

OurBook - Base de Dados

Relatório

André Butuc, 103530 [50%]
Gonçalo Silva 103668 [50%]

Índice

OurBook - Base de Dados	1
Índice	2
Introdução	3
Análise de Requisitos	4
Atores	4
Tabela 1 - Atores	4
Requisitos Funcionais	4
Tabela 2 - Requisitos Funcionais	5
Entidades do Sistema	5
Tabela 3 - Entidades do Sistema	6
Diagrama Entidade-Relacionamento (DER)	6
Imagem 1 - 1ª versão do Diagrama Entidade-Relacionamento	6
Imagem 2 - 2ª versão do Diagrama Entidade-Relacionamento	7
Esquema Relacional	8
Imagem 3 - 2ª versão do Esquema Relacional	9
Estrutura da Base de Dados no SQL Server	9
SQL DDL Tabela Utilizador	10
SQL DDL Tabela Livro	10
SQL DDL Tabela Formulário Troca	10
SQL DDL Tabela Formulário Depósito	11
Relação Tabelas da Base de Dados - Formulários Gráficos	11
User Form	11
BookPage Form	11
Profile Form	11
Courier Form	12
Índices - Base de Dados	12
Stored Procedures - Base de Dados	13
Triggers - Base de Dados	14
Conclusão	15

Introdução

Este relatório tem como propósito servir de suporte às escolhas feitas para o desenvolvimento da Base de Dados da Ourbook, assim como, a análise e avaliação prévia do sistema como um todo.

A Ourbook é uma aplicação fictícia que consiste numa rede social dedicada aos amantes de livros, com a capacidade de fazer user-matching e fornecer uma infraestrutura para utilizadores trocarem livros entre si. Após dois utilizadores estabelecerem uma ligação, passam a ser considerados “Bookmates” dentro do sistema, disponibilizando assim a opção de troca de livros. A troca é assegurada por uma entidade externa que intitulamos de estafeta.

Para o desenvolvimento da base de dados, utilizámos MySQL e o Microsoft SQL Server Management Studio, enquanto que para a interface gráfica, utilizámos o Visual Studio e a linguagem de programação Visual Basic.

Importante: para acederem à nossa Base de Dados precisarão das nossas credenciais, nomeadamente:

- Username: p5g1
- Password: AndreGoncaloRumoao20

Para apontarem a conexão para as vossas credenciais, terão que em cada formulário editar a “connection string” no método nativo Load de cada formulário.

Análise de Requisitos

Para iniciarmos o processo da "Análise de Requisitos", mapeamos quais seriam os atores do sistema.

Atores

Ator	Papel no Sistema
Utilizador	Indivíduo que tem uma conta Ourbook.
Estafeta DPD	Indivíduo que assegura a troca física dos livros

Tabela 1 - Atores

De seguida, procurámos listar as funcionalidades que os atores poderiam esperar ao interagir com o sistema.

Requisitos Funcionais

Ator	Caso de Utilização
Utilizador	<ul style="list-style-type: none"> • Depositar Livro(s) • Navegar pelo feed • Criar uma comunidade • Procurar BookMate • Trocar Livro • Utilizar o chat OurBook • Escrever uma avaliação • Adicionar um Utilizador • Reagir a Publicações • Customizar o Perfil • Realizar uma Publicação • Atribuir uma Gorjeta
Estafeta DPD	<ul style="list-style-type: none"> • Validar o QR Code OurBook • Ler Avaliações • Aceitar Gorjetas • Ler notificações de entrega

Tabela 2 - Requisitos Funcionais

Após termos obtido os requisitos funcionais, achámos necessário listar conceitos e entidades, assim como o que representam, para facilitar a construção dos diagramas.

Entidades do Sistema

Entidade	Descrição
Utilizador	Indivíduo que tem uma conta OurBook
Avaliação BookMate	Avaliação escrita por um Utilizador acerca da qualidade do seu BookMate depois de realizar uma troca de livros.
Perfil	O Perfil é a presença observável do Utilizador na plataforma.
FormulárioDepósito	Formulário que é preenchido pelo Utilizador aquando do processo “Depositar Livro”.
Livro	O Livro é o objeto que é trocado no processo de “Trocar Livro”.
Comunidade	A Comunidade é um espaço disponibilizado pela plataforma para agrupar Utilizadores com gostos semelhantes de forma a poderem partilhar Livros, artigos e Publicações entre si.
FormulárioTroca	Formulário que é preenchido por ambos os BookMates com as informações da troca de Livros.
Troca	Corresponde ao registo da Troca do processo “Troca Livro”, que tem um QR Code, cuja validação é assegurada pelo EstafetaDPD e contém a informação do FormulárioTroca.
EstafetaDPD	Indivíduo contratado pela DPD para realizar a troca dos Livros ao domicílio dos Utilizadores.
AvaliaçãoEstafeta	Avaliação escrita pelo Utilizador acerca da qualidade do serviço prestado pelo EstafetaDPD.
Gorjeta	Quantia monetária que o Utilizador atribui ao EstafetaDPD para o recompensar pelo serviço de qualidade prestado.

Tabela 3 - Entidades do Sistema

Diagrama Entidade-Relacionamento (DER)

A construção do DER requereu duas iterações de desenvolvimento, nomeadamente no início do projeto, antes do dia 22 de abril e após o feedback do professor Tiago Godinho.

1ª versão do DER:

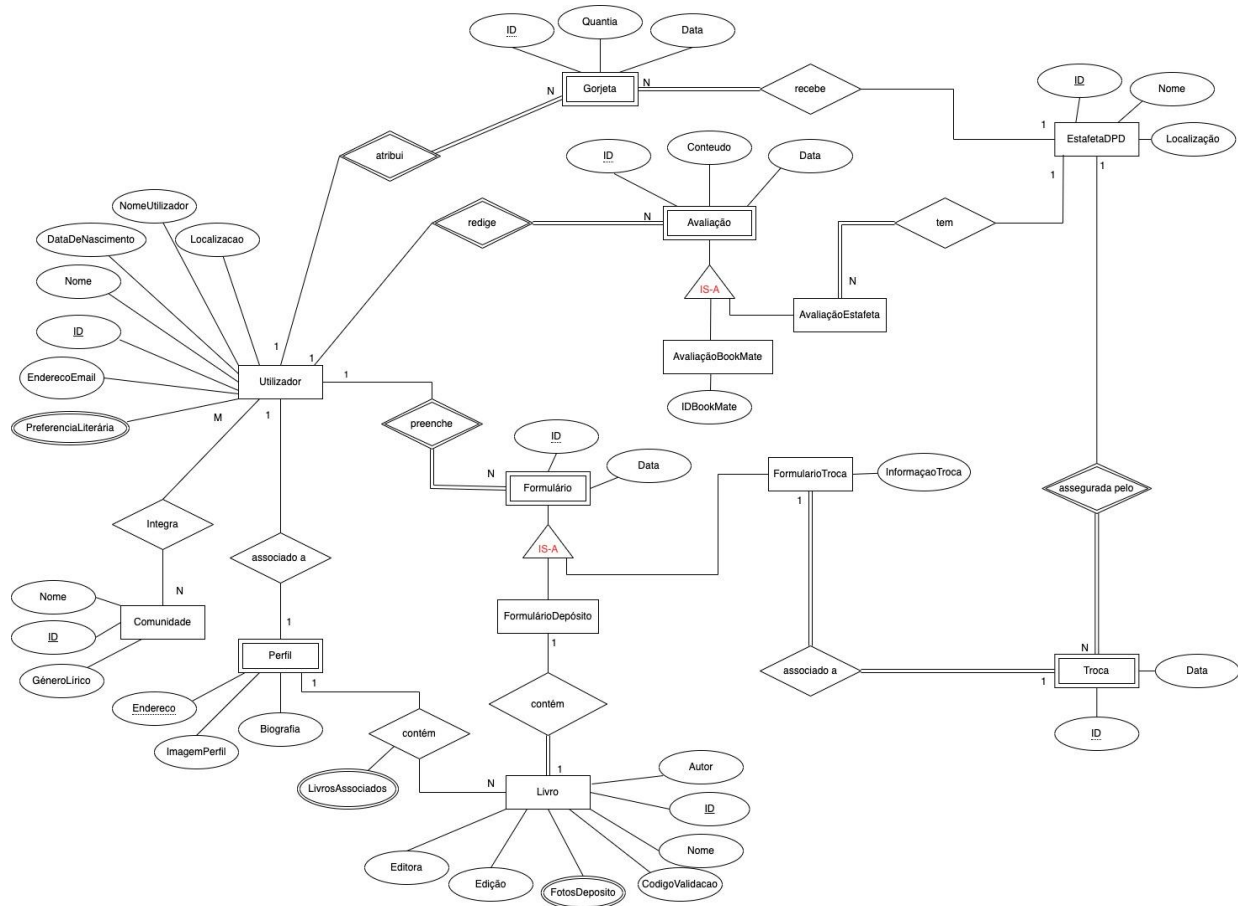


Imagem 1 - 1ª versão do Diagrama Entidade-Relacionamento

Esta versão carecia de termos fulcrais para a interpretação do mesmo e tinha demasiadas entidades desnecessárias. As ligações entre algumas entidades, nomeadamente, Utilizador-Perfil, Livro-Perfil estavam incorretas e faltavam ligações em relações do tipo “two-end-users”, ou seja, que envolviam duas instâncias de utilizadores.

Após a correção dos problemas apontados pelo professor Tiago Godinho, obtivemos a 2ª versão do Diagrama Entidade-Relacionamento:

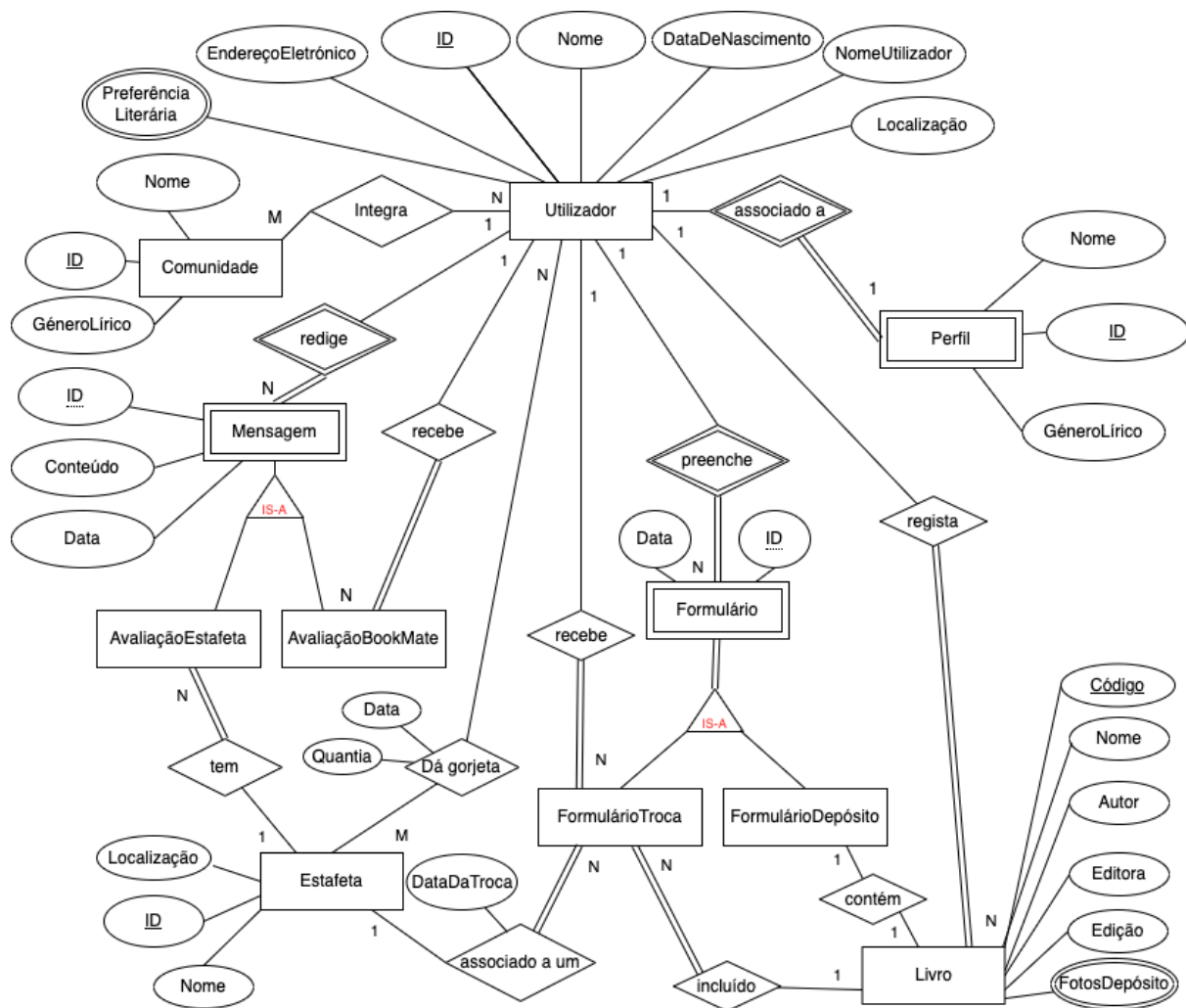


Imagem 2 - 2ª versão do Diagrama Entidade-Relacionamento

O DER que utilizámos como referencia para a construção da base de dados foi o que se encontra na Imagem 2, ou seja, a 2ª versão (melhorada e atualizada) do Diagrama Entidade-Relacionamento.

Após termos o DER definido, traduzímo-lo para um Esquema Relacional que iria-se aproximar da estrutura das tabelas na Base de Dados:

Esquema Relacional

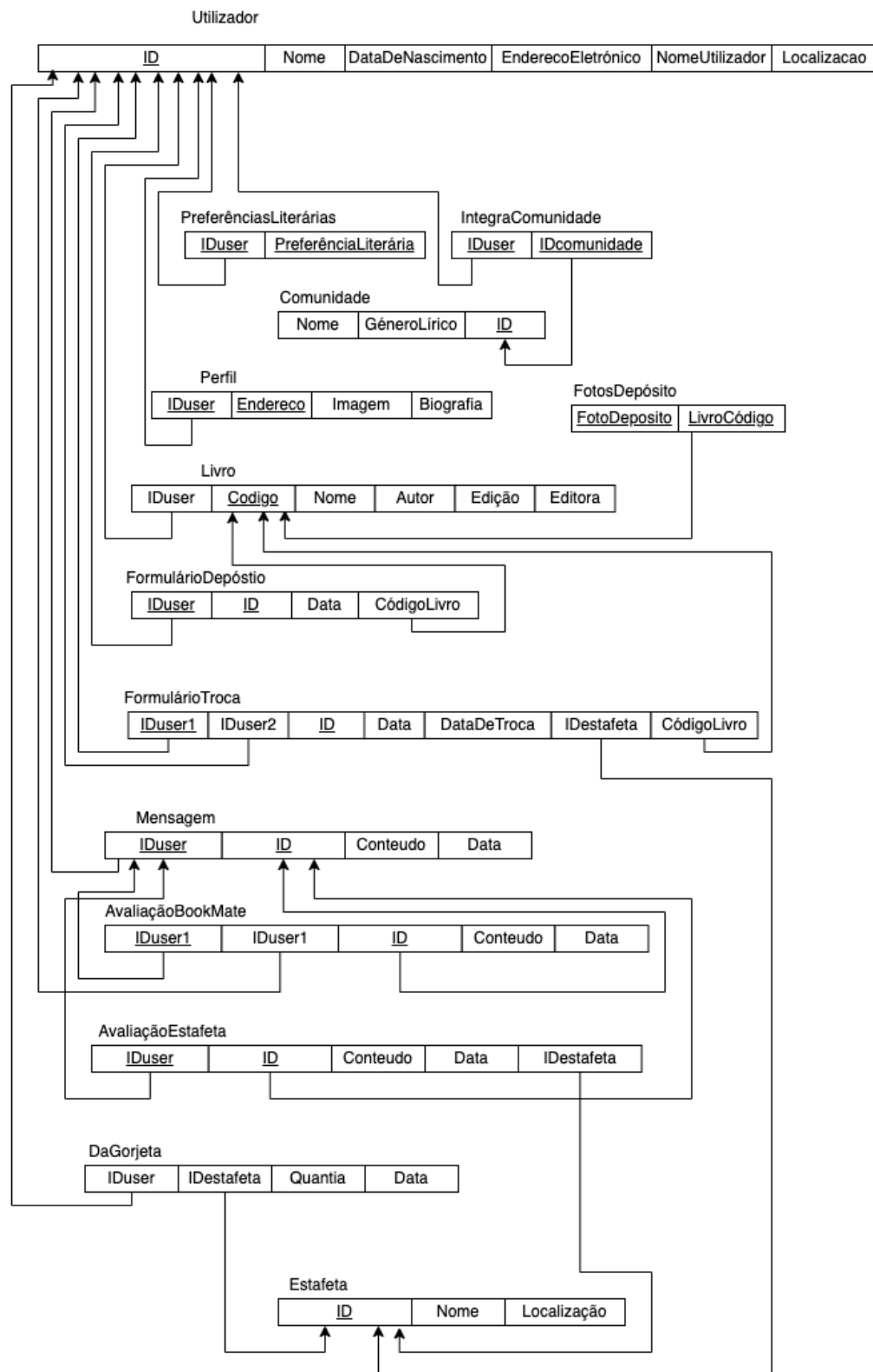
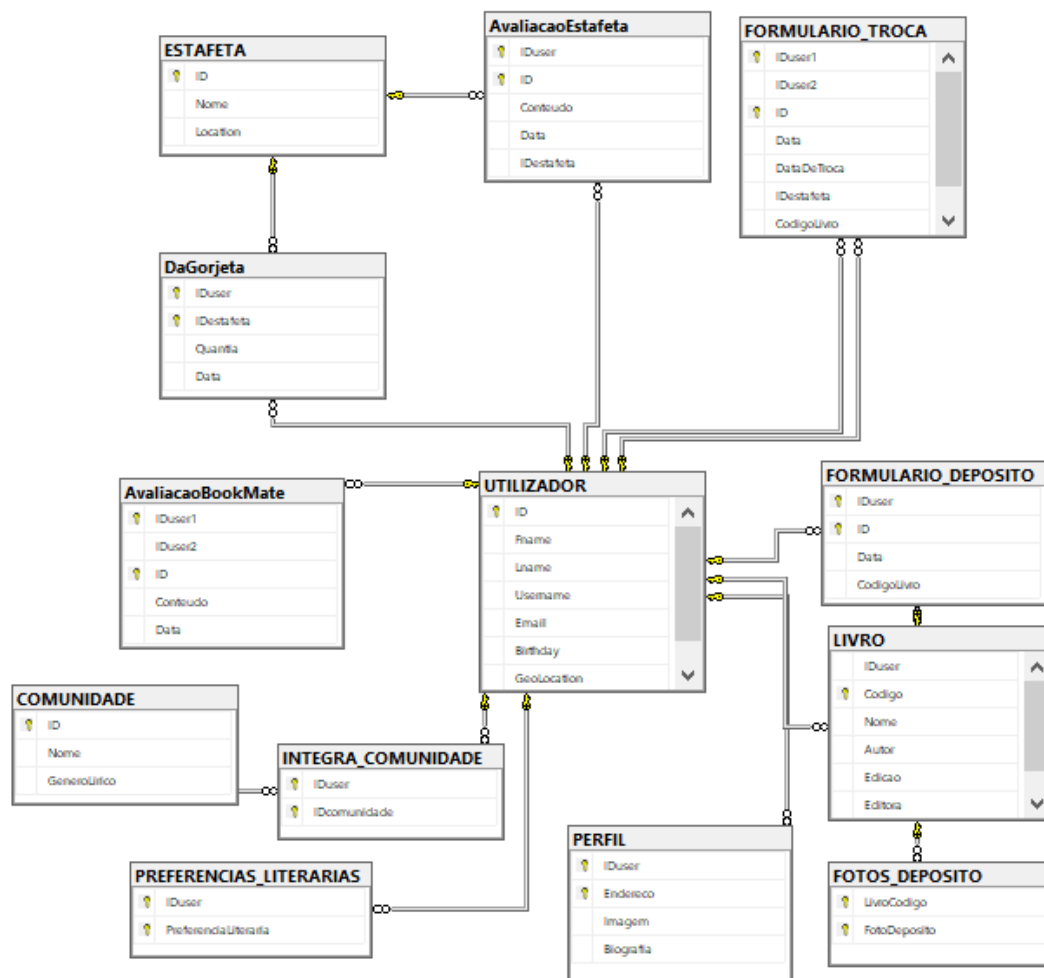


Imagem 3 - 2ª versão do Esquema Relacional

Estrutura da Base de Dados no SQL Server

Após a criação da query criacional da base de dados, obtivemos o seguinte diagrama:



Foram 13 as tabelas criadas através de comandos SQL, sendo que as principais remetiam para a do Utilizador, Livro e os Formulários:

SQL DDL Tabela Utilizador

```

CREATE TABLE UTILIZADOR (
    ID INT NOT NULL,
    Fname VARCHAR(30) NOT NULL,
    Lname VARCHAR(30) NOT NULL,
    Username VARCHAR(30) NOT NULL,
    Email VARCHAR(30) NOT NULL,
    Birthday DATE NOT NULL,
    GeoLocation VARCHAR(50),
    PRIMARY KEY(ID),
    UNIQUE(ID, Username)
)
  
```

SQL DDL Tabela Livro

```
CREATE TABLE LIVRO (  
    IDuser INT NOT NULL,  
    Codigo INT NOT NULL,  
    Nome VARCHAR(100) NOT NULL,  
    Autor VARCHAR(100) NOT NULL,  
    Edicao INT NOT NULL,  
    Editora VARCHAR(100) NOT NULL  
    PRIMARY KEY(Codigo),  
    UNIQUE(Codigo),  
    FOREIGN KEY(IDuser) REFERENCES UTILIZADOR(ID)  
)
```

SQL DDL Tabela Formulário Troca

```
CREATE TABLE FORMULARIO_TROCA (  
    IDuser1 INT NOT NULL,  
    IDuser2 INT NOT NULL,  
    ID INT NOT NULL,  
    Data DATE NOT NULL,  
    DataDeTroca DATE NOT NULL,  
    IDestafeta INT NOT NULL,  
    CodigoLivro INT NOT NULL,  
    PRIMARY KEY(IDuser1, ID),  
    FOREIGN KEY (IDuser1) REFERENCES UTILIZADOR(ID),  
    FOREIGN KEY (IDuser2) REFERENCES UTILIZADOR(ID)  
)
```

SQL DDL Tabela Formulário Depósito

```
CREATE TABLE FORMULARIO_DEPOSITO (  
    IDuser INT NOT NULL,  
    ID INT NOT NULL,  
    Data DATE NOT NULL,  
    CodigoLivro INT NOT NULL,  
    PRIMARY KEY(IDuser, ID),  
    FOREIGN KEY(IDuser) REFERENCES UTILIZADOR(ID),  
    FOREIGN KEY(CodigoLivro) REFERENCES LIVRO(Codigo)  
)
```

Relação Tabelas da Base de Dados - Formulários Gráficos

User Form

Tabelas acedidas: Utilizador

Ações: Inserção, Atualização e Remoção

BookPage Form

Tabelas acedidas: LIVRO

Ações: Inserção, Atualização e Remoção

Profile Form

Tabelas acedidas: FORMULÁRIO DEPÓSITO, FORMULÁRIO TROCA, UTILIZADOR

Ações: UserStatistics Procedure

Courier Form

Tabelas acedidas: ESTAFETA

Ações: Inserção, Atualização e Remoção

BookMate Form

Tabelas acedidas: UTILIZADOR, LIVRO, PREFERENCIAS_LITERARIAS, FORMULARIO_TROCA

Ações: Inserção, Atualização e Remoção

Índices - Base de Dados

Apesar de não termos conseguido testar a influência dos índices criados na pesquisa da nossa base de dados, achámos pertinente defini-los para assegurar a escalabilidade futura do sistema.

Os índices criados foram:

- idxUtilizador

```
CREATE INDEX idxUtilizador ON UTILIZADOR(Fname, Lname);
```

- idxLivro

```
CREATE INDEX idxLivro ON LIVRO(Nome, Autor);
```

- idxAvaliacaoBookMate

```
CREATE INDEX idxAvaliacaoBookMate ON AvaliacaoBookMate(IDuser1, IDuser2);
```

- idxAvaliacaoEstafeta

```
CREATE INDEX idxAvaliacaoEstafeta ON AvaliacaoEstafeta(IDuser, IDestafeta);
```

Stored Procedures - Base de Dados

Nome	Propósito
UserStatistics	Utilizada para obter os número de livros já depositados e número de trocas já efetuadas pelo Utilizador
RemoveBook	Remove uma linha da tabela LIVRO consoante o ID do Livro passado ou remove todos os livros associados a um ID de um utilizador.
RemoveUser	Remove uma linha da tabela do UTILIZADOR, removendo também todas as suas dependências nas outras tabelas.
GetUserList	Retorna a lista dos utilizadores.
GetMaxUserID	Retorna o ID máximo da tabela UTILIZADOR
CreateNewUser	Cria e adiciona uma nova linha à tabela do UTILIZADOR
UpdateUser	Atualiza as informações de um dado Utilizador
GetEstafetaList	Retorna a lista dos estafetas
GetMaxEstafetaID	Retorna o ID máximo da tabela ESTAFETA
CreateNewEstafeta	Cria e adiciona uma nova linha à tabela do ESTAFETA
UpdateEstafeta	Atualiza as informações de um dado Estafeta
RemoveEstafeta	Remove uma linha da tabela do ESTAFETA, eliminando todas as suas dependências de outras tabelas
CreateNewBook	Cria e adiciona uma nova linha à tabela do LIVRO
GetMaxLivroCode	Retorna o maior código da tabela LIVRO
GetBookList	Retorna a lista dos livros de um utilizador
CreateNewFormularioDeposito	Cria e adiciona uma nova linha à tabela do FORMULARIO_DEPOSITO
CreateNewFormularioTroca	Cria e adiciona uma nova linha à tabela do FORMULARIO_TROCA
GetMaxFormID	Retorna o maior valor de ID do FORMULARIO_TROCA

Triggers - Base de Dados

Criámos o trigger “bookDepositted” que é ativado quando um novo livro é adicionado à tabela LIVRO. O trigger adiciona automaticamente o novo livro à tabela FORMULARIO_DEPOSITO.

Conclusão

Este trabalho permitiu-nos consolidar o conhecimento adquirido nas aulas teórico-práticas e práticas ao longo do semestre, tendo-nos ensinado o protocolo e o procedimento a tomar e seguir aquando da criação de uma base de dados para um sistema, revelando a extrema importância da formulação de um desenho conceptual da base de dados e, investindo tempo na interpretação, filtração e validação das relações de forma a mapear e definir relações reais que façam sentido e que sejam adequadas para os dados do sistema.