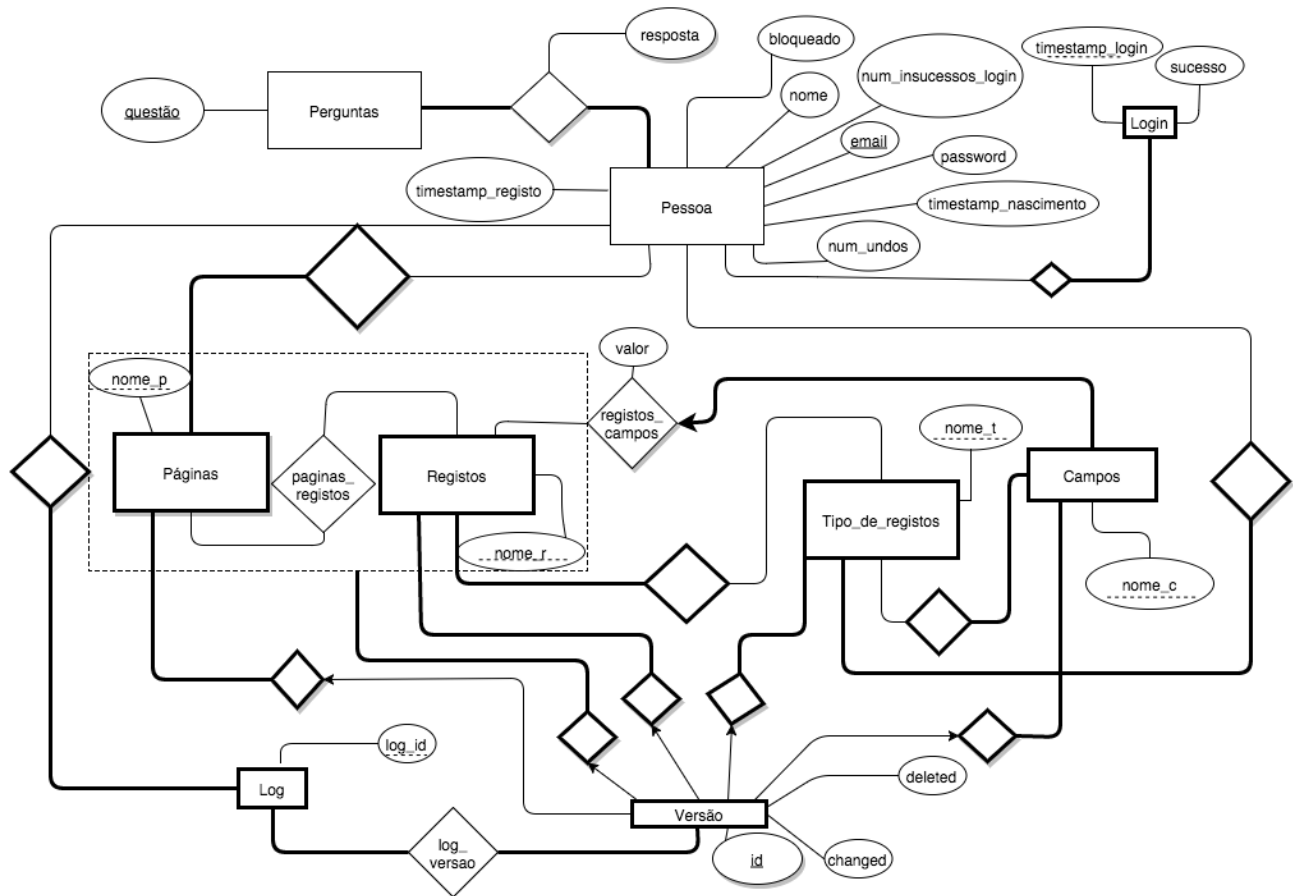
Turno: Quinta-Feira, às 08h00, LAB 14
25 horas de trabalho por aluno.

Conteúdo

1	O Modelo	2
1.1	Modelo Entidade-Associação	2
1.2	Restrições de Integridade do Modelo Entidade-Associação	3
2	Modelo Relacional	3
2.1	O Modelo	3
2.2	Restrições de Integridade do Modelo Relacional	4
3	Álgebra Relacional	4
3.1	Pergunta 1	4
3.2	Pergunta 2	4
3.3	Pergunta 3	4
3.4	Pergunta 4	4
4	Linguagem SQL	4
4.1	Pergunta 1	4
4.2	Pergunta 2	4
4.3	Pergunta 3	4
4.4	Pergunta 4	4

1 Modelo Entidade-Associação

1.1 O Modelo



1.2 Restrições de Integridade do Modelo Entidade-Associação

O Modelo Entidade-Associação não limita todas as ocorrências possíveis e por isso podem dar-se situações fora do domínio do problema.

Como por exemplo, não se consegue assegurar que a ordem das datas estejam coerentes. Isto é, não é possível garantir que o *timestamp_nascimento* seja inferior ao *timestamp_registro*, ou que o *timestamp_login* seja posterior ao *timestamp_registro*. Assim como não é possível assegurar que os números de tentativas de insucesso (*num_insucessos_login*) sejam positivos, ou o número de undo's já realizados pela pessoa (*num_undos*). Outro aspecto que não é explicável através do modelo é o número de perguntas a que a pessoa responde, ou a unicidade da associação entre os objectos mutáveis (*Páginas*, *páginas_registos*, *Registos*, *Tipos_de_registos* e *Campos*). De forma a impedir que tais situações não ocorram, o Modelo Entidade-Associação é completado com as seguintes restrições de integridade:

- RI1** Na entidade *Pessoa*, o atributo *timestamp_nascimento* é uma data anterior ao atributo *timestamp_registro*.
- RI2** Na entidade *Pessoa*, o atributo *bloqueado* é um valor lógico, verdadeiro ou falso.
- RI3** Na entidade *Pessoa*, o atributo *num_undos* é um número inteiro positivo.
- RI4** Na entidade *Pessoa*, o atributo *num_insucessos_login* é um número inteiro positivo menor ou igual a 3.
- RI5** Cada instância de *Pessoa* associa-se, em cada instante, a duas instâncias da entidade *Pergunta*.
- RI6** Na entidade *Login*, o atributo *timestamp_login* é uma data posterior ao atributo *timestamp_registro* de *Pessoa*.
- RI7** Na entidade *Login*, o atributo *sucesso* é um valor lógico, verdadeiro ou falso.
- RI8** Cada instância da entidade *Versão* tem apenas uma associação activa a *Páginas*, *páginas_registos*, *Registos*, *Tipos_de_registos* e *Campos*.
- RI9** Na entidade *Versão*, o atributo *deleted* é um valor lógico, verdadeiro ou falso.
- RI10** Na entidade *Versão*, o atributo *changed* é um valor lógico, verdadeiro ou falso.
- RI11** Uma instância da entidade *Versão* está associada a uma ou duas instâncias da entidade *Log*.
- RI12** Uma instância da entidade *Log* está associada a duas e só duas instâncias da entidade *Versão*.
- RI13** Na entidade *Log*, o atributo *log_id* é um inteiro positivo.

2 Modelo Relacional

2.1 O Modelo

Pergunta(questão, email, resposta)
email: FK(Pessoa)
notnull(resposta)

Pessoa(email, bloqueado, nome, num_insucessos_login, password, timestamp_nascimento, timestamp_registro, num_undos)
notnull(email)
notnull(bloqueado)
notnull(nome)
notnull(num_insucessos_login)
notnull(password)
notnull(timestamp_nascimento)
notnull(timestamp_registro)
notnull(num_undos)

Login(timestamp_login, email, sucesso)
email: FK(Pessoa)
notnull(sucesso)

Páginas(nome_p, email, id)
email: FK(Pessoa)
id: FK(Pessoa)

Páginas_registos(nome_p, email, nome_r, nome_t, id)
nome_p, email: FK(Páginas)
nome_r, nome_t, email: FK(Registos)
id: FK(Versão)

Registos(nome_r, nome_t, email, id)
nome_t, email: FK(tipos_de_registos)
id: FK(Versão)

Tipo_de_registos(nome_t, email, id)
email: FK(Pessoa)
id: FK(Versão)

Campos(nome_c, nome_t, email, id)
nome_t, email: FK(Tipo_de_registos)
id: FK(Versão)

Registo_Campos(nome_r, nome_t, nome_c, email, id)
nome_r, nome_t, email: FK(Registos)
nome_c, nome_t, email: FK(Campos)
id: FK(Versão)

Versão(id, *changed, deleted*)
notnull(changed)
notnull(deleted)

Log(log_id, email)
email: FK(Pessoa)

Log_versão(log_id, email, id)
log_id, email: FK(Log)
id: FK(Versão)

2.2 Restrições de Integridade do Modelo Relacional

Ao passar do Modelo Entidade-Associação para o Modelo Relacional é necessário acrescentar restrições de integridade que antes não eram necessárias. Às restrições de domínio já indicadas, adicionam-se agora as restrições de integridade referencial (sendo que de chave não temos nenhuma), as quais não eram antes necessárias considerar. Assim, as restrições RIn, com $n > 13$, são restrições novas, exclusivas do Modelo Relacional, para cobrir casos que não são possíveis modelar directamente neste modelo.

RI1 Na entidade *Pessoa*, o atributo *timestamp_nascimento* é uma data anterior ao atributo *timestamp_registro*.

RI2 Na entidade *Pessoa*, o atributo *bloqueado* é um valor lógico, verdadeiro ou falso.

RI3 Na entidade *Pessoa*, o atributo *num_undos* é um número inteiro positivo.

RI4 Na entidade *Pessoa*, o atributo *num_insucessos_login* é um número inteiro positivo menor ou igual a 3.

RI5 Cada instância de *Pessoa* associa-se, em cada instante, a duas instâncias da entidade *Pergunta*.

RI6 Na entidade *Login*, o atributo *timestamp_login* é uma data posterior ao atributo *timestamp_registro* de *Pessoa*.

RI7 Na entidade *Login*, o atributo *sucesso* é um valor lógico, verdadeiro ou falso.

RI8 Cada instância da entidade *Versão* tem apenas uma associação activa a *Páginas*, *paginas_registos*, *Registos*, *Tipos_de_registos* e *Campos*.

RI9 Na entidade *Versão*, o atributo *deleted* é um valor lógico, verdadeiro ou falso.

RI10 Na entidade *Versão*, o atributo *changed* é um valor lógico, verdadeiro ou falso.

RI11 Uma instância da entidade *Versão* está associada a uma ou duas instâncias da entidade *Log*.

RI12 Uma instância da entidade *Log* está associada a duas e só duas instâncias da entidade *Versão*.

RI13 Na entidade *Log*, o atributo *log_id* é um inteiro positivo.

RI14 Quando se elimina um tuplo da tabela *Pessoa*, devem ser eliminados os tuplos com igual valor para o atributo *email* nas tabelas *Pergunta*, *Paginas*, *Tipos_de_registos*, *Log* e *Login*, caso existam.

- RI15** Quando se elimina um tuplo da tabela *Paginas*, devem ser eliminados os tuplos com valores idênticos para os atributos *nome_p* e *email* da tabela *Paginas_registos*, caso existam.
- RI16** Quando se elimina um tuplo da tabela *Registos*, devem ser eliminados os tuplos com valores idênticos para os atributos *nome_r*, *nome_t* e *email* das tabelas *Paginas_registos* e *Registo_Campos*, caso existam.
- RI17** Quando se elimina um tuplo da tabela *Versão*, devem ser eliminados os tuplos com igual valor para o atributo *id* nas tabelas *Paginas*, *Paginas_registos*, *Registos*, *Tipo_de_registos*, *Campos*, *Registo_Campos* e *Log_versao*, caso existam.
- RI18** Quando se elimina um tuplo da tabela *Tipos_de_registos*, devem ser eliminados os tuplos com valor idêntico para os atributos *nome_t* e *email* nas tabelas *Registos* e *Campos*, caso existam.
- RI19** Quando se elimina um tuplo da tabela *Campos*, devem ser eliminados os tuplos com valor idêntico para os atributos *nome_c*, *nome_t* e *email* nas tabelas *Registo_Campos*.
- RI20** Quando se elimina um tuplo da tabela *Log*, devem ser eliminados os tuplos com valor igual para os atributos *log_id* e *email* na tabela *Log_versao*.

3 Álgebra Relacional

3.1 Pergunta 1

$\Pi_{nome_t}(\sigma_{email = 'Manel@Notebook.pt'}(Tipo_de_registos))$

3.2 Pergunta 2

$\Pi_{email}(Pessoa \bowtie \sigma_{sucesso=falso}(Login))$

3.3 Pergunta 3

$\Pi_{timestamp_nascimento}(\sigma_{nome_p = 'facebook' \wedge nome_r = 'facebook'}(Pessoa \bowtie \rho(P, Paginas) \bowtie_{P.email = R.email} \rho(R, Registos)))$

4 Linguagem SQL

4.1 Pergunta 1

```
SELECT DISTINCT nome_t
FROM Tipo_de_registos
WHERE email="Manel@notebook.pt";
```

4.2 Pergunta 2

```
SELECT DISTINCT Pessoa.email
FROM Pessoa, Login
WHERE Pessoa.email = Login.email AND sucesso = false;
```

4.3 Pergunta 3

```
SELECT timestamp_nascimento
FROM Pessoa, Paginas, Registos
WHERE Pessoa.email = Registos.email
AND Pessoa.email = Paginas.email
AND Paginas.nome_p = "facebook"
AND Registos.nome_r = "facebook";
```