

|  |
| --- |
| Complementos de Bases de Dados 2021/2022  Licenciatura em Engª. Informática |
| Relatório Técnico |

Turma: 2ºL\_EI-SW-03

Horário de Laboratório: 4ªF 11h30

Docente: Cláudio Sapateiro

Grupo

Nº 201000625, Tiago Paixão  
Nº 202000753, Nuno Reis

# Sumário Executivo

Este projeto visa a familiarização com a administração de bases de dados relacionais.

A direção do agrupamento de escolas “Agrupamento STB” tomou a decisão de desenvolver um novo sistema de informação que permita gerir, de forma integrada, todo o processo de lançamento das notas das diversas escolas, e permitir tambem que os alunos e os seus encarregados de educação tenham acesso a essa informação.

O nosso trabalho vai incidir na modelação e integração de uma nova base de dados, que dará apoio a esse novo sistema de informação. Foram exportados fragmentos de informação de sistemas existentes que suportavam as diversas escolas do agrupamento, estes apresentam-se fracamente relacionados e carecem de uma otimização.

Este relatório apresentará a documentação do projeto e algumas justificações para as principais decisões tomadas ao logo das fases de desenvolvimento.

# Especificação de Requisitos

|  |  |  |
| --- | --- | --- |
| ID | Descrição | Implementado  (S/N) |
| RF01 | O sistema deverá apresentar dados multi-idioma. | S |
| RF02 | O sistema devera permitir o registo de inscrições de cada aluno nas disciplinas. | S |
| RF03 | O sistema só devera permitir a gestão de dados de um ano letivo | S |
| RF04 | O sistema deve permitir o lançamento de notas | S |
| RF05 | O sistema deverá permitir a consulta de dados de anos letivos anteriores. | S |
| RF06 | O sistema deverá permitir o acesso do aluno e do encarregado de educação aos seus dados. | S |
| RF07 | O sistema deverá permitir aos utilizadores recuperar a password de acesso. | S |
| RF08 | O sistema deverá permitir aos utilizadores alterar a password de acesso. | S |
| RF09 | O sistema deverá calcular algumas estatísticas. | S |

# Alterações/Melhorias à Fase 1

Descrição das alterações e/ou melhorias que foram realizadas sobre o projeto apresentado na fase1. Incluir uma descrição sucinta (e.g., lista de tópicos) que permita inferir em concreto qual a alteração/melhoria implementada.

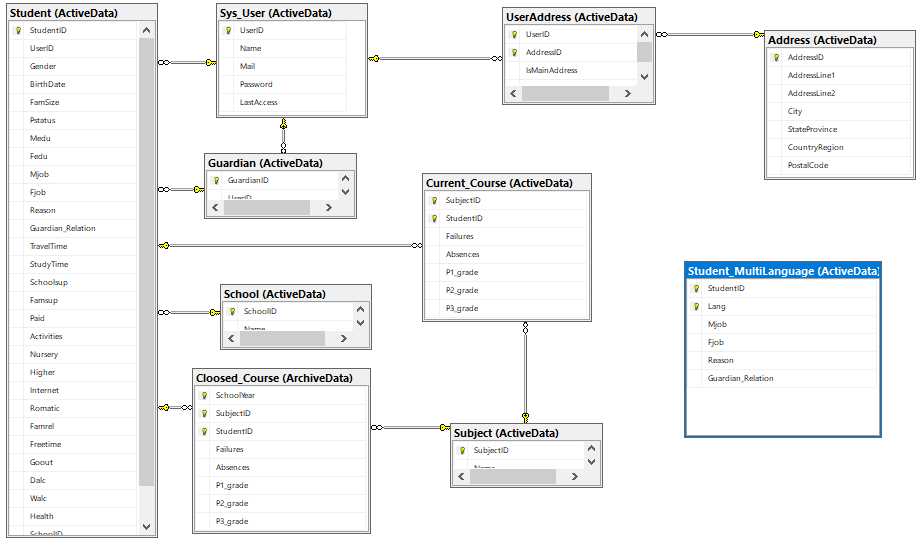
Passou a ser possível o acesso do aluno e do encarregado de educação aos seus dados, aos utilizadores recuperar a password de acesso e aos utilizadores alterar a password de acesso.

# Modelo Relacional (Modelo de dados)

## Diagrama do Modelo Entidade Relação

Imagem do diagrama entidade relação (opcional).

## Diagrama do Modelo Relacional



# Dimensionamento do Layout

|  |  |  |
| --- | --- | --- |
| Nome Tabela | Dimensão do Registo | Nº de Registos (inicial/final) |
| Subject | (4+80) 84 bytes por cada registo. | Inicial: 3  Final: |
| Address | (4+120+120+60+60+100+30) 494 bytes por cada registo, mais de 1300 registos por ano, ou seja, 642200 bytes por ano. | Inicial: 1947  Final: |
| Sys\_User | (4+12+140+70+8) 234 bytes por cada registo, 1300 por ano, ou seja, 304200 bytes por ano. | Inicial: 1947  Final: |
| UserAddress | (4+4+1+1) 10 bytes por cada registo, mais de 1300 por ano, ou seja, 13000 bytes por ano. | Inicial: 1947  Final: |
| School | (4+80) 84 bytes por cada registo. | Inicial: 2  Final: |
| Guardian | (4+4) 8 bytes por cada registo, 650 registos por ano, ou seja, 5200 bytes por ano. | Inicial: 0  Final: |
| Student | (4+4+1+3+1+1+1+1+40+40+40+40+1+1+1+1+1+1+1+1+1+1+1+1+1+1+1+1+4+4) 200 bytes por cada registo, 650 registos por ano, ou seja, 130000 bytes por ano. | Inicial: 1947  Final: |
| Student\_MultiLanguage | (4+10+20+20+20+40) 114 bytes por cada registo. | Inicial: 0  Final: |
| Cloosed\_Course | (2+4+4+1+1+1+1+1) 15 bytes por cada registo, 700 registos por ano (multiplicado por 3 anos) por ano, ou seja, 31500 bytes por ano. | Inicial: 5541  Final: |
| Current\_Course | (4+4+1+1+1+1+1) 13 bytes por cada registo. | Inicial: 0  Final: |
| ServidorMail |  |  |

|  |  |  |
| --- | --- | --- |
| Nome Filegroup | Tabelas associadas | Parâmetros |
| ActiveData | Subject  Address  Sys\_User  UserAddress  School  Guardian  Student  Student\_MultiLanguage  Current\_Course  ServidorMail | Dimensão inicial: 7901  Dimensão final: 10 405  Taxa de crescimento: 1 589 200 bytes |
| ArchiveData | Cloosed\_Course | Dimensão inicial: 5541  Dimensão final: 5541  Taxa de crescimento: |

# Schemas

|  |  |
| --- | --- |
| Nome | Descrição |
| OldData | Este schema tem como objetivo o suporte á importação dos dados, primeiro os dados são importados diretamente para este schema sem nenhum tratamento e posteriormente são importados para as tabelas correspondentes. |
| ActiveData | Este schema tem como objetivo o agrupamento das tabelas que contêm dados que podem ser alterados. |
| ArchiveData | Este schema tem como objetivo o agrupamento das tabelas que contêm dados que não podem ser alterados (dados apenas de leitura). |

# Views

|  |  |
| --- | --- |
| Nome | Descrição |
|  |  |

# Functions

|  |  |  |  |
| --- | --- | --- | --- |
| Nome | Tipo | Atributos | Descrição |
| dbo.fnHashPassword | String |  | Permite obter a codificação de uma password em SHA1. |
| dbo.fnGetStudentGrades | Table |  | Permite obter as notas do aluno |

# Stored procedures

|  |  |  |
| --- | --- | --- |
| Nome | Atributos | Descrição |
| dbo.insertNewStudent | @currentTime nvarchar (60)  @userID int  @addressID int | Permite adicionar um novo aluno. |
| dbo.insertNewSubject |  | Permite adicionar uma nova disciplina. |
| dbo.insertRowCloosed\_Course |  |  |
| dbo.registerStudentInSubject |  | Permite inscrever um aluno numa disciplina. |
| dbo.startSchoolYear | @StudentID int  @SubjectID int  @media float  @failures tinyint  StudentID\_cursor CURSOR FOR  SubjectID\_cursor CURSOR | Permite abrir um novo ano letivo. |
| dbo.endSchoolYear |  | Permite fechar o ano letivo atual. |
| dbo.spMediaNotasPorEscola | @school nvarchar(30)  @sum float  @average float  @count int | Permite calcular a media de notas no ano letivo por escola. |
| dbo.spMediaNotasPorEscolaPorPeriodo | @school nvarchar(30)  @sum float  @average float  @count int  @period int | Permite calcular a média de notas por ano letivo e período letivo por escola. |
| dbo.spLancarNota | @studentID int  @subjectID int | Permite lançar a nota de um aluno numa disciplina. |
| dbo.spInscreverAluno | @studentID int  @subjectID int | Permite inscrever um aluno numa disciplina. |
| dbo.spAtualizarNota | @studentID int  @subjectID int | Permite atualizar a nota de um aluno numa disciplina. |
| dbo.spTotalAlunosInscritos | @totalAlunos int  @taxaCrescimento float  @subject nvarchar(30) | Permite calcular o total de alunos inscritos em cada disciplina. |
| dbo.migrateOldData |  | Permite migrar os dados da OldData para as tabelas correspondentes. |
| dbo.spConsultarNotas | @guardianID int @guardian int | Permite consultar os dados de um aluno, pelo aluno ou pelo seu encarregado de educação. |
| dbo.changePassword | @mail nvarchar(70) @newPassword1 nvarchar(32) @newPassword2 nvarchar(32) | Permite aos utilizadores recuperar a password de acesso. |
| dbo.changeToNewPassword | @mail nvarchar(70) @oldPassword nvarchar(32) @newPassword1 nvarchar(32) @newPassword2 nvarchar(32) | Permite aos utilizadores alterar a password de acesso. |

# Triggers

|  |  |  |  |
| --- | --- | --- | --- |
| Nome | Tipo | Tabela | Descrição |
|  |  |  |  |

# Consultas

## Verificação da conformidade dos dados

*Descrever as consultas da verificação da conformidade dos dados no novo modelo.*

## Outras consultas

*Documentar outras consultas desenvolvidas no projeto.*

# Índices

## Views

|  |  |
| --- | --- |
| Nome | Descrição |
| *dbo.view\_TaxaCrescimento* | *Esta view permite a taxa de crescimento, …* |
|  |  |

## Índices

|  |  |  |
| --- | --- | --- |
| Designação | Tabela | Justificação/Consultas |
| *Nome do índice* | *Tabela indexada* | *Consulta otimizada, justificando as opções tomadas, e.g., colunas indexadas, colunas incluídas…* |
|  |  |  |

## Otimização e Execução de Consultas

*Apresentar os resultados da execução das consultas, com informação sobre as estatísticas e planos de execução, nos seguintes cenários:*

*- Sobre a base de dados original (não normalizada);*

*- Sobre a base de dados otimizada (normalizada) sem índices;*

*- Sobre a base de dados otimizada (normalizada) com índices.*

# Backup e Recuperação

*Descrição fundamentada da política de backups implementada, e descrição dos possíveis cenários de recuperação.*

# Segurança e Controlo de Acessos

*Definição de Utilizadores, Roles, Schemas e Encriptação.*

## Níveis de acesso à informação

## Encriptação

# Controlo de Concorrência

*Definir níveis de isolamento adotados no controlo transacional.*

# MongoDB

*Descrição do código implementado em MongoDB, justificando as opções tomadas.*

# Descrição da Demonstração

## Requisitos implementados

*Para os requisitos apresentados no capitulo 2, identificar a solução implementada através da referência ao código utilizado (restrições da BD, consultas/views, SPs, triggers,…).*

|  |  |
| --- | --- |
| ID | Solução |
| *RF0#* | *Restrição CHECK na tabela ….* |
| *RF##* | *SP\_nome\_... e Trigger tg\_....* |
|  |  |
| *RFM01* |  |

## Scripts de demonstração

*Identificar na tabela abaixo, a sequência de execução dos scripts com o código (consultas, chamadas a SPs,…) que permita verificar o correto funcionamento da base de dados face aos requisitos, por exemplo:*

*o Dados de teste que façam despoletar situação de inconformidade (e.g., RI implementadas com checks ou triggers);*

*o Script SQL para testar a execução dos sp;*

|  |  |  |
| --- | --- | --- |
| Número | Nome do ficheiro | Descrição |
| *1* | *RI.sql* | *Verificação das restrições de integridade.* |
|  |  |  |

# Conclusões

Com a realização deste projeto ficamos mais familiarizados com a administração de bases de dados relacionais, apesar de não termos conseguido implementar tudo o que nos era pedido. Conseguimos aplicar o conhecimento que adquirimos em anos anteriores na disciplina Bases de Dados e no ano atual.