

|  |
| --- |
| Complementos de Bases de Dados 2020/2021  Licenciatura em Engª. Informática |
| 1ª Fase Relatório Técnico |

Turma: SW02

Horário de Laboratório: 5ª feira 13h00

Docente: João Santos

Grupo:

Nº180221011, Tiago Farinha  
Nº180221015, Francisco Moura

# Sumário Executivo

Este relatório representa a documentação de análise, levantamento de requisitos e desenvolvimento de um projeto no âmbito da unidade curricular de Complementos de Bases de Dados.

O projeto consiste na reestruturação de uma base de dados. A nova base de dados será intitulada de AgrupamentoSTB e, a antiga, de OldSTB. Serão apresentados os requisitos funcionais da nova base de dados, assim como, as entidades que nela atuam e o modelo relacional. Além disso, serão também indicados procedures, novos dados etc. aplicados à nova base de dados.

# Especificação de Requisitos

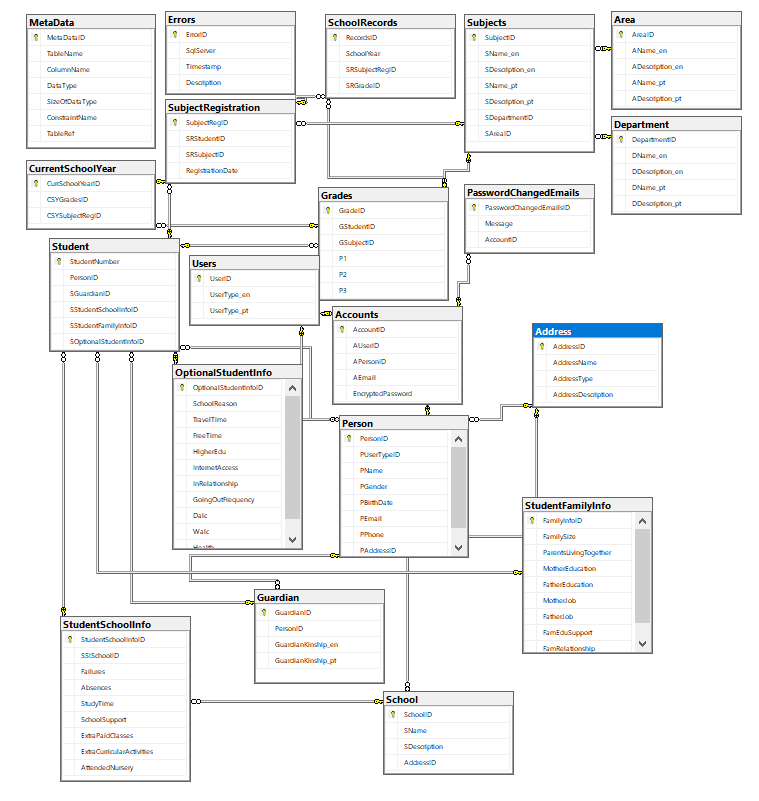
* 1. Requisitos funcionais

|  |  |  |
| --- | --- | --- |
| ID | Descrição | Implementado  (S/N) |
| *RF01* | *O sistema deverá permitir a um aluno/encarregado de educação criar uma conta e codificar a sua Password* | *S* |
| RF02 | O sistema deverá permitir alterar a password de um utilizador | S |
| RF03 | O sistema deverá permitir fazer a autenticação | +- |
| RF04 | O sistema deverá permitir adicionar um novo aluno | S |
| RF05 | O sistema deverá permitir adicionar um novo Encarregado de Educação | S |
| RF06 | O sistema deverá permitir registar as inscrições nas disciplinas de cada aluno para o ano letivo corrente | S |
| RF07 | O sistema deverá permitir a gestão dos anos letivos | S |
| RF08 | O sistema deverá permitir gerir os utilizadores e os seus acessos | +- |
| RF09 | O sistema deverá permitir adicionar dados multi-idioma | +- |
| RF10 | O sistema deverá permitir alterar as notas dos alunos num determinado Período | S |
| RF11 | O sistema deverá permitir inscrever um aluno numa determinada disciplina | S |
| RF12 | O sistema deverá permitir a abertura de um novo ano letivo | S |
| RF13 | O sistema deverá permitir o fecho do ano letivo corrente | S |
| RF14 | O sistema deverá permitir o cálculo das médias das notas e a sua taxa de crescimento relativamente a anos anteriores | S |
| RF15 | O sistema deverá permitir devolver uma mensagem de erro “amigável” aquando da execução de SPs, Functions etc com dados incorretos. | S |
| RF16 | O sistema deverá permitir o envio de um email aquando da alteração de password de um utilizador | S |

2.2Atores

|  |  |
| --- | --- |
| Nome | Descrição |
| Users | UserID (PK), UserType |
| Accounts | AccountID (PK), AEmail, EncryptedPassword, AUserID (FK) |
| Address | AddressID (PK), AddressName, AddressType, AddressDescription |
| Department | DepartmentID (PK), DName\_en, DDescription\_en , DName\_pt , DDescription\_pt |
| Area | AreaID (PK), AName\_en, ADescription\_en, AName\_pt, ADescription\_pt |
| Subjects | SubjectID (PK), SName\_en, SDescription\_en, SName\_pt, SDescription\_pt, SDepartmentID (FK), SAreaID (FK) |
| Guardian | GuardianID (PK), PersonID (FK), GuardianKinship\_en, GuardianKinship\_pt |
| Student | StudentNumber (PK), PersonID (FK), SGuardianID (FK), SStudentSchoolInfoID (FK), SStudentFamilyInfoID (FK), SOptionalStudentInfoID (FK) |
| Person | PersonID (PK), PersonRole\_en, PersonRole\_pt, PName, PGender, PBirthdate, PEmail, PPhone, PAddressID (FK) |
| School | SchoolID (PK), SName, SDescription, AddressID (FK) |
| StudentSchoolInfo | StudentSchoolInfoID, SSISchoolID (FK), Failures, Absences, StudyTime, SchoolSupport, ExtraPaidClasses, ExtraCurricularActivities, AttendedNursery |
| StudentFamilyInfo | FamilyInfoID (PK), FamilySize, ParentsLivingTogether, MotherEducation, FatherEducation, MotherJob, FatherJob, FamEduSupport, FamRelationship |
| OptionalStudentInfo | OptionalStudentInfo (PK), SchoolReason, TravelTime, FreeTime, HigherEdu, InternetAccess, InRelationship, GoingOutFrequency, Dalc, Walc, Health |
| SubjectRegistration | SubjectRegID (PK), SRStudentID (FK), SRSubjectID (FK), RegistrationDate |
| Grades | GradeID (PK), GStudentID (FK), GSubjectID (FK), P1, P2, P3 |
| SchoolRecords | RecordsID (PK), SchoolYear, SRSubjectRegID (FK), SRGradeID (FK) |
| CurrentSchoolYear | CurrSchoolYearID (PK), CSYGradesID (FK), CSYSubjectRegID (FK) |
| Erros | ErroID, SqlServer, Timestamp, Description |
| Metadata | MetadataID, TableName, ColumnName, DataType, SizeOfDataType, ConstraintName, TableRef |

# Modelo Relacional (*Modelo de dados*)



# Dimensionamento do Layout

*Descriminando a projeção de necessidades que conduz ao layout proposto*

*Identificação do espaço ocupado por tabela*

|  |  |  |
| --- | --- | --- |
| Nome Tabela | Dimensão do Registo (KB) | Nº de Registos (inicial/final) |
| *Accounts* | *0* | *0/0* |
| Address | 128 | *1947/1949* |
| Area | 16 | 2/2 |
| CurrentSchoolYear | 16 | 0/1 |
| Department | 16 | 2/2 |
| Errors | 0 | 0/0 |
| Grades | 208 | 5841/5842 |
| Guardian | 80 | 1947/1947 |
| MetaData | 16 | 0/111 |
| OptionalStudentInfo | 112 | 1947/1947 |
| PasswordChangedEmails | 0 | 0/0 |
| Person | 480 | 3894/3894 |
| School | 16 | 2/2 |
| SchoolRecords | 168 | 5841/5841 |
| Student | 80 | 1947/1947 |
| StudentFamilyInfo | 112 | 1947/1947 |
| StudentSchoolInfo | 80 | 1947/1947 |
| SubjectRegistration | 184 | 0/5842 |
| Subjects | 16 | 3/3 |
| Users | 48 | 0/3 |

# Views

|  |  |
| --- | --- |
| Nome | Descrição |
| Get\_NewID | *Esta view permite obter um novo ID aleatório.* |

# Functions

|  |  |  |  |
| --- | --- | --- | --- |
| Nome | Tipo | Atributos | Descrição |
| randomString | *Varchar(MAX)* | @startingString,  @counter | *Permite obter uma String aleatória de caracteres* |
| Hashing | Varchar(MAX) | @String | Permite encriptar uma String (neste caso, é usado para encriptar o email e a password dos utilizadores) |

# Stored procedures

|  |  |  |
| --- | --- | --- |
| Nome | Atributos | Descrição |
| AbrirAnoLetivo |  | *Permite criar a tabela do novo ano letivo usando o catálogo de metadados.* |
| FecharAnoLetivo |  | Permite passar os dados do ano letivo corrente para o histórico usando o catálogo de metadados |
| *InscreverAlunoDisciplina* | @StudentNumber,  @SubjectID | Permite inscrever um aluno numa disciplina |
| *AtualizarNota* | @StudentNumber, @SubjectID, @Year, @Period, @NewGrade | Permite atualizar a nota de um aluno numa certa disciplina e Período. |
| *AlunosInscritosETaxaCrescimento* | - | Permite ver o total de alunos de alunos inscritos em cada uma das disciplinas no ano aberto face ao ano anterior e a respetiva taxa de crescimento. |
| *MediaPorAnoETaxaCrescimento* | @year | Permite ver a média das notas de todos os anos por escola num determinado ano e a respetiva taxa de crescimento face ao ano anterior. |
| insertGrade | @temp\_GStudentID, @temp\_GSubjectID,  @temp\_grade1, @temp\_grade2, @temp\_grade3 | Permite adicionar notas a um determinado aluno numa determinada disciplina. |
| Autenticar | @email,  @Password | Permite fazer o login de um utilizador |
| Terminar\_Sessao | - | Permite terminar a sessão do utilizador que está com a sessão iniciada |
| Registar | @email,  @Password,  @UserType | Permite criar uma nova conta de Utilizador |
| AlterarPalavrapasse | @CurrPassword, @NewPassword1, @NewPassword2 | Permite a um utilizador alterar a password |
| AssociarFKStudentSchoolInfoID | - | Permite associar a FK StudentSchoolInfoID à tabela Student de forma a ligar as informações escolares dos alunos pertencentes à OldSTB |
| InscricoesDisciplinas | - | Permite popular as inscrições dos alunos nas disciplinas dos dados da OldSTB |
| RecuperarPalavraPasse | @email | Permite ao utilizador recuperar a palavra passe da sua conta |
| AcederInformacao | - | Permite ao utilizador aceder aos seus dados |
| CriarAluno | @StudentID, @Name, @Email, @GuardianID | Permite a criação de um novo aluno. |
| CriarEncarregado | @Name, @Email, @GuardianKinship\_en | Permite a criação de um novo Encarregado de Educação. |
| CriarDisciplina | @Name\_en, @Description\_en, @Name\_pt, @Description\_pt, @DepartmentID, @AreaID | Permite a criação de uma nova disciplina. |
| CriarArea | @Name\_en, @Description\_en, @Name\_pt, @Description\_pt | Permite a criação de uma nova área. |
| CriarDepartamento | @Name\_en, @Description\_en, @Name\_pt, @Description\_pt | Permite a criação de um novo departamento. |

# Consultas

## Verificação da conformidade dos dados

1. use AgrupamentoSTB;
2. -- |--------------------------------------------------------------------------------------|
3. -- | 2.1.3 - VERIFICAÇÃO DA NOVA BD |
4. -- |--------------------------------------------------------------------------------------|
5. -- a) Média de notas no ano letivo por escola
6. select SchoolYear as 'Ano', SDescription as 'Escola', avg((P1 + P2 + P3)/3) as 'Média' from Grades g
7. inner join Student s on g.GStudentID = s.StudentNumber
8. inner join SchoolRecords sr on g.GradeID = sr.SRGradeID
9. inner join StudentSchoolInfo ssi on s.SStudentSchoolInfoID = ssi.StudentSchoolInfoID
10. inner join School sc on ssi.SSISchoolID = sc.SchoolID
11. group by SchoolYear, SDescription order by SchoolYear;
12. select \* from Student
13. -- --------------------------------------------------------------------------------------
14. -- b) Média de notas por ano letivo e período letivo por escola
15. select SchoolYear as 'Ano', SDescription as 'Escola', avg(P1) as 'Média P1', avg(P2) as 'Média P2', avg(P3) as 'Média P3' from Grades g
16. inner join Student s on g.GStudentID = s.StudentNumber
17. inner join SchoolRecords sr on g.GradeID = sr.SRGradeID
18. inner join StudentSchoolInfo ssi on s.SStudentSchoolInfoID = ssi.StudentSchoolInfoID
19. inner join School sc on ssi.SSISchoolID = sc.SchoolID
20. group by SchoolYear, SDescription;
21. -- --------------------------------------------------------------------------------------
22. -- c) Total de alunos por escola/ano letivo
23. select SchoolYear as 'Ano', SDescription as 'Escola', COUNT(StudentNumber) / 3 as 'Número de alunos'
24. from SchoolRecords sr
25. inner join Grades g on sr.SRGradeID = g.GradeID
26. inner join Student s on g.GStudentID = s.StudentNumber
27. inner join StudentSchoolInfo ssi on s.SStudentSchoolInfoID = ssi.StudentSchoolInfoID
28. inner join School sc on ssi.SSISchoolID = sc.SchoolID
29. group by SchoolYear ,SDescription;
30. -- --------------------------------------------------------------------------------------
31. -- --------------------------------------------------------------------------------------
32. -- |--------------------------------------------------------------------------------------|
33. -- | EXTRAS/Outras Consultas |
34. -- |--------------------------------------------------------------------------------------|
35. -- OldSTB
36. use OldSTB;
37. -- 1) Número de alunos rapazes da escola Mousinho da Silveira inscritos em BD nos últimos 3 anos (240 alunos)
38. SELECT (SELECT COUNT([Student Number]) FROM dbo.[2017 student-BD] where school = 'MS' and sex = 'M')
39. +
40. (SELECT COUNT(Student\_Number) FROM dbo.[2018 student-BD] where school = 'MS' and sex = 'M')
41. +
42. (SELECT COUNT(Student\_Number)FROM dbo.[2019 student-BD] where school = 'MS' and sex = 'M'
43. ) AS 'Total de alunos rapazes da Mousinho Silveira'
44. -- 2) Número de alunos com notas iguais ou superiores a 10 no 1º Período
45. select (SELECT COUNT([Student Number]) FROM dbo.[2017 student-BD] where P1 >= 10)
46. +
47. (SELECT COUNT(Student\_Number) FROM dbo.[2018 student-BD] where P1 >= 10)
48. +
49. (SELECT COUNT(Student\_Number)FROM dbo.[2019 student-BD] where P1 >= 10
50. )
51. +
52. (SELECT COUNT(Student\_Number)FROM dbo.[2017 student-CBD] where P1 >= 10
53. )
54. +
55. (SELECT COUNT(Student\_Number)FROM dbo.[2018 student-CBD] where P1 >= 10
56. )
57. +
58. (SELECT COUNT(Student\_Number)FROM dbo.[2019 student-CBD] where P1 >= 10
59. )
60. +
61. (SELECT COUNT(Student\_Number)FROM dbo.[2017 student-MAT1] where P1 >= 10
62. )
63. +
64. (SELECT COUNT(Student\_Number)FROM dbo.[2018 student-MAT1] where P1 >= 10
65. )
66. +
67. (SELECT COUNT(Student\_Number)FROM dbo.[2019 student-MAT1] where P1 >= 10
68. ) AS 'Total de alunos com nota >= 10 no P1'
69. -- 3) Dados de um aluno específico
70. select [Student Number], school, sex, [Birth Date], guardian from dbo.[2017 student-BD] where [Student Number] = 1;
71. -- 4) Disciplinas, Áreas e Departamentos
72. select (select distinct Class from dbo.[2017 student-BD]) union
73. (select distinct Class from dbo.[2017 student-CBD]) union
74. (select distinct Class from dbo.[2017 student-MAT1]);
75. -- 5) Número de chumbos de um aluno ao acaso
76. select [Student Number] as 'Número de Estudante', convert(int, a.failures) + convert(int, b.failures) + convert(int, c.failures) as 'Nº de Chumbos' from dbo.[2017 student-BD] a join [dbo].[2017 student-CBD] b on a.[Student Number] =b.Student\_Number
77. join dbo.[2017 student-MAT1] c on b.Student\_Number = c.Student\_Number where [Student Number] = 164;
78. -- ----------------------------------------------------------------------------------------
79. -- AgrupamentoSTB
80. use AgrupamentoSTB;
81. -- 1) Número de alunos rapazes da escola Mousinho da Silveira inscritos em BD nos últimos 3 anos (240 alunos)
82. select count(\*) as 'Total de alunos rapazes da Mousinho Silveira' from Person p
83. inner join Student s on p.PersonID = s.PersonID
84. inner join StudentSchoolInfo ssi on s.SStudentSchoolInfoID = ssi.StudentSchoolInfoID
85. inner join School sc on ssi.SSISchoolID = SchoolID
86. where p.PGender = 'M' and sc.SName = 'MS';
87. -- 2) Número de alunos com notas iguais ou superiores a 10 no 1º Período
88. select count(P1) from Grades where P1 >= 10;
89. -- 3) Dados de um aluno especifico
90. select StudentNumber, SName, PGender, PBirthDate, GuardianKinship\_en from Student s
91. inner join Person p on s.PersonID = p.PersonID
92. inner join StudentSchoolInfo ssi on s.SStudentSchoolInfoID = ssi.StudentSchoolInfoID
93. inner join School sc on ssi.SSISchoolID = sc.SchoolID
94. inner join Guardian g on s.SGuardianID = g.GuardianID
95. where StudentNumber = 1;
96. -- 4) Disciplinas, Áreas e Departamentos
97. select SName\_pt as UC, SDescription\_pt as 'Descrição da UC', AName\_pt as 'Área', ADescription\_pt as 'Descrição da Área',
98. DName\_pt as Departamento, DDescription\_pt as 'Descrição do Departamento'
99. from Subjects s inner join Area a on s.SAreaID = a.AreaID
100. inner join Department d on s.SDepartmentID = DepartmentID;
101. -- 5) Número de chumbos de um aluno ao acaso
102. select StudentNumber, Failures from Student s inner join StudentSchoolInfo ssi on s.SStudentSchoolInfoID = ssi.StudentSchoolInfoID
103. where StudentNumber = 164;
104. select \* from Student;

## Outras consultas

select \* from Accounts;

select \* from Address;

select \* from Area;

select \* from CurrentSchoolYear;

select \* from Department;

select \* from Errors;

select \* from Grades;

select \* from Guardian;

select \* from MetaData;

select \* from OptionalStudentInfo;

select \* from PasswordChangedEmails;

select \* from Person;

select \* from School;

select \* from SchoolRecords;

select \* from Student;

select \* from StudentFamilyInfo;

select \* from StudentSchoolInfo;

select \* from SubjectRegistration;

select \* from Subjects;

select \* from Users;

# Descrição da Demonstração

*Para o correto funcionamento da nova base de dados (AgrupamentoSTB) é necessário correr os scripts na seguinte ordem:*

1. *DataBaseCreation\_Layout;*
2. *CREATETabelasNovaBD;*
3. *Migração de Dados;*
4. *NewData;*
5. *MetaData;*
6. *ScriptVerificaçãoDaBD;*
7. *ScriptGestãoUtilizadores;*
8. *ScriptStoredProcedures;*

# Conclusões

Podemos concluir, com a finalização desta 1ª fase e para uma 1ª normalização, que o nosso projeto cumpre a maioria dos requisitos pedidos de forma relativamente eficiente e de forma efetivamente rápida.

Aspetos onde podemos definitivamente melhorar será o suporte multilingue e uma maior dispersão da informação para outras subcategorias.