



Universidade do Minho
Licenciatura em Ciências da Computação

Unidade Curricular de Bases de Dados

Ano Lectivo de 2021/2022

MUMU rede de livrarias Ida. Parte 2

**André Barbosa (A91684), Francisco Paulino
(A91666), Hugo Nogueira (A91686), Rafael
Carvalho(A91642)**

Janeiro 2022

BD

Data de Recepção	
Responsável	
Avaliação	
Observações	

MUMU rede de livrarias Ida. Parte 2

André Barbosa (A91684), Francisco Paulino (A91666), Hugo Nogueira (A91686), Rafael Carvalho(A91642)

Janeiro 2022

Resumo

Este relatório tem como objetivo dar continuidade ao trabalho desenvolvido anteriormente, referente a uma base de dados para uma rede de livrarias.

Serão abordados temas como a normalização de dados do modelo lógico apresentado pelo grupo de trabalho, (até à terceira forma normal), bem como a Indexação do Sistema de Dados e os Procedimentos implementados.

Seguidamente será apresentada a conceção e implementação de um sistema de Dados em MongoDB, que consiste da migração da base de dados relacional, referente ao trabalho anterior, para uma base de dados NoSQL.

Para tal será definido inicialmente o Esquema da Base de Dados do novo sistema, seguido da criação da base de dados e das coleções necessárias e do devido processo de migração, dos dados do sistema relacional para o sistema não relacional. Serão também explorados os dados em MongoDB, através de interrogações elaboradas.

Após o desenvolvimento destas etapas será apresentado um ponto de vista geral, por parte do grupo, de modo a denotar as vantagens e as desvantagens do projeto realizado, bem como possíveis sugestões de otimização.

Área de Aplicação: Desenvolvimento e implementação de Sistemas de Bases de Dados, Migração de Sistemas de Base de Dados, Arquitetura de Bases de Dados.

Palavras-Chave: Normalização, Indexação, Base de Dados Relacional, Base de NoSQL, Base de dados orientada a documentos, MongoDB.

Índice

Resumo	i
Índice	ii
Índice de Figuras	iii
1. Sistema relacional implementado	1
1.1. Apresentação do sistema	1
1.2. Normalização de dados	2
1.3. Indexação do sistema de Dados	5
1.4. Procedimentos Implementados	5
2. Conceção e Implementação de um Sistema de Dados em MongoDB	8
2.1. Definição do esquema da Base de Dados	8
2.2. Criação da Base de Dados e das Coleções	9
2.3. O Processo de Migração de Dados	10
2.4. Exploração de Dados em MongoDB	11
3. Conclusões e Trabalho Futuro	12
Referências	13
Lista de Siglas e Acrónimos	14

Índice de Figuras

Figure 1-Modelo Relacional	2
Figure 2-Tabela LivrariaFuncionarioTurno	4
Figure 3- Tabela Livraria	4
Figure 4-Tabela Funcionário	4
Figure 5-Tabela Turno	5
Figure 6-Código de criação de índices	5
Figure 7-Criação do Procedimento saldo por nif	6
Figure 8-Código para usar o Procedimento definido anteriormente	6
Figure 9-Criação do Procedimento clientes que mais gastam	6
Figure 10-Criação do Procedimento contagem de vendas num dia	6
Figure 11-Criação do Procedimento funcionarios em livraria	7
Figure 12-Criação do Procedimento que autores um sócio leu	7
Figure 13-Modelo conceptual	8
Figure 14-Coleções da base de Dados	9
Figure 15-Documentos da coleção Funcionário	10
Figure 16-Código de procura de cliente usando o seu NIF	11
Figure 17-Código de procura dos clientes que mais gastam	11
Figure 18-Código de procura das vendas que foram efetuadas num dia	11
Figure 19-Código de procura dos funcionários em cada livraria	11

1. Sistema relacional implementado

1.1. Apresentação do sistema

O sistema que vamos utilizar como base para as propostas da segunda parte tem como base o sistema desenvolvido na primeira parte do trabalho, uma rede de livrarias.

A figura abaixo representa o modelo lógico do sistema de base de dados relacional proposto para as livrarias, que foca em três grupos: área de gestão da Livraria, Consumidor e o Staff.

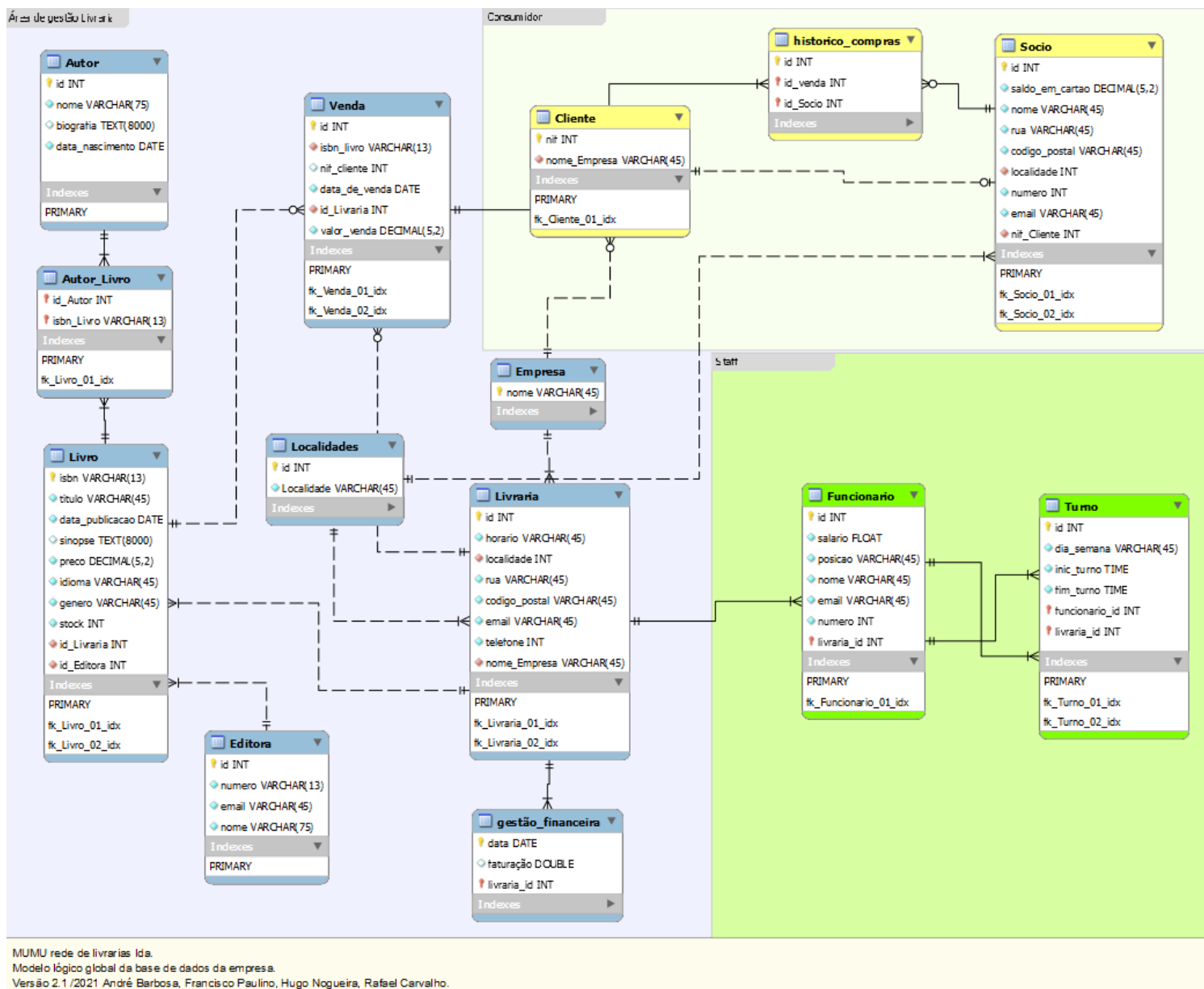


Figure 1-Modelo Relacional

1.2. Normalização de dados

O processo de normalização é uma técnica formal de verificação das diversas tabelas base de um esquema de uma base de dados, que se baseia na análise das chaves primárias e das dependências funcionais de todos os seus atributos. É um processo progressivo, que assenta na execução de uma série de etapas, cada uma delas correspondendo a uma forma normal específica, que vai transformando o esquema de uma base de dados sucessivamente em direção a um estado mais robusto e menos vulnerável.

A primeira forma normal:

Uma relação está na primeira forma normal (1FN) se todos os valores dos seus atributos forem atômicos, não contendo grupos de dados repetidos.

- Um atributo atômico é aquele que não pode ser decomposto.
- Um grupo repetitivo é um conjunto de um ou mais atributos com vários valores que estão relacionados.
- Na prática, podemos dizer que uma relação na qual as interseções entre colunas (atributos) e linhas (registros) apenas contêm um único valor – um valor atômico – estão na 1FN.

Requisitos da 1FN:

Os requisitos para que uma dada tabela esteja na 1FN são os seguintes:

- Cada tabela tem uma chave primária.
- Os valores de cada um dos atributos de uma tabela são todos atributos atômicos.
- Não existem grupos de dados repetidos

Segunda forma normal:

Uma relação está na segunda forma normal (2FN) se todos os seus atributos não-primos (que não fazem parte de uma chave primária) forem totalmente dependentes da sua chave primária.

Terceira forma normal:

Uma tabela está na 3FN se todos os atributos que não sejam chaves primárias sejam mutuamente independentes, não havendo assim dependências funcionais transitivas.

Requisitos da 3FN:

Os requisitos para que uma tabela esteja na 3FN são os seguintes:

- Todos os requisitos da 1FN e da 2FN devem-se verificar.
- Os atributos que não dependam da chave primária devem ser “eliminados”:
- qualquer atributo que é dependente não só da chave primária como também de um outro qualquer atributo, deve ser transferido para outra tabela.
- Não devem existir dependências transitivas.

Exemplo de aplicação das três formas normais apresentadas anteriormente:

livraria_id	horario	localidade	codigo_postal	email	telefone	funcionario_id	salario	posicao	nome	email	numero	turno_id	dia_semana	dia_semana	inic_turno	fim_turno
1	09:00-22:00	1	4700-093	mumu@gmail.com	253253253	1	900	Gerente	Rosalino Matos	rosas@gmail.com	915868749	1	Segunda	Segunda	09:00:00	17:00:00
1	09:00-22:00	1	4700-093	mumu@gmail.com	253253253	1	900	Gerente	Rosalino Matos	rosas@gmail.com	915868749	2	Terça	Terça	09:00:00	17:00:00
1	09:00-22:00	1	4700-093	mumu@gmail.com	253253253	1	900	Gerente	Rosalino Matos	rosas@gmail.com	915868749	2	Terça	Terça	17:00:00	22:00:00
1	09:00-22:00	1	4700-093	mumu@gmail.com	253253253	1	900	Gerente	Rosalino Matos	rosas@gmail.com	915868749	3	Quarta	Quarta	09:00:00	17:00:00
1	09:00-22:00	1	4700-093	mumu@gmail.com	253253253	1	900	Gerente	Rosalino Matos	rosas@gmail.com	915868749	4	Quinta	Quinta	09:00:00	17:00:00
1	09:00-22:00	1	4700-093	mumu@gmail.com	253253253	1	900	Gerente	Rosalino Matos	rosas@gmail.com	915868749	5	Sexta	Sexta	09:00:00	17:00:00
2	09:00-22:00	4	4700-198	mumu@gmail.com	253253254	1	900	Gerente	Carlota Cambalhota	carla@gmail.com	915666749	1	Segunda	Segunda	09:00:00	17:00:00
2	09:00-22:00	4	4700-198	mumu@gmail.com	253253254	1	900	Gerente	Carlota Cambalhota	carla@gmail.com	915666749	1	Segunda	Segunda	17:00:00	22:00:00
2	09:00-22:00	4	4700-198	mumu@gmail.com	253253254	1	900	Gerente	Carlota Cambalhota	carla@gmail.com	915666749	2	Terça	Terça	09:00:00	17:00:00
2	09:00-22:00	4	4700-198	mumu@gmail.com	253253254	1	900	Gerente	Carlota Cambalhota	carla@gmail.com	915666749	2	Terça	Terça	17:00:00	22:00:00
3	09:00-22:00	5	1600-093	mumu@gmail.com	253253255	1	900	Gerente	Filipe Bastos	filipe@gmail.com	965855749	1	Segunda	Segunda	09:00:00	17:00:00
3	09:00-22:00	5	1600-093	mumu@gmail.com	253253255	1	900	Gerente	Filipe Bastos	filipe@gmail.com	965855749	1	Segunda	Segunda	17:00:00	22:00:00
3	09:00-22:00	5	1600-093	mumu@gmail.com	253253255	1	900	Gerente	Filipe Bastos	filipe@gmail.com	965855749	2	Terça	Terça	09:00:00	17:00:00
3	09:00-22:00	5	1600-093	mumu@gmail.com	253253255	1	900	Gerente	Filipe Bastos	filipe@gmail.com	965855749	2	Terça	Terça	17:00:00	22:00:00
1	09:00-22:00	1	4700-093	mumu@gmail.com	253253253	2	700	Full-timer	Ana Matos	aninha@gmail.com	914848449	1	Segunda	Segunda	09:00:00	17:00:00
1	09:00-22:00	1	4700-093	mumu@gmail.com	253253253	2	700	Full-timer	Ana Matos	aninha@gmail.com	914848449	1	Segunda	Segunda	17:00:00	22:00:00
1	09:00-22:00	1	4700-093	mumu@gmail.com	253253253	2	700	Full-timer	Ana Matos	aninha@gmail.com	914848449	2	Terça	Terça	09:00:00	17:00:00
1	09:00-22:00	1	4700-093	mumu@gmail.com	253253253	2	700	Full-timer	Ana Matos	aninha@gmail.com	914848449	2	Terça	Terça	17:00:00	22:00:00
1	09:00-22:00	1	4700-093	mumu@gmail.com	253253253	2	700	Full-timer	Ana Matos	aninha@gmail.com	914848449	3	Quarta	Quarta	09:00:00	17:00:00
1	09:00-22:00	1	4700-093	mumu@gmail.com	253253253	2	700	Full-timer	Ana Matos	aninha@gmail.com	914848449	4	Quinta	Quinta	09:00:00	17:00:00
1	09:00-22:00	1	4700-093	mumu@gmail.com	253253253	2	700	Full-timer	Ana Matos	aninha@gmail.com	914848449	5	Sexta	Sexta	09:00:00	17:00:00

Figure 2-Tabela LivrariaFuncionarioTurno

id	horario	localidade	rua	codigo_postal	email	telefone
1	09:00-22:00	1	Rua das Sardinhas	4700-093	mumu@gmail.com	253253253
2	09:00-22:00	4	Rua Santo António	4700-198	mumu@gmail.com	253253254
3	09:00-22:00	5	Rua Benfica	1600-093	mumu@gmail.com	253253255

Figure 3- Tabela Livraria

id	livraria_id	salario	posicao	nome	email	numero
1	1	900	Gerente	Rosalino Matos	rosas@gmail.com	915868749
1	2	900	Gerente	Carlota Cambalhota	carla@gmail.com	915666749
1	3	900	Gerente	Filipe Bastos	filipe@gmail.com	965855749
2	1	700	Full-timer	Ana Matos	aninha@gmail.com	914848449
2	2	700	Full-timer	Mixi Mitsubishi	mitsu@gmail.com	927768749
2	3	700	Full-timer	Marta Carta	marta@gmail.com	915859749

Figure 4-Tabela Funcionário

Ao longo do nosso trabalho é possível verificar que as formas normais foram respeitadas, estando o nosso trabalho normalizado.

id	funcionario_id	livraria_id	dia_semana	inic_turno	fim_turno
1	1	1	Segunda	09:00:00	17:00:00
1	1	2	Segunda	09:00:00	17:00:00
1	1	3	Segunda	09:00:00	17:00:00
1	2	1	Segunda	17:00:00	22:00:00
1	2	2	Segunda	17:00:00	22:00:00
1	2	3	Segunda	17:00:00	22:00:00
2	1	1	Terça	09:00:00	17:00:00
2	1	2	Terça	09:00:00	17:00:00
2	1	3	Terça	09:00:00	17:00:00
2	2	1	Terça	17:00:00	22:00:00
2	2	2	Terça	17:00:00	22:00:00
2	2	3	Terça	17:00:00	22:00:00
3	1	1	Quarta	09:00:00	17:00:00
4	1	1	Quinta	09:00:00	17:00:00
5	1	1	Sexta	09:00:00	17:00:00

Figure 5-Tabela Turno

1.3. Indexação do sistema de Dados

Os índices são usados para a recuperação de dados de uma base de dados de forma mais eficiente, melhorando o desempenho das *queries*.

A criação de um índice envolve o uso da instrução CREATE INDEX, que permite nomear o índice, especificar a tabela e qual coluna ou colunas a indexar. Esta apresenta como vantagem a aceleração das operações SELECT e WHERE, provocando, contudo, o abrandamento das operações de INSERT e UPDATE. Assim criaremos índices para colunas de maior consulta e pouca atualização, tais como os ISBN dos livros e NIFS dos clientes.

```
1 CREATE INDEX IDX_01 ON livro(isbn);
2 CREATE INDEX IDX_02 ON cliente(nif);
```

Figure 6-Código de criação de índices

1.4. Procedimentos Implementados

De forma a facilitar a consulta da nossa base de dados implementamos alguns procedimentos.

Quando quisermos procurar o saldo em cartão de um dado sócio usando o seu NIF usamos o procedimento “saldo_por_nif” que foi definido como:

```
DELIMITER $$
CREATE PROCEDURE saldo_por_nif (IN nif int)
BEGIN
SELECT saldo_em_cartao From Socio
WHERE nif_Cliente=nif ;
END
$$
```

Figure 7-Criação do Procedimento saldo por nif

Basta usar o comando CALL seguido do nome do procedimento e o NIF que queremos usar.

```
CALL saldo_por_nif(282381382);
```

Figure 8-Código para usar o Procedimento definido anteriormente

O resto dos procedimentos são feitos de forma análoga.

Procura dos clientes que mais gastam:

```
DELIMITER $$
CREATE PROCEDURE clientes_que_mais_gastam()
BEGIN
SELECT nif_cliente,sum(valor_venda) as gasto_total FROM Venda
GROUP BY nif_cliente
ORDER BY sum(valor_venda) DESC;
END
$$
```

Figure 9-Criação do Procedimento clientes que mais gastam

Contagem de vendas efetuadas num dia:

```
DELIMITER $$
CREATE PROCEDURE conta_vendas_dia( IN data1 DATE)
BEGIN
SELECT COUNT(data_de_venda) FROM Venda
WHERE data_de_venda= data1;
END
$$
```

Figure 10-Criação do Procedimento contagem de vendas num dia

Que funcionários há em cada livraria:

```
DELIMITER $$
CREATE PROCEDURE funcionarios_livraria ( )
BEGIN
SELECT Funcionario.nome,Livraria.id as livraria FROM Funcionario
INNER JOIN Livraria
ON Funcionario.livraria_id=Livraria.id;
END
$$
```

Figure 11-Criação do Procedimento funcionarios em livraria

Que autores é que um sócio leu:

```
DELIMITER $$
CREATE PROCEDURE autores_socio( IN nif int)
BEGIN
SELECT Socio.nif_cliente,Venda.isbn_livro,Autor.nome
From Socio
INNER JOIN historico_compras
ON Socio.id=historico_compras.id_Socio
INNER JOIN Venda
ON historico_compras.id_venda=Venda.id
INNER JOIN Autor_livro
ON Venda.isbn_livro=Autor_livro.isbn_Livro
INNER JOIN Autor
ON Autor_livro.id_Autor=Autor.id
WHERE Socio.nif_cliente=nif;
END
$$
```

Figure 12-Criação do Procedimento que autores um sócio leu

2. Conceção e Implementação de um Sistema de Dados em MongoDB

2.1. Definição do esquema da Base de Dados

O esquema da base de dados que vai ser usado para a implementação da mesma é o esquema conceptual que foi usado para a parte 1 do trabalho, no entanto, visto que este foi criado a pensar numa base de dados SQL, não regerá por completo a base de dados em MongoDB, mas desempenhará um papel de orientação para o grupo de trabalho.

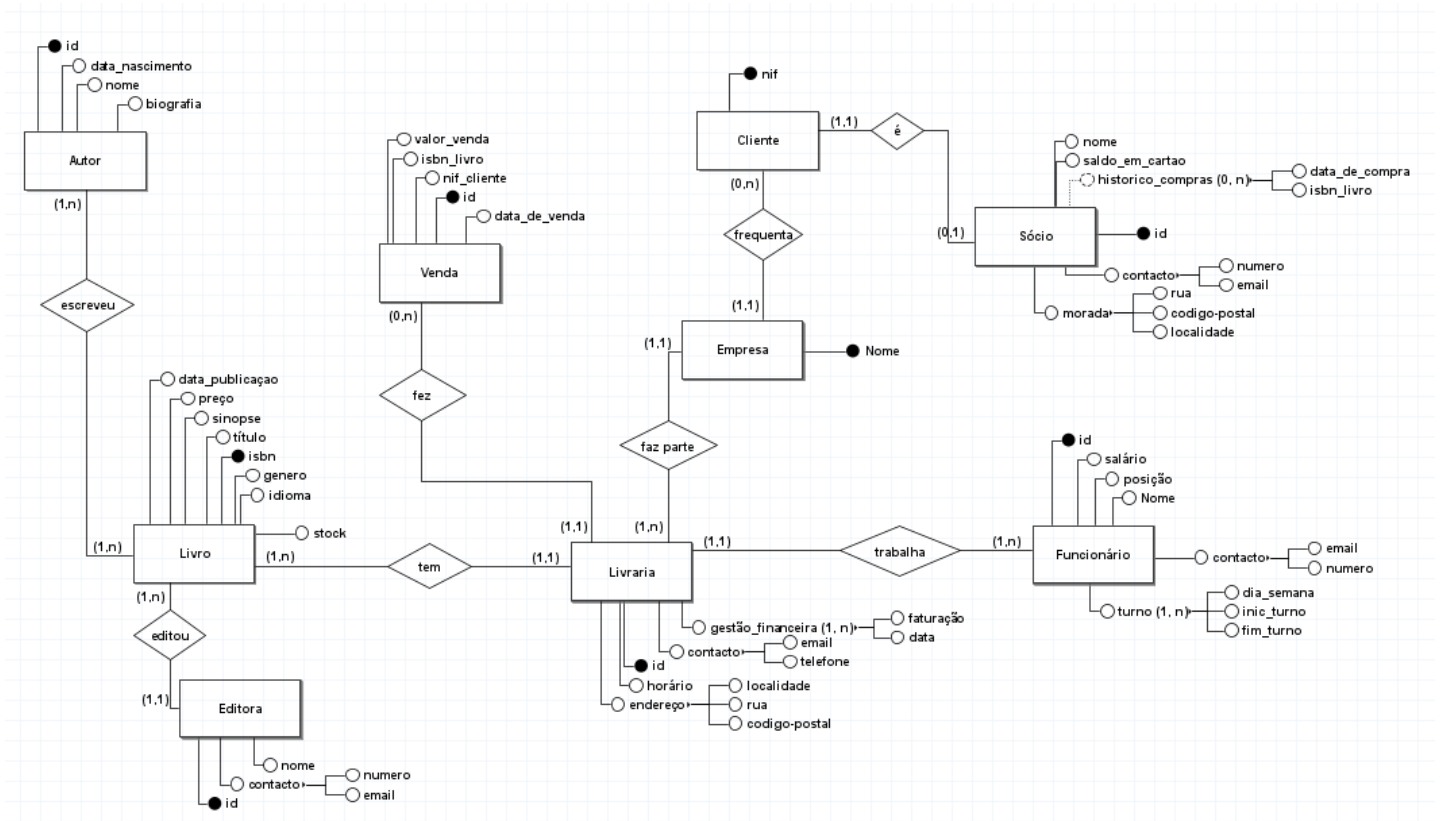


Figure 13-Modelo conceptual

2.2. Criação da Base de Dados e das Coleções

A criação da base de dados, embora se baseie no modelo conceptual de uma base de dados relacional, vai apenas aproveitar alguns aspetos deste.

As entidades que no modelo relacional iriam servir de tabelas, agora vão ser coleções, nas quais vão haver documentos, e como uma base de dados orientada a documentos não está presa a uma forma específica, os atributos definidos servem apenas como uma ideia do que cada coleção irá conter.

Exemplo de algumas coleções criadas:

cliente				
Storage size: 20.48 kB	Documents: 5	Avg. document size: 46.00 B	Indexes: 1	Total index size: 20.48 kB
editora				
Storage size: 20.48 kB	Documents: 3	Avg. document size: 99.00 B	Indexes: 1	Total index size: 20.48 kB
empresa				
Storage size: 20.48 kB	Documents: 1	Avg. document size: 34.00 B	Indexes: 1	Total index size: 20.48 kB
funcionario				
Storage size: 20.48 kB	Documents: 6	Avg. document size: 168.00 B	Indexes: 1	Total index size: 20.48 kB

Figure 14-Coleções da base de Dados

As quais depois do processo de migração de dados vão conter os documentos com a informação que previamente estava nas tabelas da base de dados relacional.

```
> _id: Object
  id: 1
  livraria_id: 1
  salario: 900
  posicao: "Gerente"
  nome: "Rosalino Matos"
  email: "rosas@gmail.com"
  numero: 915868749
```

```
> _id: Object
  id: 1
  livraria_id: 2
  salario: 900
  posicao: "Gerente"
  nome: "Carlota Cambalhota"
  email: "carla@gmail.com"
  numero: 915866749
```

```
> _id: Object
  id: 1
  livraria_id: 3
  salario: 900
  posicao: "Gerente"
  nome: "Filipe Bastos"
  email: "filipe@gmail.com"
  numero: 965855749
```

```
> _id: Object
  id: 2
  livraria_id: 1
  salario: 700
  posicao: "Full-timer"
  nome: "Ana Matos"
  email: "aninha@gmail.com"
  numero: 914848449
```

Figure 15-Documentos da coleção Funcionário

2.3. O Processo de Migração de Dados

O processo de migração foi feito com ajuda do programa *Studio 3T*, no qual é apenas necessário selecionar a base de dados SQL origem e a base de dados MongoDB destino,

entre as quais pretendemos fazer migração de dados. O processo de migração é simples e é feito automaticamente. Este processo consiste em passar as tabelas do modelo relacional a *collections* do modelo orientado a documentos, e os documentos de cada collection vão ser os dados que estavam nas tabelas em SQL.

2.4. Exploração de Dados em MongoDB

Na exploração de dados vamos usar os exemplos que definimos anteriormente.

Procura do saldo em cartão de um cliente usando o seu NIF:

```
> db.socio.aggregate([{$match:{nif_Cliente:282381382}},{$group:{_id:"$nome",saldo:{$first:"$saldo_em_cartao"}}}])
< { _id: 'Susana Rita', saldo: Decimal128("3.00") }
```

Figure 16-Código de procura de cliente usando o seu NIF

Procura dos clientes que mais gastam:

```
> db.venda.aggregate([{$group:{_id:"$nif_cliente",total:{$sum:"$valor_venda"}} },{$sort:{total:-1}}])
< { _id: 282381382, total: Decimal128("51.49") }
  { _id: 259842352, total: Decimal128("41.40") }
  { _id: 277777412, total: Decimal128("31.50") }
```

Figure 17-Código de procura dos clientes que mais gastam

Quantas vendas foram efetuadas num determinado dia:

```
> db.venda.count({data_de_venda:ISODate('2021-12-01T00:00:00.000+00:00')})
< 5
```

Figure 18-Código de procura das vendas que foram efetuadas num dia

Que funcionários há em cada livraria:

```
> db.funcionario.aggregate([{$lookup:{from:"livraria",localField:"livraria_id",foreignField:"id",as:"m"}},{$group:{_id:"$nome",livraria_id:{$first:"$m.rua"}}}]);
< { _id: 'Rosalino Matos', livraria_id: [ 'Rua das Sardinhas' ] }
  { _id: 'Ana Matos', livraria_id: [ 'Rua das Sardinhas' ] }
  { _id: 'Miki Mitsubishi', livraria_id: [ 'Rua Santo António' ] }
  { _id: 'Filipe Bastos', livraria_id: [ 'Rua Benfica' ] }
  { _id: 'Marta Carta', livraria_id: [ 'Rua Benfica' ] }
  { _id: 'Carlota Cambalhota',
    livraria_id: [ 'Rua Santo António' ] }
```

Figure 19-Código de procura dos funcionários em cada livraria

3. Conclusões e Trabalho Futuro

Na segunda parte deste projeto, no que diz respeito ao sistema relacional implementado previamente, conseguimos concluir, através do estudo da normalização, que as diversas tabelas do sistema de base de dados relacional se encontram devidamente normalizadas e organizadas. Contudo, foram apenas verificadas as 3 primeiras formas normais. Foram também implementados com sucesso alguns índices, de forma a melhorar o desempenho das queries, através da recuperação de dados mais eficiente.

No que diz respeito à conceção e implementação do sistema de dados em MongoDB, o esquema de base de dados definido no modelo NoSQL é similar ao do modelo relacional, o que favorece o processo de migração de dados.

Após a implementação e desenvolvimento da base de dados no modelo NoSQL, constatamos que era necessário o uso de métodos de referência, de modo a evitar utilização de espaço desnecessário, provocado pela acumulação de documentos dentro de outros e aumentado as chances de ocorrer duplicação de informação.

Concluindo, e tendo em conta o que referimos anteriormente, consideramos que os sistemas de base de dados SQL e NoSQL elaborados pelo grupo são robustos e encontram-se preparados para futuras expansões.

Referências

MongoDB documentation. Available from: <https://docs.mongodb.com/>

Studio 3T documentation. Available from: <https://studio3t.com/knowledge-base/articles/getting-started/>

SQL index information. Available from: <https://www.tutorialspoint.com/sql/sql-indexes.htm>

Overall guidance from: “Connolly, T., Begg, C., Database Systems - A Practical Approach to Design, Implementation, and Management, 6th Edition, Pearson, 2015”

Lista de Siglas e Acrónimos

NIF número de identificação fiscal

SQL structured query language

NoSQL not only SQL