

Complementos de Bases de Dados 2021/2022

Licenciatura em Eng^a. Informática

Relatório Técnico

Turma:

Horário de Laboratório:

Docente: Luís Cassaca

Grupo

Nº160221013, Gabriel Ambrósio

1. Sumário Executivo

O projeto final da disciplina de Complementos de Bases de Dados visa a familiarização com a administração de bases de dados relacionais.

O agrupamento de escolas “Agrupamento STB” tem uma nova direção. Para melhorar a gestão de dados interno, a nova direção decidiu que precisava de um novo sistema de informação, para melhorar todo o processo de lançamento de notas das escolas, como também permitir aos alunos e encarregados de educação tenham acesso a informação sobre as mesmas.

Esta reestruturação começa com a otimização do modelo relacional, com vista a providenciar um suporte eficiente das operações.

2. Especificação de Requisitos

ID	Descrição	Implementado (S/N)
RF01	O sistema deverá permitir registar alunos com dados adicionais, como número de aluno, nome, moradas, etc.	S
RF02	O sistema deverá permitir representar dados multi-idioma referentes às colunas Mjob, Fjob, reason e guardian.	N
RF03	O sistema deverá permitir registar inscrições nas disciplinas de um aluno, num determinado ano letivo.	S
RF04	O sistema deverá permitir gerir informação por ano letivo.	S
RF05	O sistema deverá permitir guardar dados que não pertencem ao ano letivo atual, em tabelas de histórico.	N
RF06	O sistema deverá permitir o acesso às notas aos alunos e encarregados.	S
RF07	O sistema deverá permitir fazer a autenticação com recurso a uma password e email.	S
RF08	O sistema deverá permitir recuperar a password.	N

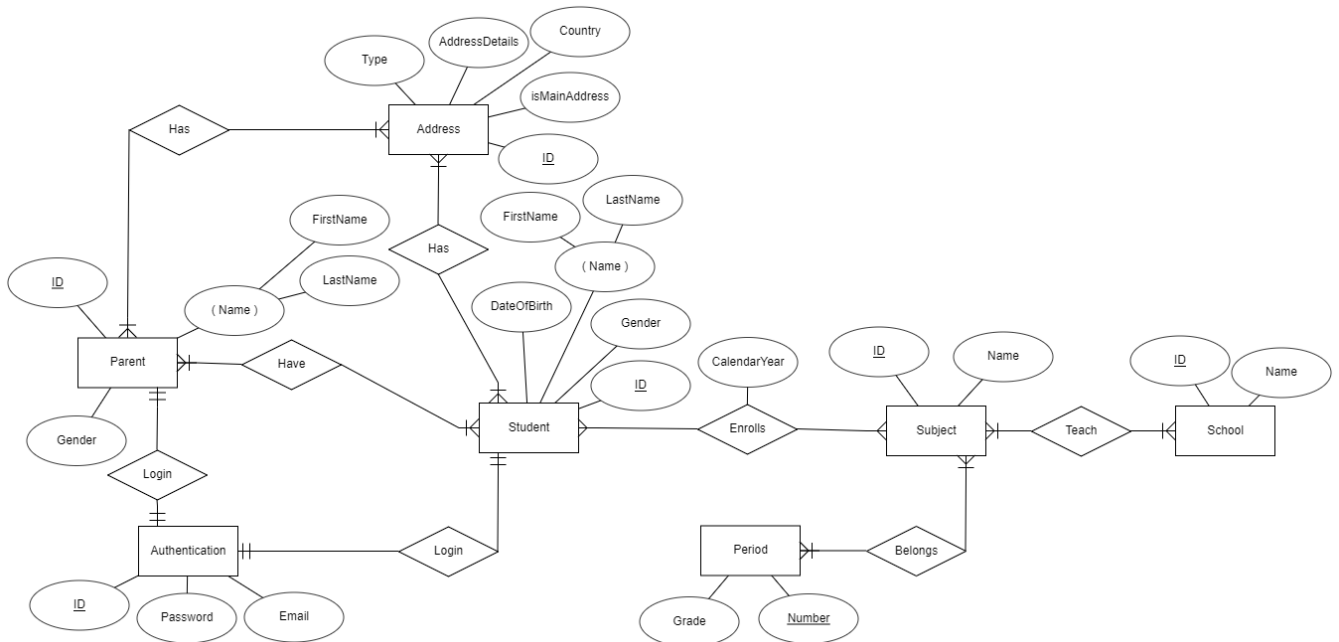
Relatório Técnico – Complementos de Bases de Dados

RF09	O sistema deverá permitir alterar a password.	N
RF10	O sistema deverá permitir enviar um email ao utilizador quando a password for alterada.	N
RF11	O sistema deverá permitir atualizar a nota de um aluno, numa determinada disciplina.	S
RF12	O sistema deverá permitir abrir um ano letivo	N
RF13	O sistema deverá permitir fechar um ano letivo	N
RF14	O sistema deverá permitir visualizar a média de notas de todos os anos por escola, num determinado ano.	S
RF15	O sistema deverá permitir visualizar a média de notas no ano letivo por escola	S
RF16	O sistema deverá permitir visualizar a média de notas por disciplina	S
RF17	O sistema deverá permitir visualizar a média de notas por disciplina/escola	S
RF18	O sistema deverá permitir visualizar o Total de alunos por escola/ano letivo	S

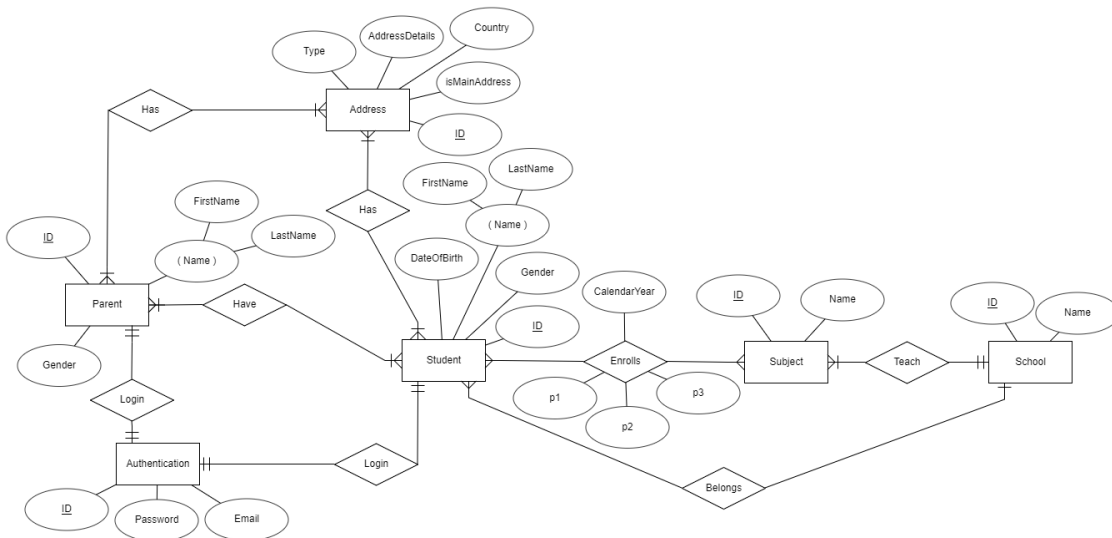
3. Modelo Relacional (Modelo de dados)

3.1 Diagrama do Modelo Entidade Relação

Versão 1

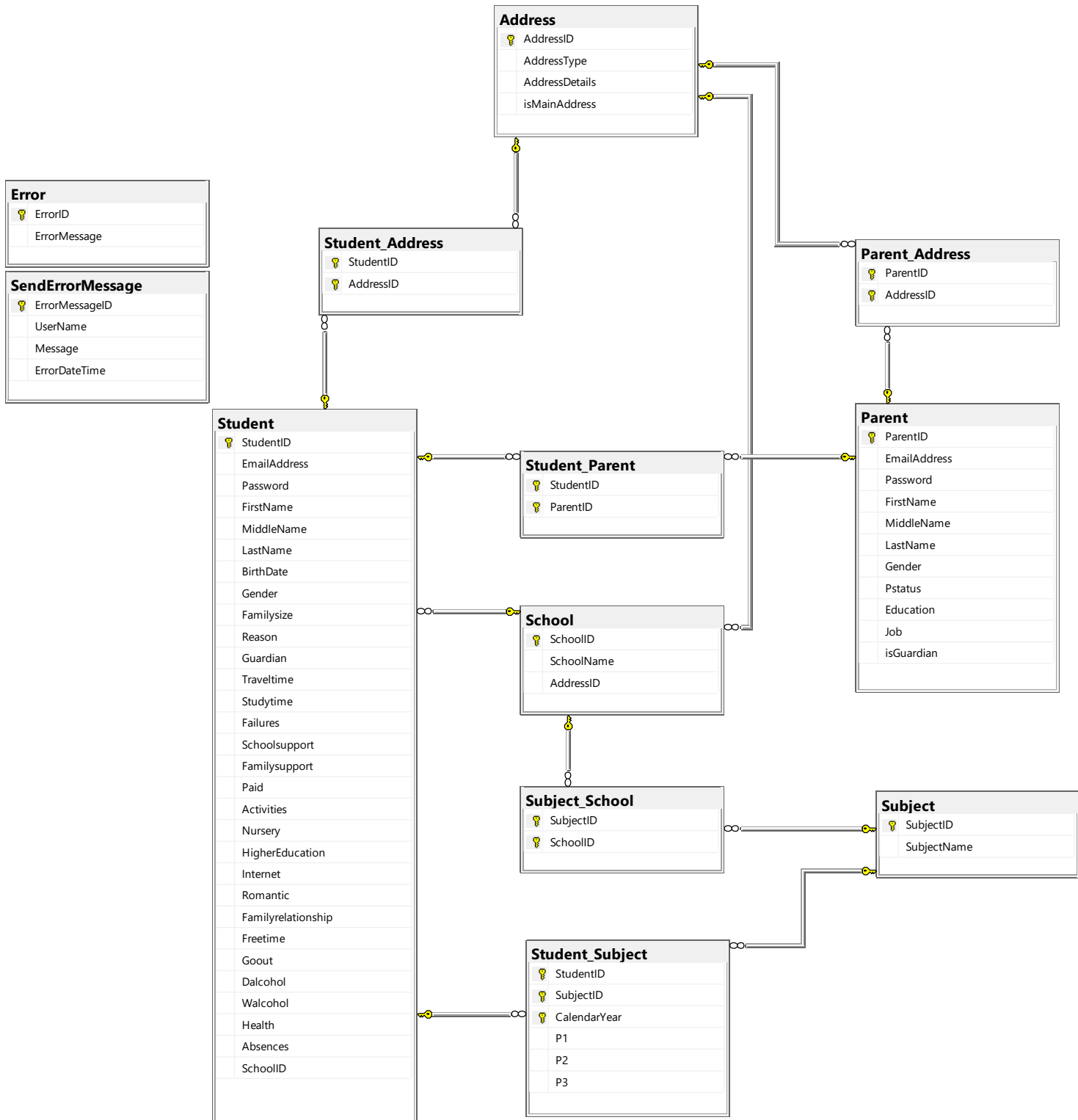


Versão 2 Final



3.2 Diagrama do Modelo Entidade Relação

Versão Final



Relatório Técnico – Complementos de Bases de Dados

4. Dimensionamento do Layout

Nome Tabela	Dimensão do Registo	Nº de Registos (inicial/final)
Student	(8KB*1024)/(4+50+50+50+50+50+3+1+1+ 15+10+1+1+1+1+1+1+1+1+1+1+1+1+1+1+ 1+1+1+1+4)= 26,77124183006536	1947
Parent	(8KB*1024)/(4+50+50+50+50+50+1+1+1+ 50+1)= 26,5974025974026	3894
Student_Parent (Parentesco)	(8KB*1024)/(4+4)= 1024	3894
Subject	(8KB*1024)/(4+50)= 151,7037037037037	3
Student_Subject (Inscrição)	(8KB*1024)/(4+4+4+1+1+1)= 546,1333333333333	5841
School	(8KB*1024)/(4+50+3)= 143,71929824561403	3
Subject_School (Escola ensina Disciplina)	(8KB*1024)/(4+4)= 1024	9
Address	(8KB*1024)/(4+1+50+1)=146,2857142857 1428	1949
Student_Address	(8KB*1024)/(4+4)= 1024	1947
Parent_Address	(8KB*1024)/(4+4) = 1024	3894

Relatório Técnico – Complementos de Bases de Dados

Nome Filegroup	Tabelas associadas	Parâmetros
PersonFG	Student, Parent, Student_Parent, Address, Student_Address, Parent_Address	Size = 50, Maxsize = 100, Filegrowth = 10
SchoolFG	School, Subject, Subject_School, Student_Subject	Size = 50, Maxsize = 100, Filegrowth = 10

5. Schemas

Nome	Descrição
Person	Este schema tem como objetivo agrupar as tabelas Student, Parent, Student_Parent, Address, Student_Address, Parent_Address.
School	Este schema tem como objetivo agrupar as tabelas School, Subject, Subject_School, Student_School.

6. Views

Nome	Descrição
dbo.view_oldDataInfo	Esta view permite obter uma lista com toda a informação disponível na base de dados antiga das escolas (Informação do excel).
Dbo.view_OldDataSchoolAverage	Esta view permite obter a lista de médias por escola da base de dados antiga.
Dbo.view_OldDataYearAverage	Esta view permite obter uma lista que contém a média de notas por ano civil da base de dados antiga.
Dbo.view_OldDataSchoolYearAverage	Esta view permite obter uma lista que contém a média de notas por escola e ano da base de dados antiga.

Relatório Técnico – Complementos de Bases de Dados

BD2017Average/CBD2017Average/MAT12017Average	Estas views permitem obter as médias das disciplinas num determinado ano, devido à forma como foram guardados os dados antigos, esta foi a melhor forma encontrada para conseguir mostrar esta informação.
Dbo.view_OldDataStudentTotalPerSchoolYear	Esta view permite obter uma lista que contém o número total de alunos por escola por ano da base de dados antiga.
Dbo.view_SchoolYearAverage	Esta view permite obter uma lista com a média de notas por escola e por ano da nova base de dados otimizada.
Dbo.view_SubjectAverage	Esta view permite obter as médias das disciplinas da nova base de dados.
Dbo.view_SubjectSchoolAverage	Esta view permite obter uma lista com as médias por disciplina e escola da base de dados otimizada.
Dbo.view_StudentTotalPerSchoolYear	Esta view permite obter uma lista o número total de alunos por escola e ano.

7. Functions

Nome	Tipo	Atributos	Descrição
dbo.viewGrades	Table	@StudentID INT	Permite obter as notas de um determinado aluno.

8. Stored procedures

Nome	Atributos	Descrição
dbo.sp_createStudent	@SchoolID int @EmailAddress nvarchar(50) @Password nvarchar(50) @FirstName nvarchar(50) @MiddleName nvarchar(50) @LastName nvarchar(50) @BirthDate date @Gender nvarchar(1) @FamilySize nvarchar(3) @Reason nvarchar(15) @Guardian nvarchar(10)	Permite adicionar um novo aluno.

Relatório Técnico – Complementos de Bases de Dados

	@Traveltime tinyint @Studytime tinyint @Failures tinyint @Schoolsupport nvarchar(3) @Famylsupport nvarchar(3) @Paid nvarchar(3) @Activities nvarchar(3) @Nursery nvarchar(3) @HigherEducation nvarchar(3) @Internet nvarchar(3) @Romantic nvarchar(3) @Familyrelationship tinyint @Freetime tinyint @Goout tinyint @Dalcohol tinyint @Walcohol tinyint @Health tinyint @Absences tinyint	
dbo.sp_createGuardian	@EmailAddress nvarchar(50) @Password nvarchar(50) @FirstName nvarchar(50) @MiddleName nvarchar(50) @LastName nvarchar(50) @Gender nvarchar(1) @Pstatus nvarchar(1) @Education tinyint @Job nvarchar(50) @isGuardian bit	Permite adicionar um novo encarregado.
dbo.sp_createRelationship	@StudentID int @ParentID int	Permite criar um parentesco entre um aluno e um encarregado.
dbo.sp_createAddress	@AddressDetails nvarchar(50) @AddressType nvarchar(1) @isMainAddress bit	Permite criar uma morada.
Dbo.sp_createSchool	@SchoolName nvarchar(50) @AddressID int	Permite criar uma escola.
Dbo.sp_createSubject	@SubjectName nvarchar(50)	Permite criar uma disciplina.
Dbo.sp_teachSubject	@SubjectID int @SchoolID int	Permite adicionar uma disciplina a uma escola.

Relatório Técnico – Complementos de Bases de Dados

dbo.sp_addStudentAddress dbo.sp_addParentAddress	@StudentID/@ParentID int @AddressID int	Permite adicionar um aluno/encarregado a uma morada
dbo.sp_enrollStudent	@StudentID int @SubjectID int @CalendarYear int	Permite inscrever um aluno numa disciplina num determinado ano.
dbo.sp_updateGrade	@StudentID int @SubjectID int @CalendarYear int @Period tinyint @Grade tinyint	Permite atualizar as notas de um aluno numa disciplina de um determinado ano.

9. Triggers

Nome	Tipo	Tabela	Descrição
dbo.trigger_codificarStudentPassword	AFTER INSERT	dbo.Student	Encripta a password de um estudante quando este é inserido na bd.
dbo.trigger_codificarStudentEmail	AFTER INSERT	dbo.Student	Encripta o email de um estudante quando este é inserido na bd.
dbo.trigger_codificarParentPassword	AFTER INSERT	dbo.Parent	Encripta a password de um encarregado quando este é inserido na bd.
dbo.trigger_codificarParentEmail	AFTER INSERT	dbo.Parent	Encripta o email de um encarregado quando este é inserido na bd.

10. Consultas

10.1 Verificação da conformidade dos dados

Consulta para verificar média de notas por disciplina por ano:

```

select SubjectID, CalendarYear, avg(notas) as 'Average' from (select StudentID,SubjectID, CalendarYear, ((cast(P1 + P2 + P3 as decimal))/3) as notas from ProjetoCBD.dbo.Student_Subject) as t
group by SubjectID, CalendarYear
order by SubjectID, CalendarYear

select t.SubjectID, SubjectName, CalendarYear, avg(notas) as 'Average' from (select StudentID,SubjectID, CalendarYear, ((cast(P1 + P2 + P3 as decimal))/3) as notas from ProjetoCBD.dbo.Student_Subject) as t
inner join Subject
on t.SubjectID = Subject.SubjectID
group by t.SubjectID, CalendarYear, SubjectName
order by SubjectID, CalendarYear

```

Relatório Técnico – Complementos de Bases de Dados

Consulta para verificar a média de todos os alunos por ano e disciplina.

```
--Media de todos os alunos por ano por disciplina
select Student.StudentID, t.SubjectID, CalendarYear, ((cast(P1 + P2 + P3 as decimal))/3) as notas, SchoolName from ProjetoCBD.dbo.Student_Subject as t
inner join Subject_School
on t.SubjectID = Subject_School.SubjectID
inner join School
on School.SchoolID = Subject_School.SchoolID
inner join Student
on Student.SchoolID = School.SchoolID and Student.StudentID = t.StudentID
```

Alternativa ao 4º requisito (Total de alunos por escola/ano letivo)

```
--Alternativa
select CalendarYear, SchoolName, count(distinct Student.StudentID) as 'Student Count' from ProjetoCBD.dbo.Student_Subject as t
inner join Subject_School
on t.SubjectID = Subject_School.SubjectID
inner join School
on School.SchoolID = Subject_School.SchoolID
inner join Student
on Student.SchoolID = School.SchoolID and Student.StudentID = t.StudentID
group by SchoolName, CalendarYear
order by CalendarYear, SchoolName
```

11. Backup e Recuperação

Em relação à política de backups, atendendo aos requisitos do projeto, optámos pela seguinte solução:

Backup completo semanal;

Backup diferencial diário;

Transaction log de hora a hora;

Full backup semanal: O full backup guarda toda a informação constante na base de dados numa única operação. Como regista o log sequence number, permite que seja possível recuperar a base de dados a partir de um dado momento temporal. Apesar de ser uma operação que demora algum tempo, compensa com o facto de garantir que todos os dados estarão presentes no caso de se precisar de contar com as cópias numa restauração. Como este tipo de backup guarda uma grande quantidade de dados, não há necessidade de o efetuar todos os dias. Assim é efetuado apenas semanalmente.

Diferential backup diário: O backup diferencial guarda os dados que tenham sido alterados desde o último backup, diferencial ou full, efetuado. Ou seja, para que se possa executar uma restauração basta utilizar o último backup full para comparar o conteúdo e copiar todas as alterações realizadas.

Transaction log: Ter um backup full semanal e um backup diferencial diário é necessário para limitar o número de backups de logs necessários para a restauração. A frequência com que devem ser feitos os backups dos logs

vai depender da tolerância à exposição de perda de trabalho balanceado/equilibrado pelo número de backups de log que podem ser armazenados, geridos e potencialmente restaurados.

12. Segurança e Controlo de Acessos

12.1 Níveis de acesso à informação

Pelo que foi pedido no enunciado foram criados três roles, AdminRole, UserRole, SchoolRole e cinco utilizadores, um admin, um aluno, um encarregado e duas escolas. Com base nos privilégios de cada um destes perfis, concedemos e negámos algumas permissões de visualização, execução, inserção, entre outras.

```
--Atribuições de permissões ao role de Administrador
GO
GRANT SELECT, INSERT, DELETE, UPDATE, EXECUTE on SCHEMA::dbo TO AdminRole
GO

--Atribuições de permissões ao role de Utilizador
GO
GRANT SELECT on SCHEMA::dbo TO UserRole
GO
GO
GRANT SELECT ON OBJECT::Student TO UserRole;
GO
GO
GRANT SELECT ON OBJECT::School TO UserRole;
GO
GO
GRANT SELECT ON OBJECT::Student_Subject TO UserRole;
GO
GO
GRANT SELECT ON OBJECT::Subject_School TO UserRole;
GO
GO
DENY SELECT ON OBJECT::Address TO UserRole;
GO

--Atribuições de permissões ao role da Escola
GO
GRANT SELECT on OBJECT::Student TO SchoolRole
GO

GO
DENY DELETE ON OBJECT::Student_Subject TO SchoolRole;
GO
```

12.2 Encriptação

Foi feita também a encriptação da password e email, dos alunos e encarregados que estão na base de dados.

```
-- Encriptar campos password
go
open symmetric key passwordKey decryption
by certificate EncryptCertificate
update Student
set encryptPassword =
ENCRYPTBYKEY(KEY_GUID('passwordKey'),Password)
go

go
open symmetric key passwordKey decryption
by certificate EncryptCertificate
update Parent
set encryptPassword =
ENCRYPTBYKEY(KEY_GUID('passwordKey'),Password)
go

-- Encriptar campo email
go
open symmetric key passwordKey decryption
by certificate EncryptCertificate
update Student
set encryptEmail =
ENCRYPTBYKEY(KEY_GUID('passwordKey'),EmailAddress)
go

go
open symmetric key passwordKey decryption
by certificate EncryptCertificate
update Parent
set encryptEmail =
ENCRYPTBYKEY(KEY_GUID('passwordKey'),EmailAddress)
go
```

Emails:

Results	Messages
EmailAddress	encryptEmail
1 NomeAluno1@gmail.com	0x001A4735B05DA149A4049B9F1517CC0402000000B42208...
2 NomeAluno2@gmail.com	0x001A4735B05DA149A4049B9F1517CC0402000000991C5C...
3 NomeAluno3@gmail.com	0x001A4735B05DA149A4049B9F1517CC04020000002FE32A...
4 NomeAluno4@gmail.com	0x001A4735B05DA149A4049B9F1517CC0402000000348AF26...
5 NomeAluno5@gmail.com	0x001A4735B05DA149A4049B9F1517CC040200000046E359...
6 NomeAluno6@gmail.com	0x001A4735B05DA149A4049B9F1517CC0402000000B060600...
7 NomeAluno7@gmail.com	0x001A4735B05DA149A4049B9F1517CC04020000009686BF...
8 NomeAluno8@gmail.com	0x001A4735B05DA149A4049B9F1517CC04020000001BE827...
9 NomeAluno9@gmail.com	0x001A4735B05DA149A4049B9F1517CC0402000000970FDA...
10 NomeAluno10@gmail.c...	0x001A4735B05DA149A4049B9F1517CC0402000000BF07F64...
11 NomeAluno11@gmail.c...	0x001A4735B05DA149A4049B9F1517CC0402000000D5D93...
12 NomeAluno12@gmail.c...	0x001A4735B05DA149A4049B9F1517CC040200000060911F2...
13 NomeAluno13@gmail.c...	0x001A4735B05DA149A4049B9F1517CC0402000000126EA6...
14 NomeAluno14@gmail.c...	0x001A4735B05DA149A4049B9F1517CC0402000000ED3C48...
15 NomeAluno15@gmail.c...	0x001A4735B05DA149A4049B9F1517CC040200000005F3BD...
16 NomeAluno16@gmail.c...	0x001A4735B05DA149A4049B9F1517CC040200000005E95FD...
17 NomeAluno17@gmail.c...	0x001A4735B05DA149A4049B9F1517CC0402000000B2303F...
18 NomeAluno18@gmail.c...	0x001A4735B05DA149A4049B9F1517CC0402000000ADCE5...

Query executed successfully.

Passwords:

Results		Messages
	Password	encryptPassword
1	123456	0x001A4735B05DA149A4049B9F1517CC0402000000ACE8757...
2	123456	0x001A4735B05DA149A4049B9F1517CC04020000003152BEF...
3	123456	0x001A4735B05DA149A4049B9F1517CC04020000000363271...
4	123456	0x001A4735B05DA149A4049B9F1517CC04020000008F6345B...
5	123456	0x001A4735B05DA149A4049B9F1517CC0402000000613A417...
6	123456	0x001A4735B05DA149A4049B9F1517CC040200000005746DE...
7	123456	0x001A4735B05DA149A4049B9F1517CC0402000000AEC2EC...
8	123456	0x001A4735B05DA149A4049B9F1517CC04020000001A6BF7E...
9	123456	0x001A4735B05DA149A4049B9F1517CC040200000005E09FE6...
10	123456	0x001A4735B05DA149A4049B9F1517CC0402000000FC79636...
11	123456	0x001A4735B05DA149A4049B9F1517CC0402000000036A0E89...
12	123456	0x001A4735B05DA149A4049B9F1517CC0402000000035E6801...
13	123456	0x001A4735B05DA149A4049B9F1517CC0402000000B437BFE...
14	123456	0x001A4735B05DA149A4049B9F1517CC040200000008A4650B...
15	123456	0x001A4735B05DA149A4049B9F1517CC0402000000062F82B8...
16	123456	0x001A4735B05DA149A4049B9F1517CC0402000000EC85FE9...
17	123456	0x001A4735B05DA149A4049B9F1517CC040200000005CEA5D...
18	123456	0x001A4735B05DA149A4049B9F1517CC0402000000DBE908...
19	123456	0x001A4735B05DA149A4049B9F1517CC0402000000000000...
Query executed successfully.		

13. Descrição da Demonstração

13.1 Requisitos implementados

ID	Solução
RF01	Usando a procedure sp_createStudent é possível adicionar alunos à nova BD com os novos dados.
RF03	Para inscrever um aluno numa disciplina é possível usar a procedure sp_enrollStudent.

Relatório Técnico – Complementos de Bases de Dados

RF04	Através das views criadas, é possível aceder a toda a informação relativa as notas por ano, e também através da procedures sp_updateGrade possível atualizar as notas relativas a um ano letivo.
RF06	Usando a view viewGrades é possível aceder às notas de um determinado aluno, através de roles criados os encarregados e os alunos deverão poder visualizar esta informação.
RF07	Foi tido em consideração a autenticação na criação das tabelas Student e Parent, estas contêm atributos para guardar a password e o email dos mesmos para depois poderem fazer login.
RF11	Através da procedure sp_updateGrade possível atualizar as notas relativas a um ano letivo de um determinado aluno pelo id.
RF14	É possível visualizar esta informação usando a view criada para o efeito, StudentAveragePerYearSubjectSchool.
RF15	É possível visualizar esta informação usando a view criada para o efeito, SchoolYearAverage.
RF16	É possível visualizar esta informação usando a view criada para o efeito, SubjectAverage.
RF17	É possível visualizar esta informação usando a view criada para o efeito, SubjectSchoolAverage.
RF18	É possível visualizar esta informação usando a view criada para o efeito, StudentTotalPerSchoolYear.

13.2 Scripts de demonstração

Número	Nome do ficheiro	Descrição
1	RI.sql	Verificação das restrições de integridade.
1	OldData.bak	Restore da base de dados antiga. O import da informação do excel não foi conseguida através de scripts, as tentativas foram também submetidas, mas foram encontrados muitos problemas. Logo foi necessário fazer import manualmente através da função Import Data do SQL SERVER, nas Tasks de uma BD.
2	Script 1 - Layout	Cria o layout para a nova base de dados usada no projeto.
3	Script 2 - Create	Cria todas as tabelas, relações e chaves necessárias para o projeto.

Relatório Técnico – Complementos de Bases de Dados

4	Script 3 - Populate	Vai popular a nova base de dados com a informação antiga, retirada da base de dados antiga OldData.
5	Script 1 - Permissões	Cria os roles e atribui as permissões.
6	Script 6 - Encriptação	Encripta todas as passwords e emails das pessoas existentes na BD.
7	Script 4 - Queries	Agora é possível fazer as queries de verificação da migração.
8	Script 5 - SP	Cria todas as stored procedures para os requisitos do projeto.
9	Script 7 - Backup e Restore	Por fim é possível fazer o backup e restore.

14. Conclusões

Das componentes deste projeto que foram concluídas, foram encontradas algumas dificuldades.

Primeiro na importação dos dados da base de dados antiga, foram tentados vários métodos como o OPENROWSET e OPENDATASOURCE, mas sempre deram vários problemas e erros. Para continuar o desenvolvimento do projeto o import foi feito através do SQL Server e da funcionalidade Import Data, que faz import de um ficheiro para uma tabela.

Em segundo não foi possível completar a parte de autenticação, são guardadas as informações necessárias, e também é possível encriptá-la, mas não é feito automaticamente através de triggers.

Em terceiro, na criação das stored procedures, algumas davam erros quando criadas (ex: sp_createStudent), apesar de serem criadas com sucesso e funcionarem como pretendidas.

Por fim também existiram dificuldades no backup e restore, foi possível fazer um backup full, diferencial e dos logs para um backup device, mas a base de dados ficava num estado infinito de (Restoring...), resolvido ao correr o seguinte código “RESTORE DATABASE ProjetoCBD WITH RECOVERY”.

No que será a maior falta neste projeto, devido a limitação de tempo, e por ser feito por apenas um aluno, não foi possível desenvolver os scripts necessários para a conclusão dos pontos de Metadados, Indices, Transactions e MongoDB.

Em suma, apesar de existirem limitações no projeto, penso que tratar-se de um projeto que poderia ser melhorado, mas mostra nas componentes concluídas que a matéria da disciplina foi aprendida e executada com sucesso.