

# Complementos de Bases de Dados

2021/2022



Licenciatura em Engenharia Informática

Projeto – Época de Recurso

O projeto final da disciplina de Complementos de Bases de Dados visa a familiarização com a administração de bases de dados relacionais. O trabalho incidirá ao nível da BD e SGBD e sobre a lógica subjacente à utilização dos dados para os requisitos colocados; **não se pretende o desenvolvimento das camadas da lógica aplicacional e apresentação.**

## 1 Descrição Geral

O agrupamento de escolas “Agrupamento STB” tem uma nova direção.

Uma das medidas adotadas pela nova direção passa por desenvolver um novo sistema de informação que lhe permita gerir, de forma integrada, todo o processo de lançamento de notas das diversas escolas do agrupamento, como permitir também que os alunos e os encarregados de educação tenham acesso a essa informação.

Com a implementação do novo sistema torna-se necessário modelar e integrar, numa nova Base de Dados, os fragmentos de informação que foram exportados de sistemas existentes e que suportavam até à data as diversas escolas do agrupamento.

Constitui-se anexo a este enunciado um conjunto de ficheiros, que contêm os dados extraídos dos outros sistemas. No entanto estes apresentam-se fracamente relacionados, carecendo de uma otimização segundo as boas práticas de modelação e regras da normalização, com vista a providenciar um suporte eficiente das operações e ainda permitir intervenções no âmbito da sua escalabilidade.

## 2 Desenvolvimento

### 2.1 Modelo Relacional

Considerando os requisitos abaixo enumerados deverá ser desenvolvido, e carregado, um Modelo Relacional que os assegure de forma eficiente:

1. Deverão ser considerados campos adicionais ao dataset de modo a registar:
  - a. Número do Aluno

- b. Nome do Aluno
  - c. Moradas (tanto o aluno como o encarregado de educação podem ter mais de uma, sendo que apenas uma será a morada principal).
2. O sistema deverá apresentar dados multi-idioma. Os dados referentes às colunas *Mjob*, *Fjob*, *reason* e *guardian* devem apresentar os dados em Português (PT) e Inglês (EN). O sistema deve apresentar a robustez necessária para que sejam adicionados novos idiomas a qualquer momento e sem alterar a estrutura da BD.
  3. Deverão ser registadas as inscrições nas disciplinas de cada aluno, para determinado ano letivo.
  4. Deverá ser possível gerir os anos letivos, sendo **que apenas um ano letivo pode estar aberto** e permitir a gestão dos dados como o lançamento de notas ou inscrição de novos alunos nas disciplinas.
  5. Os dados do ano letivo corrente (inscrições nas disciplinas e notas do ano letivo) devem estar em tabelas apropriadas. Para o ano de 2021 (“Inscricoes2021”, “Notas2021”). A restante informação deverá ser guardada em tabelas de histórico (“InscricoesFechadas”, “NotasFechadas”). A informação do histórico deverá estar organizada pelo ano letivo que representa (ex: 2020).
- 6. Gestão de utilizadores:**
- a. Cada aluno poderá ter acesso às suas notas, pelo que deverão ser registados os dados de acesso à aplicação por parte do aluno.
  - b. Cada encarregado de educação também terá acesso aos dados dos seus educandos.
  - c. A autenticação perante a aplicação (por alunos e encarregados de educação) será feita com recurso à conta de email e password
  - d. Recuperar Password – O processo de recuperação de password é o seguinte:
    - O utilizador solicita a recuperação de password e é gerado um token que é enviado para o utilizador e só é válido por 1h
    - O utilizador com esse token pode invocar uma nova password, sendo necessário indicar o token e a nova password duas vezes)
  - e. O utilizador também pode alterar a password. Para isso deve indicar a password antiga e a nova password duas vezes)
  - f. Sempre que password do utilizador for alterada deve ser enviado um e-mail ao utilizador com a indicação de que a password foi alterada.
- Para efetuar a simulação do e-mail deverá registar essa informação numa tabela apropriada.

No desenvolvimento do modelo relacional deverá considerar as opções na definição pertinente dos tipos de dados, *default values*, *constraints* (*primary key*, *foreign key*, *unique* e *check*) e *triggers*.

## 2.2 Layout da BD

Para a definição do *layout* da base de dados, deverá suportar-se num conjunto de dados, nomeadamente:

- Espaço ocupado por registo de cada tabela;
- Espaço ocupado por cada tabela com o número atual de registos;
- Propor uma taxa de crescimento por tabela (inferindo dos dados existentes);
- Dimensionar o nº e tipos de acessos.

A definição dos Filegroups com o respetivo tipo, dimensão inicial, taxa de crescimento e dimensão máxima prevista deverá ser informada pelo levantamento enumerado.

## 2.3 Verificação da nova BD

**Dados originais:** Para os dados originais defina uma nova base de dados “OldData” para onde deve importar toda a informação do sistema antigo.

**Migração de dados:** Defina os scripts necessários para a migração dos dados antigos para a nova base de dados. **Todos os dados inseridos na base de dados deverão ser carregados via script.**

Para os dados não existentes no modelo de dados poderão ser criados dados “dummy”, como por exemplo “Nome Aluno 1”, “Nome Aluno 2”, etc...

De modo a verificar a conformidade da migração de dados, produza um conjunto de queries que quando executado sobre as duas bases de dados (OldData e nova bd depois da migração de dados), obtém o mesmo resultado.

Requisitos mínimos:

- Média de notas no ano letivo por escola
- Média de notas por disciplina
- Média de notas por disciplina/escola
- Total de alunos por escola/ano letivo

## 2.4 Programação

Todos os requisitos apresentados e passíveis de programação, deverão ser implementados através da solução apropriada e justificada, devendo ser criados Stored Procedures, Functions, Triggers ou outros

de acordo com a solução encontrada. As alíneas apresentadas no ponto 2.1 deverão ser obrigatoriamente implementadas.

**Complementarmente defina ainda *stored procedures* para:**

1. Abrir Ano Letivo (ver ponto 2.1.5)
2. Fechar Ano Letivo (ver ponto 2.1.5)
3. Inscrever alunos na disciplina
4. Atualização de nota do aluno numa determinada disciplina
5. Total de alunos de alunos inscritos em cada uma das disciplinas no ano aberto face ao ano anterior e a respetiva taxa de crescimento.
6. Média de notas de todos os anos por escola num determinado ano e a respetiva taxa de crescimento face ao ano anterior.

### **Tratamento de Erros**

Todo o código desenvolvido no projeto deverá fazer as validações necessárias e incluir tratamento de erros. O tratamento de erros deve ser gerido de forma centralizada. Na ocorrência de um erro deverá ser enviada uma mensagem amigável ao utilizador e criar um *log* de erro que identifique o erro gerado, o utilizador (sql server) e o *timestamp* do evento.

## **2.5 Catálogo/Metadados**

### **Abertura de ano letivo**

Como referido nos requisitos os dados referentes a inscrições no ano letivo e lançamento das respetivas notas apenas ocorrerão para o ano letivo aberto. No processo de abertura de ano letivo deverá criar as tabelas necessárias para efetuar o lançamento da informação a partir do catálogo da BD. Deverá ter como base as tabelas do ano letivo anterior que achar necessárias.

No momento de criação das tabelas tenha em consideração todas as restrições (*not null*, *primary key*, *foreign key*, ...) de modo a manter a coerência dos dados.

O processo deverá também inscrever automaticamente nas novas tabelas todos os alunos que não obtiveram aprovação nas várias disciplinas no ano anterior.

### **Fecho de ano letivo**

Sempre que é fechado um ano letivo, recorrendo ao catálogo, deverão ser copiados os dados para as tabelas de histórico.

## 2.6 Índices

Considerando que a secretaria do agrupamento necessita de fazer no final de cada ano letivo análises e relatórios sobre as notas dos alunos implemente:

- Criar uma nova base de dados STBRel, com o esquema necessário à implementação dos tópicos seguintes;
- Stored procedure que gere dados de histórico desde o ano 1960 até 2020 para:
  - 9 disciplinas
    - Português, Inglês, Frances, Matemática, Ciências, Físico-química, Educação Visual, TIC, Educação Física;
  - 4 escolas
    - Escola Fernando Pessoa, Escola José Saramago, Escola Eça de Queiros, Escola de Bocage;
  - Para cada ano/disciplina o número de alunos deve variar entre os 2000 e os 2500

Recorrendo só aos dados de histórico crie o seguinte conjunto de views:

- A taxa de crescimento de cada ano, face ao ano anterior, no que ao número de alunos diz respeito.
- Calcular, por ano, a percentagem de alunos com nota final (P3) maior ou igual a 15
- Calcular, por cada ano, a escola com melhor média final

### Notas:

- A taxa de crescimento calcula-se com a seguinte fórmula  $\frac{\text{ano atual} - \text{ano anterior}}{\text{ano anterior}}$

Com vista à otimização da execução das consultas propostas, defina, justificadamente, os índices pertinentes. Inclua no relatório a justificação apresentada.

### SQL Profiler e Tuning Advisor

Contraste os planos de execução das queries anteriores sobre a base de dados que otimizou e as queries equivalentes sobre a base de dados original. No caso da nova base de dados devem ser apresentados os planos de execução com e sem índices. Inclua no relatório o comentário ao que observa.

## 2.7 Backup e Recuperação

Defina e implemente uma estratégia de backup e recuperação para o Sistema de Informação (incluindo a base de dados de relatórios), concretizando aspetos como:

1. Modelo de Recuperação.
2. Tipo de Backup (integral, diferencial e log). Para este efeito, assuma e apresente os pressupostos quanto à carga do sistema, de forma a distribuir justificadamente o tipo de backup e definir a rotação dos dispositivos de armazenamento.
3. Simule vários cenários de ocorrência de “crash” da base de dados e apresente a sequência de recuperação. Descreva os vários cenários no relatório.
4. Supondo que as tabelas de base (exemplo: disciplinas, jobs, reason,..) não são sujeitas frequentemente a alterações, indique como poderia otimizar o plano de backups.

## 2.8 Níveis de acesso à informação

Defina os objetos necessários e os privilégios de acesso aos objetos da base de dados, para os seguintes tipos de utilizadores:

1. Administrador: Tem acesso a toda a informação.
2. LogUser por escola: Tem acesso a toda a informação da sua escola em modo de consulta.
3. Escola: Só tem acesso à informação dos seus alunos e apenas pode gerir as tabelas de estudantes, inscrições, notas e de histórico. Não pode eliminar dados das tabelas de histórico.

Nota: Para os pontos 2 e 3, deve apenas considerar as escolas GP – Gabriel Pereira e MS – Mousinho da Silveira.

Considere também a necessidade de implementar na base de dados as contas de acesso diferenciadas para cada um dos utilizadores anteriormente identificados.

## 2.9 Encriptação

Implemente o código necessário à encriptação ou hashing dos campos relativos à morada e password dos utilizadores do sistema. Justifique a escolha da metodologia escolhida para cada um dos campos. Inclua no relatório a justificação apresentada.

## 2.10 Controlo de transações

Defina os níveis de isolamento para os seguintes cenários:

- Execução das *stored procedures*: “Abrir ano letivo”, “Atualizar nota do aluno” e “Fechar ano letivo”, se executadas “simultaneamente” em sessões concorrentes num cenário de conflito;

Identifique e solucione ainda outros cenários possíveis que podem ocorrer no acesso concorrente ao site. Inclua no relatório a solução implementada para cada um dos cenários identificados.

## 2.11 MongoDB

Com o objetivo de permitir aos encarregados de educação acompanhar o percurso académico dos alunos pelos quais são responsáveis, o agrupamento de escolas “Agrupamento STB”, vai permitir a consulta, através de um site da seguinte informação:

- As notas de cada aluno por ano, período e disciplina
- Média final de cada aluno por ano e disciplina

De modo a não adicionar carga ao sistema atual, o site vai ser implementado num sistema autónomo, tendo por base o SGBD MongoDB, para tal devem ser implementadas as seguintes tarefas:

1. Criar a base de dados STBWeb;
2. Implemente as “collections” necessárias de modo a suportar a informação disponibilizada no site;
3. Defina no SQL Server as consultas necessárias para popular a base de dados STBWeb;
4. Importar para o MongoDB os resultados da alínea anterior;
5. Defina os comandos para:
  - Listar por Encarregado de Educação o “histórico de notas” dos alunos ao seu encargo;
  - Listar por aluno as notas de um determinado ano letivo, e a respectiva média final;
  - Listar por disciplina, os respetivos alunos e notas obtidas;
6. Adicionar a cada aluno informação que permita registar um relatório de avaliação no final de cada período: A informação a registar é a seguinte:
  - Período (P1, P2, ou P3)
  - Nome do Professor responsável pelo relatório
  - Relatório (texto livre)
  - Data

Insira alguns relatórios de exemplo, e apresente a informação adicionada ordenada pelo nome do Professor.

### 3 Regras de Execução

O não seguimento de qualquer das regras elencadas nas secções seguintes pode ser motivo de anulação do projeto.

#### 3.1 Grupos de Trabalho e Acompanhamento

O projeto deverá ser desenvolvido preferencialmente por grupos de 2 alunos.

Para o acompanhamento estarão disponíveis as aulas de laboratório e os horários de dúvidas de acordo com o publicado no *moodle*. Poderão ainda ser utilizados fóruns de discussão para o efeito que estão disponíveis no *moodle*.

#### 3.2 Calendário e Entregas

O projeto rege-se pelo seguinte calendário:

Data	Descrição
22/02/2022 até às 23h55	Entrega do Projeto

A entrega deve ser realizada através da plataforma *Moodle*, em *link* disponível para o efeito (discriminado por docente/turno de laboratório).

Deverá compor a entrega um ficheiro .ZIP com o seguinte conteúdo:

- Relatório
  - Documentando o projeto e justificações para as principais decisões tomadas em todas as secções de requisitos relativos a cada uma das fases de desenvolvimento.
  - Será disponibilizada no moodle a *template* para execução do relatório.
- Scripts desenvolvidos
  - O código implementado deverá ser devidamente comentado

O nome do ficheiro ZIP segue a seguinte nomenclatura: NumeroAluno1\_numeroAluno2.zip

Entregas fora de prazo serão alvo de uma penalização de 1 valor por hora nas 10 horas subsequentes à entrega, findas as quais o projeto será anulado.



Todos os materiais entregues (nomeadamente relatório e scripts) serão executados no software anti-plágio da instituição.

### 3.3 Avaliação

#### **Componentes de Avaliação**

A nota mínima do projeto é de 9,5 valores.

#### **Grelha de avaliação**

<b>Componente</b>	<b>Valor</b>
Modelo	12,5%
Layout + Migração de Dados	10%
Programação e Tratamento de Erros	12,5%
Metadados	5%
Índices	7,5%
Backup e Recuperação	7,5%
Níveis de acesso	7,5%
Encriptação	2,5%
Controlo de transações	7,5%
MongoDB	7,5%
Relatório e Apresentação	20%
<b>Total</b>	<b>100%</b>

#### **Discussões**

Em cada discussão o grupo deverá ter tudo preparado para realizar a demonstração e responder às questões do docente, assim que for solicitado no horário que lhe será atribuído e previamente publicado no moodle.

Durante a discussão as questões podem ser individualizadas. Consequentemente podem resultar notas diferentes para os elementos do grupo de trabalho.

### Anexos

Constituem-se anexos a este enunciado, disponíveis no moodle: 1) ficheiro zip com os dados originais.