Departamento de Engenharia de Eletrónica e Telecomunicações e de Computadores

Licenciatura em Engenharia Informática e de Computadores



**Relatório Trabalho Prático: Fase 1**  
Sistemas de Informação 2  
Grupo 07

Bernardo Rodrigues – nº 44784  
Ricardo Bogas – nº 42394

Índice

1. Introdução
   1. Objetivos de aprendizagem
   2. Análise do enunciado
2. Construção do modelo de dados
3. **Introdução**

Para a primeira fase do trabalho prático da unidade curricular Sistemas de Informação 2 (SI2) é pedido que se realize o sistema de base de dados de um Instituto de ensino, juntamente com um conjunto de funcionalidades do sistema.

* 1. **Objectivos de Aprendizagem**

Com o presente trabalho, é pedido ao aluno que demonstre capacidades em:

- Desenvolver um modelo de dados adequado aos requisitos, normalizado até à 3NF;

- Utilizar corretamente controlo transacional;

- Utilizar corretamente níveis de isolamento;

- Utilizar corretamente vistas;

- Utilizar corretamente procedimentos armazenados;

- Utilizar corretamente gatilhos;

- Utilizar corretamente funções;

- Saber justificar o uso na solução apresentada de vistas, procedimentos armazenados, gatilhos e funções.

- Escrever um relatório técnico sobre o trabalho desenvolvido

**1.2 Análise do Enunciado**

Numa primeira instância, é pedido que, através da análise minuciosa do documento referente aos requisitos do sistema, se realize o modelo conceptual e relacional de dados.

Posteriormente devem ser realizadas um conjunto de alíneas com funcionalidades do sistema, garantindo sempre a integridade dos dados e a certeza no resultado pretendido.

1. **Construção do modelo de dados**
   1. **Análise do documento de requisitos do sistema e realização do modelo conceptual**

Depois da leitura e interpretação dos requisitos do sistema foi possível retirar a informação necessária para a realização do modelo de dados.

Através do texto realizou-se o Modelo Entidade-Associação do sistema (presente nos anexos, ModeloEASI2.png).

Para o *Departamento*, *Secção* e *Curso* criaram-se entidades, sendo a *Secção* uma entidade fraca pois necessita da presença de um departamento para identificação (vários departamentos podem ter as mesmas secções).

Á entidade *Curso* foi associada a entidade *Ano* que relata que anos e semestres estão presentes em cada curso.

Posteriormente criou-se a entidade *Professor*, que pertence a uma secção. Para que se pudesse ter um professor coordenador de uma secção, criaram-se duas associações entre ambas as entidades, uma que revela a que secção pertence o professor e outra que representa o professor coordenador de cada secção.

Após a criação da entidade *UC* (Unidade Curricular), criou-se a associação *Ensina* (N, M) para descrever que professores ensinam uma unidade curricular e que unidades curriculares são ensinadas por um professor (para suportar a característica de um professor poder leccionar várias UC e cada UC ser ensinada por um conjunto de professores).

Estas UCs estão inseridas em cursos, mas poderão aparecer em cursos diferentes. Então foi criada a associação *UCdeCurso* que, à semelhança da associação *Ensina*, representa as UC de cada curso e o curso em que cada UC é oferecida.

Criou-se a entidade *Aluno*. Cada aluno pode matricular-se em cursos (sendo que pode matricular-se em mais do que um). Tendo isto em consideração, à associação de *Aluno* com *Curso* foi dado o nome de *Matrícula*, com a data em que o aluno se matriculou no curso (data\_inic) e com a data de conclusão do mesmo (data\_conc, sendo que esta estará a *NULL* caso o aluno não tenha ainda terminado o curso).

Por fim foi criada a entidade fraca *Inscrição* associada a *Aluno* e a *UC.* Esta entidade representa a inscrição de um aluno numa unidade curricular.

* 1. **Criação do modelo relacional**

Assumindo a correcção do modelo criado no ponto anterior, criou-se o modelo relacional, com as relações que serão posteriormente criadas no modelo físico.

Posto isto, o modelo relacional apresentado é o seguinte:

Departamento(**sig\_un**, desc)   
RI: sig\_un tem um máximo de 6 caracteres.

Curso(**sig\_un**, desc, sig\_dep)  
RI: sig\_un tem um máximo de 6 caracteres.   
FK: sig\_dep referente a Departamento(sig\_un)

Aluno(**num\_aluno**, num\_cc, nome, endereço, cod\_post, localidade, data\_nasc)  
RI: num\_cc tem um máximo de 8 characteres; cod\_post tem um máximo de 7 caaracteres  
AK: num\_cc

Secção(**sig\_un**, desc, **sig\_dep**)  
RI: sig\_un tem um máximo de 6 caracteres  
FK: sig\_dep referente a Departamento(sig\_un)

Professor(**num\_cc**, nome, area\_esp, sig\_secção, sig\_dep, coord\_sec)  
FK: (sig\_secção, sig\_dep) referente a Secção(sig\_un, sig\_dep); (coord\_sec, sig\_dep) referente a Secção(sig\_un, sig\_dep)

UC(**sig\_un**, num\_creditos, desc)  
RI: sig\_un tem um máximo de 6 caracteres.

UCdeCurso(**sig\_uc**, **sig\_curs**, ano, semestre)  
RI: sig\_uc tem um máximo de 6 caracteres; sig\_curs tem um máximo de 6 caracteres; semestre é menor que 6.  
FK: sig\_uc referente a UC(sig\_un); sig\_curs referente a Curso(sig\_un)

Ano(**ano**, **semestre**, **sig\_curso**)  
RI: semestre entre 1 e 2;  
FK: sig\_curs referente a Curso(sig\_un)

Inscrição(**ano**, nota, **num\_aluno**, **sig\_uc**)  
RI: nota é menor ou igual a 20.  
FK: num\_aluno referente a Aluno(num\_aluno); sig\_uc referente a UC(sig\_un)

Ensina(**sig\_uc**, **prof\_cc**, ano)  
FK: sig\_uc referenta a UC(sig\_un); prof\_cc referente a Professor(num\_cc)

Matricula(**num\_aluno**, **sig\_curso**, data, média)  
RI: média é menor ou igual a 20  
FK: num\_aluno referente a Aluno(num\_aluno); sig\_curs referente a Curso(sig\_un)

Após a realização do modelo relacional, viu-se a necessidade de acrescentar uma nova relação que representa a matricula de um aluno num ano, possibilitando que o aluno se matricule em cursos diferentes em anos diferentes. Então criou-se a seguinte:

MatriculaAlunoEmAno(**ano**, **semestre**, **sig\_curs**, **num\_aluno**)  
RI: semestre entre 1 e 2  
FK: (ano, semestre, sig\_curs) referente a Ano(ano, semestre, sig\_curs)

* 1. Criação do modelo físico

Após criado o modelo relacional, passou-se para a linguagem T-SQL, traduzindo o modelo relacional (que é o mais próximo da linguagem) nas instruções de criação das tabelas. O documento *create\_drop.sql* dos anexos refere à criação do modelo físico.

1. Criação das funcionalidades
   1. Criação e remoção do modelo físico

Como enunciado no ponto anterior, no documento *create\_drop.sql* encontra-se o código T-SQL de criação do modelo físico. Juntamente, está o código para remoção do mesmo. Tanto para a criação como para a remoção teve-se em conta a ordem para que não houvesse erro na sua execução relativos a chaves estrangeiras.

* 1. Preencher o modelo físico com dados de teste

TODO

* 1. Inserir, Remover e atualizar informação de um departamento, secção e unidade curricular

Para a inserção, remoção e atualização dos dados referentes a um departamento, secção e UC foram criados *stored procedures*, como requisitado.

Foi utilizada a mesma lógica para todas as inserções, sendo esta a de verificação de existência dos dados antes da sua inserção, para prevenir algum erro e a existência de dados duplicados.  
  
Nas atualizações foi utilizada a mesma lógica que na inserção, com o pormenor de que a atualização pode não ser a todos os atributos de uma relação. Para isso assumiu-se que os parâmetros passados a NULL não seriam utilizados.

O código T-SQL destes procedimentos encontra-se nos anexos, documentos: *p\_inserirDepartamento.sql, p\_removerDepartamento.sql, p\_atualizarDepartamento.sql;  
p\_inserirSeccao.sql, p\_removerSeccao.sql, p\_atualizarSeccao.sql;*

* 1. Criar a estrutura geral de um curso

Para a criação da estrutura de um curso, sem unidades curriculares associadas, é necessário a inserção do curso com as seus atributos, passados como parâmetros ao procedimento, e dos anos que esse curso terá.

Como tal, verifica-se a existência do curso na base de dados, em primeiro lugar. Caso não exista então insere-se e entra-se num ciclo que irá inserir os anos na tabela, consoante o numero de anos que o curso tem

O código T-SQL destes procedimentos encontra-se nos anexos, documento *p\_criarCurso.sql*

* 1. Inserir e remover uma unidade curricular num semestre de um curso

A inserção e remoção de uma unidade curricular de um curso é feita através da inserção ou remoção da mesma na tabela UCdeCurso.

A lógica implementada na criação deste procedimento foi de criar a UC passada como parâmetro, caso ela não exista, e posteriormente inseri-la na tabela UCdeCurso

O código T-SQL destes procedimentos encontra-se nos anexos, documento *p\_inserirUCemCurso.sql*

* 1. Matricular aluno num curso, sem inscrições em unidades curriculares

Para matricular um aluno num curso segui-se a mesma lógica do ponto anterior. Caso seja o primeiro curso do aluno (neste caso identificado pelo seu cartão de cidadão) este é inserido a tabela Aluno. Posteriormente é introduzida a matricula do aluno no curso.

O código T-SQL destes procedimentos encontra-se nos anexos, documento *p\_matricularAlunoEmCurso.sql*

* 1. Inscrever aluno numa unidade curricular num dado ano
  2. Atribuir nota a uma unidade curricular frequentada por um aluno num dado ano
  3. Criar função TotalCréditos que, dado um curso, devolve o número total de créditos do curso
  4. Emitir automaticamente o certificado de conclusão de um curso para um dado aluno, após a conclusão da ultima UC.
  5. Listar os pares (sigla de unidade curricular, ano) para todas as unidades curriculares sem inscrições num período compreendido entre dois anos dados.
  6. Criar a vista CreditosCurso
  7. Obter os números dos alunos que, tendo feito a sua primeira matrícula num dado curso há mais de 3 anos, ainda não concluíram esse curso.