

**Sistemas de Informação 2**

**Docente:** Nuno Datia

**Grupo:** G12

**Alunos:**

Gonçalo Azevedo, nº 43554

Bruno Canelas, nº 43555

12 novembro 2018

***Introdução:***

Nesta primeira fase do projeto da unidade curricular de Sistemas de Informação 2 foi necessário criar uma base de dados para armazenamento de dados sobre Conferências.

Em primeiro lugar foi necessário criar modelos de dados de suporte à base de dados, sendo eles um Modelo Entidade-Associação e um Modelo Relacional. Ambos os modelos poderão ser conferidos em pontos mais abaixo.

De seguida foi criada a base de dados utilizando a linguagem T-SQL. Toda a implementação da base de dados poderá ser conferida em ficheiros enviados em anexo na pasta denominada “Files\_SQL”.

Ao longo da implementação foi necessário resolver as restrições de integridade envolvidas no problema. Ao longo do relatório serão apresentadas as restrições, bem como a forma utilizada para a sua resolução.

***Modelo Entidade – Associação:***

O Modelo Entidade – Associação apresentado na figura “Modelo 1”, tem como objetivo apresentar uma solução para as associações entre as entidades necessárias para a base de dados. No modelo apresentado encontra-se uma possível solução para as associações.

***Modelo Relacional:***

A partir da criação do Modelo Entidade – Associação, foi criado o Modelo Relacional para a base de dados. Este pode ser encontrado na figura “Modelo 2” e tem como objetivo facilitar a compreensão das associações realizadas no modelo anterior. O modelo apresentado foi adaptado de forma a ser possível uma melhor implementação da base de dados em linguagem SQL. Essas alterações foram necessárias para que o modelo respeitasse a 3ª Forma Normal.

***Descrição da Resolução:***

Após a realização dos dois modelos foi então iniciada a implementação da base de dados em SQL. Para cada uma das tabelas do Modelo Relacional foi também criada uma vista respetiva. Esta vista serve para disponibilizar ao utilizador apenas a informação importante e necessária para o correto funcionamento da base de dados. Cada uma das tabelas e respetiva vista será explicada em seguida.

*Instituição:*

Uma Instituição contem os seguintes descritores:

* Nome;
* País;
* Morada;

O identificador único de uma Instituição na base de dados é o seu nome, e a partir deste conseguimos realizar todas as operações necessárias. Na vista de uma Instituição um utilizador pode verificar toda a informação armazenada da Instituição.

A Instituição não contém qualquer restrição de integridade, como tal não serão utilizados Triggers os Procedimentos Armazenados.

*Utilizador:*

Um utilizador contém os seguintes descritores:

* Id;
* Email;
* Nome;
* Nome de Instituição;

O identificador único de um Utilizador é o seu Id na base de dados. O Nome da Instituição é uma chave estrangeira para a Instituição. Isto permite-nos saber a que instituição o Utilizador pertence.

Tal como a Instituição, o Utilizador não contém restrições de integridade, logo não são utilizados Tiggers nem Procedimentos Armazenados.

*Conferência:*

Uma Conferência é caracterizada por:

* Nome;
* Ano;
* Acrónimo;
* Id de Presidente;
* Data limite de Submissão de Artigos;
* Data limite de Revisão de Artigos;

O identificador único de uma Conferência na base de dados é a combinação do seu Nome com o seu Ano de realização, o que permite que a mesma conferencia possa ser realizada em anos diferentes. O Id de Presidente é uma chave estrangeira para Utilizador e permite saber qual o presidente de uma Conferência.

A vista e uma Conferência mostra todos os seus descritores, exceto o Id de Presidente que é substituído pelo Email de Presidente, pois a compreensão fica mais simples para um utilizador da base de dados.

Ao contrário do que acontece com a Instituição e o Utilizador, a Conferência contém restrições de integridade e como tal são utilizados Triggers para as resolver. Como tal forma criados 3 Triggers:

* INSERT: Ao inserir uma conferência a Data limite de Submissão de Artigos não pode ser Superior à Data limite de Revisão de Artigos, caso isto aconteça, a Conferência não é inserida; Sempre que inserimos esta tem que ter um presidente, como tal, ao fazermos a sua inserção é também criado um Registo para o Id de Presidente pretendido.
* DELETE: Apagar uma Conferência apenas é possível se esta não tiver artigos associados; Ao apagar uma Conferência é também necessário apagar os seus Registos de utilizadores, presidentes, autores e revisores, o que é resolvido pelo Trigger.
* UPDATE: Numa Conferência apenas se pode alterar o Acrónimo e Id de Presidente; Sempre que se altera o Id de Presidente tem que se garantir que existem Utilizadores possíveis de serem selecionados como novo presidente. O Trigger tem essa funcionalidade, ou seja, caso não seja possível realizar a alteração esta fica sem efeito; O Trigger tem também a funcionalidade de passar o presidente atual a utilizador.

*Registo:*

Um Registo tem os seguintes descritores:

* Id de Utilizador;
* Nome de Conferência;
* Ano de Conferência;
* Posição;

O identificador de um Registo na base de dados é a combinação do Id de Utilizador, Nome de Conferência e Ano de Conferência. O Id de Utilizador é uma chave estrangeira para Utilizador e permite identificar a que Utilizador pertence o Registo, enquanto que o Nome de Conferência e Ano de Conferência são uma chave estrangeira para Conferência e permitem identificar a que Conferência pertence o Registo.

A vista de um Registo substitui o Id de Utilizador pelo seu Email e mantém os seguintes descritores.

*Artigo:*

Um Artigo é descrito da seguinte forma:

* Id;
* Resumo;
* Estado;
* Nome de Conferência;
* Ano de Conferência;
* Data de Submissão;

O seu identificador único na base de dados é o Id, e o Nome de Conferência e Ano de Conferência são a chave estrangeira que permite identificar qual a conferência a que pertence o Artigo. O estado do Artigo pode ser: aceite, rejeitado ou em revisão.

A vista de um Artigo contém todos os seus descritores.

Para resolver as suas restrições de integridade foram utilizados Triggers para cada uma das operações:

* INSERT: Sempre que é inserido um Artigo, este tem que estar no de estado “em revisão”, caso contrário não pode ser inserido; Não podem ser realizadas submissões de Artigos cuja Data de Submissão é superior à Data limite de Submissão de uma Conferência.
* DELETE: apenas podem ser apagados Artigos que ainda se encontram dentro da Data Limite de Submissão definida pela Conferência, caso a data tenha terminado não podem ser apagados pois já se encontram no período de revisão.
* UPDATE: Apenas se pode alterar o resumo de um Artigo.

*Autor:*

Um Autor tem os seguintes descritores:

* Id de Autor;
* Id de Artigo;

O seu identificador na base de dados é a combinação dos dois descritores, e o Id de Artigo é uma chave estrangeira para Artigo que permite saber a que Artigo pertence o Autor.

A vista de um Autor substitui o Id de Autor pelo Seu Email.

Para resolver restrições de integridade do Autor foi necessário criar Triggers para as suas operações:

* INSERT: Não se pode adicionar um Autor a um Artigo caso a data limite de submissão de artigos tenha sido ultrapassada; Caso não existam Utilizadores disponíveis para serem Autores, então não se pode adicionar um Autor ao Artigo.
* DELETE: Não se podem remover Autores de Artigos caso a data limite de submissão de Artigos tenha sido ultrapassada; Para se remover um Autor de um Artigo, primeiro é necessário torna-lo um Utilizador.
* UPDATE: Não é possível realizar alterações a Autores.

*Ficheiro:*

Um ficheiro é descrito por:

* Id;
* Texto;
* Id de Artigo;

O seu identificador na base de dados é o Id e o Id de Artigo é uma chave estrangeira de Artigo que permite saber a que Artigo o Ficheiro pertence.

A vista de um Ficheiro contém todos os descritores de Ficheiro.

Para resolver restrições de integridade foi necessária a criação de Triggers nas operações necessárias:

* INSERT: Não é possível adicionar um Ficheiro caso a data limite de submissão de Artigos tenha sido ultrapassada.
* DELETE: Não é possível apagar um Ficheiro de um Artigo caso a data limite de submissão de Artigos tenha sido ultrapassada.
* UPDATE: Não é possível realizar alterações a um Ficheiro caso a data limite de submissão de Artigos tenha sido ultrapassada.

*Revisor:*

Um Revisor é descrito por:

* Id de Artigo;
* Id de Revisor;

O seu identificador na base de dados é a combinação dos seus dois descritores. O Id de Artigo é uma chave estrangeira para Artigo e serve para identificar que Artigo o Revisor irá rever. O Id de Revisor é uma chave estrangeira para Utilizador e serve para obter as informações relativas ao Revisor.

A vista de Revisor contém um Email a substituir o Id de Revisor.

Para resolver restrições de integridade foram criados Triggers nas operações do Revisor:

* INSERT: Não é possível adicionar um Revisor caso não haja Utilizadores disponíveis; Ao inserir um Revisor altera-se a posição de um Utilizador.
* DELETE: Para se remover um Revisor de uma Revisão, altera-se a sua posição de só depois se pode remover.
* UPDATE: Não é possível fazer alterações a um Revisor.

*Revisão:*

Uma Revisão é descrita por:

* Id de Artigo;
* Id de Revisor;
* Texto;
* Nota;
* Data de Revisão;

O seu identificador na base de dados é o conjunto de Id de Artigo e Id de Revisor. O Id de Revisor e Id de Artigo são a chave estrangeira para Revisor que permite obter a informação de quem revê o Artigo.

A vista da Revisão substitui o Id de Revisor pelo seu Email.

Para a resolução de restrições de integridade foram criados Triggers para as operações da Revisão:

* INSERT: Apenas é possível inserir uma Revisão caso a data limite de Revisao não tenha sido ultrapassada.
* DELETE: Não é possível remover uma Revisão caso a data limite de Revisão tenha sido ultrapassada.
* UPDATE: Caso a data limite de Revisão tenha sido ultrapassada não é possível fazer alterações a uma Revisão; Apenas se pode alterar a Nota e o Resumo de um Revisão.