



**Universidade do Minho**

Escola de Engenharia

Mestrado Integrado em Engenharia Informática

## **Unidade Curricular de Bases de Dados**

Ano Letivo de 2015/2016

### **Gestão de dados - Biblioteca**

**André Santos (a61778), Jéssica Pereira (a71164),  
Mariana Carvalho (a67635)**

Janeiro, 2016

Data de Receção	
Responsável	
Avaliação	
Observações	

## **Gestão de dados - Biblioteca**

**André Santos (a61778), Jéssica Pereira (a71164),  
Mariana Carvalho (a67635)**

Janeiro, 2016

## Resumo

O projeto realizado consiste no desenvolvimento de uma base de dados que tem por base os requisitos de funcionamento da Biblioteca Geral da Universidade do Minho (BGUM), sendo considerados com especial relevância os sistemas de consultas bibliográficas e de requisições.

Na primeira fase de desenvolvimento, foi feito um levantamento de requisitos e a sua análise, a fim de determinar a viabilidade do mesmo. Analisados os requisitos, foi elaborado um modelo conceptual, onde foram identificadas as entidades e os relacionamentos entre as mesmas, bem como os atributos que caracterizam cada uma.

Numa segunda fase do projeto foi criado o modelo lógico através da derivação do modelo conceptual. Foi respeitado e validado o processo de normalização através de cada forma normal no esquema lógico e demonstradas as realizações das transações, constando também a sua validação. Foram respeitadas as restrições de integridade. No final desta fase foi estimado o tamanho inicial da base de dados, e feita uma previsão do seu crescimento futuro.

Na terceira e última fase de desenvolvimento do projeto, o esquema lógico foi traduzido para um SGBG, procedendo-se à implementação do modelo físico (criação e povoamento). As transações escolhidas na segunda fase foram traduzidas para linguagem SQL, e foi feita uma estimativa dos requisitos do espaço em disco com base no SGBD escolhido. Por último, definiram-se vistas dos utilizadores e as respetivas regras de acesso.

**Área de Aplicação:** Desenvolvimento de Sistemas de Bases de Dados

**Palavras-Chave:** Modelação Conceptual, Modelação Lógica, Modelação Física, Notação *Chen*, Diagramas ER, Normalização



# Conteúdos

Resumo	i
Conteúdos	iii
Índice de Figuras	vi
Índice de Tabelas	viii
1. Introdução	1
1.1. Contextualização	1
1.2. Apresentação do Caso de Estudo	2
1.3. Motivação	3
1.4. Objetivos	3
1.5. Estrutura do Relatório	3
2. Modelo Conceptual	5
2.1. Levantamento de Requisitos	5
2.2. Identificação de Entidades	7
2.2.1. Livro	7
2.2.2 Autor e Editora	7
2.2.3 Exemplar e Utilizador	7
2.2.4 Utilizador	8
2.2.5 Requisições	8
2.2.6 Localização	8
2.2.7 Colecção	8
2.3. Identificação de Relacionamentos	9
2.3.1 Livro - Autor	9
2.3.2 Livro - Editora	9
2.3.3 Livro - Exemplar	9
2.3.4 Exemplar - Requisição	10
2.3.5 Exemplar - Utilizador	10
2.3.6 Exemplar - Localização	10
2.3.7 Requisição - Utilizador	10
2.3.8 Colecção – Livro	11
2.4. Identificação de Atributos	11
2.4.1 Atributos de Livro	11
2.4.2 Atributos de Autor	12

2.4.3 Atributos de Editora	12
2.4.4 Atributos de Exemplar	12
2.4.5 Atributos de Requisição	12
2.4.6 Atributos de Utilizador	13
2.4.7 Atributos de Localização	13
2.4.8 Atributos de Colecção	14
2.5. Identificação das chaves	14
2.5.1 Livro	14
2.5.2 Autor	14
2.5.3 Editora	14
2.5.4 Exemplar	14
2.5.5 Requisição	15
2.5.6 Localização	15
2.5.7 Utilizador	15
2.5.8 Colecção	15
2.6. Validação do Modelo Conceptual segundo os requisitos	16
2.7. Validação do Modelo Conceptual segundo as transações	22
2.8 Modelo Conceptual Final	23
3. Modelo lógico	24
3.1. Derivação dos Relacionamentos	24
3.1.1 Derivação do Modelo para Obtenção de Tabelas	24
3.1.2 Escolha das Chaves Estrangeiras	26
3.2 Validação segundo Regras de Normalização	29
3.2.1 - 1ª Forma Normal (1FN)	29
3.2.2 - 2ª Forma Normal (2FN)	29
3.2.3 - 3ª Forma Normal (3FN)	32
3.2.4 Forma Normal Boyce-Codd (FNBC)	32
3.3 Validação segundo Transações	33
3.3.1 Qual(is) a(s) localização(ões) (piso, estante e prateleira) dos exemplares de um livro com determinado título?	33
3.3.2 - Efetuar uma requisição	34
3.3.3 - Efectuar Reserva	35
3.4 - Verificação das Restrições de Integridade	36
3.4.1 - Integridade de Domínio	36
3.4.2 - Integridade de Entidade	37
3.4.3 - Integridade Referencial	37
3.4.4 – Restrições de Multiplicidade	37
3.4.5 – Restrições Gerais	37
3.5 Validação de modelo lógico com utilizador	38
3.6 Tamanho inicial e Crescimento Futuro	38

3.6.1 Tamanho inicial	38
3.6.2 Crescimento futuro	39
4. Modelo físico	40
4.1 Tradução do modelo lógico para um SGBD e consequente implementação	40
4.1.1 Relações Base	40
4.1.2 Representação de Atributos Derivados	44
4.1.3 Restrições Gerais	44
4.2 Escolha de Índices	45
4.3 Análise de Transações	45
4.3.1 Qual(is) a(s) localização(ões) (piso, estante e prateleira) dos exemplares de um livro com determinado título?	45
4.3.2 Efectuar requisição	46
4.3.3 Efectuar uma reserva	47
4.4 Estimativa dos Requisitos de Espaço em Disco	48
4.4.2 Povoamento e tamanho inicial	48
4.4.3 Crescimento Futuro	49
4.5 Definição das Vistas de Utilização e Regras de Acesso	50
5. Conclusões e Trabalho Futuro	53
Bibliografia	54
Lista de Siglas e Acrónimos	55
Anexos	56
I. Dicionário de Dados – Entidades	57
II. Dicionário de Dados – Relacionamentos	59
III. Dicionário de Dados – Atributos	60

## Índice de Figuras

Figura 1 - Entidade Coleccao	16
Figura 2 - Relacionamento "Coleccao tem Livro". Os atributos de cada entidade estão omissos.	16
Figura 3 - Relacionamento "Livro escrito por Autor". Os atributos das entidades estão omissos.	16
Figura 4 - Entidade "Autor"	17
Figura 5 - Relacionamento "Livro publicado por Editora. Os atributos das entidades estão omissos.	17
Figura 6 - Entidade "Editora"	17
Figura 7 - Entidade "Livro"	17
Figura 8 - Relacionamento "Coleccao tem Livro". Os atributos das entidades estão omissos.	18
Figura 9 - Relacionamento "Livro tem Exemplar". Os atributos das entidades estão omissos.	19
Figura 10 - Relacionamento "Exemplar tem Localizacao" . Os atributos das entidades foram omitidos.	19
Figura 11 - Entidade "Exemplar"	20
Figura 12 - Relacionamento "Exemplar reservado por Utilizador". Atributos das entidades omitidos.	20
Figura 13 - Relacionamentos e entidades que traduzem a requisição de um exemplar por um utilizador.	20
Figura 14 - Entidade "Requisicao"	21
Figura 15 - Entidade "Utilizador"	21
Figura 16 - Modelo conceptual final	23
Figura 17 - Esquema Lógico Final	28
Figura 18 – Mapa de Transação – Consultar localização de exemplar por título	33
Figura 19 - Mapa de transação – Efectuar requisição	34
Figura 20 - Mapa de transação - Requisitar Exemplar reservado	35
Figura 21 - Tabela representativa da entidade Livro	41
Figura 22 - Tabela representativa do atributo multivalor CDU	41



Figura 23 - Tabela representativa da entidade Coleccao	41
Figura 24 - Tabela representativa da entidade Autor	42
Figura 25 - Tabela representativa da derivação do relacionamento Livro – Autor	42
Figura 26 - Tabela representativa da entidade Editora	42
Figura 27 - Tabela representativa da derivação do relacionamento Livro – Editora	42
Figura 28 - Tabela representativa da entidade Exemplar	42
Figura 29 - Tabela representativa da entidade Localizacao	43
Figura 30 - Tabela representativa da entidade Utilizador	43
Figura 31 - Tabela representativa da derivação do relacionamento Exemplar – Utilizador	43
Figura 32 - Tabela representativa da entidade Requisicao	43
Figura 33 – Nº de registos e tamanho inicial das tabelas	49

## Índice de Tabelas

Tabela 1 - Permissões de cada tipo de utilizador	50
Tabela 2 - Tabela das Entidades	57
Tabela 3 - Tabela de Relacionamentos	59
Tabela 4 -- Tabela de atributos da tabela Livro	60
Tabela 5 - Tabela de atributos da tabela Autor	60
Tabela 6 -- Tabela de atributos do relacionamento Autor - Livro	61
Tabela 7 - Tabela de atributos da tabela Editora	61
Tabela 8 -- Tabelas de atributos do relacionamento Livro - Editora	61
Tabela 9 - Tabela de atributos da tabela Exemplar	61
Tabela 10 - Tabela de atributos da tabela Requisição	62
Tabela 11 - Tabela de atributos da tabela Utilizador	62
Tabela 12 - Tabela de atributos do relacionamento Exemplar - Utilizador	63
Tabela 13 - Tabela de Atributos da tabela Localização	63
Tabela 14 - Tabela dos atributos da tabela Colecção	63

# **1. Introdução**

O projeto consiste na implementação de uma base de dados para a gestão documental da Biblioteca Geral da Universidade do Minho. Neste capítulo faz-se a contextualização do caso de estudo, onde se descrevem as bibliotecas de universidades académicas, das quais a BGUM é um exemplo. Seguidamente faz-se a apresentação do caso de estudo, onde são identificados os principais serviços prestados pela biblioteca. Por último, apresentam-se as motivações para o desenvolvimento desta base de dados em concreto e os objetivos que se pretende atingir com a mesma.

## **1.1. Contextualização**

As universidades são consideradas como portais maiores do conhecimento. Através delas são fomentadas atividades como a investigação e o ensino, de extrema importância para o desenvolvimento das comunidades e regiões em que estão inseridas. Estas atividades envolvem a aquisição de competências práticas, bem como o desenvolvimento do pensamento crítico e analítico. O sucesso da aquisição dessas capacidades depende em grande parte da obtenção de informação por diversas fontes, tornando-se por isso necessário organizar de uma forma cuidada a informação de cada uma das fontes, por forma a permitir um acesso fácil à informação das mesmas.

Apesar de a Internet muitas vezes ser indispensável pelo acesso fácil a uma grande quantidade de informação, a verdade é que na maioria dos casos essa informação não está estruturada, catalogada e não é relevante. As bibliotecas académicas são essenciais na resolução desse problema, uma vez que a informação se encontra catalogada e a maioria dos meios de informação são relevantes, podendo-se encontrar muita da bibliografia sobre temáticas de diversas áreas científicas, bem como artigos de investigação de qualidade. Esta capacidade de organização da informação que as bibliotecas possuem pode ser em grande parte atribuída aos sistemas informáticos que suportam o seu normal funcionamento, que quando bem desenhados, são capazes de responder eficazmente às necessidades dos seus utilizadores, algo que de outra forma seria impossível. Isto explica a adoção generalizada de sistemas informáticos, em particular pelas bibliotecas. Uma biblioteca académica, nomeadamente a BGUM, também não foge à regra.

Nesse contexto, a Universidade do Minho dispõe de um departamento responsável pelos serviços documentais, os SDUM (Serviços Documentais da Universidade do Minho). A principal competência dos SDUM é facultar à academia os recursos bibliográficos necessários para o desempenho das funções de ensino e investigação. Os SDUM gerem o Fundo Documental da Universidade do Minho, que é o catálogo que referencia toda a bibliografia e outros recursos informativos das várias bibliotecas que fazem parte da Universidade do Minho. A BGUM é apenas uma dessas bibliotecas e será nela que este projeto se irá centrar.

## **1.2. Apresentação do Caso de Estudo**

A BGUM oferece uma vasta gama de serviços aos seus utilizadores. Desses serviços, foram selecionados os mais relevantes e os que ao mesmo tempo mais proveito poderiam tirar de terem uma base de dados como suporte:

- Consulta de presença dos fundos documentais das bibliotecas;
- Empréstimo de publicações para leitura domiciliária;
- Assistência a pesquisas em bases de dados a pedido dos utilizadores;

De igual modo, os SDUM possuem uma plataforma WEB onde se podem efetuar as seguintes ações (apenas as mais pertinentes):

- Consultar o catálogo bibliográfico da UM (livros, revistas e outros suportes de informação disponíveis nas bibliotecas UM), bases de dados bibliográficas, revistas em formato eletrónico e outros recursos de informação;
- Consultar ficha de utilizador, onde é possível renovar e reservar empréstimos e publicações.

Embora a BGUM possua bibliografia de diversos tipos (livros, revistas, artigos, publicações, etc..), no âmbito deste trabalho decidiu-se modelar apenas a parte do sistema referente aos livros.

### **1.3. Motivação**

Os sistemas de informação estão a crescer em dimensão e existe uma necessidade premente de uma resposta que para além de eficiente, seja eficaz. O caso de estudo não só permite perceber o funcionamento de uma biblioteca, numa primeira abordagem, bem como entender o funcionamento da BGUM.

Além do mais, o caso de estudo pode ser reaproveitado para outros casos de estudo, como por exemplo comércio a retalho, dado que existe uma convergência entre localização dos produtos e catalogação dos mesmos. Não obstante, existem particularidades do caso de estudo que não podem ser consideradas, tais como reservas e empréstimos para essa extensão. Essas soluções podem ser de mais baixo custo, uma vez que é um produto feito por medida, com a manutenção com um maior controlo, podendo ser integradas com outras tecnologias, como JAVA, .NET, etc.

Finalmente, o caso de estudo é uma oportunidade para solidificar de forma estruturada conhecimentos fundamentais na conceção, desenho e implementação de uma base de dados relacional.

### **1.4. Objetivos**

O objetivo principal deste caso de estudo é criar um sistema de Base de Dados que respeite os requisitos da BGUM de forma a satisfazer os seus utilizadores. Para alcançar este propósito foi seguido um modelo de desenvolvimento de base de dados por etapas: começou-se pela recolha das informações relevantes para a base de dados a desenvolver, seguindo-se uma análise das informações recolhidas, elaboração dos modelos conceptual, lógico e físico em conformidade com os requisitos, seguindo-se uma validação dos mesmos. A documentação de todo o processo encontra-se agregada no presente relatório.

### **1.5. Estrutura do Relatório**

Este documento encontra-se dividido em cinco capítulos: o primeiro, capítulo introdutório, o segundo, referente à modelação conceptual, o terceiro, referente à modelação lógica, o quarto, que diz respeito ao modelo físico e por fim o quinto e último capítulo, onde são tecidas as conclusões e analisado o trabalho futuro.

No primeiro capítulo, sobre o modelo conceptual, estão definidas as seguintes secções: análise de requisitos do caso de estudo, descrição das entidades identificadas, identificação e descrição dos relacionamentos entre entidades, identificação dos atributos de cada entidade, identificação do domínio dos atributos, identificação das chaves primárias, a partir da

identificação de chaves-candidatas, a validação do modelo concetual segundo os requisitos e transações, e a apresentação do modelo concetual – visão global – num diagrama.

No capítulo seguinte é apresentado o processo de transição do modelo conceptual para o modelo lógico, a validação do mesmo e uma análise do possível crescimento do sistema.

No capítulo referente à modelação física, é revista a implementação do sistema no SGBD elegido, justificada a sua utilização e descrito o processo de implementação e criação da base de dados.

## 2. Modelo Conceptual

Neste capítulo detalha-se as etapas seguidas para a elaboração de um modelo conceptual para a base de dados da BGUM. Começa-se por identificar os requisitos do problema bem como explicar de que forma se chegou a tais requisitos. Seguidamente identificam-se as principais entidades e seus atributos bem como os relacionamentos entre elas. Termina-se com a identificação das chaves para cada entidade e a validação do modelo face aos requisitos.

### 2.1. Levantamento de Requisitos

Antes de se ter dado início ao desenvolvimento deste projeto fez-se um levantamento de requisitos, com o objetivo de conhecer quais as necessidades fundamentais de uma biblioteca, em particular na BGUM, no que diz respeito à gestão de livros e às operações que os seus utilizadores têm com o sistema.

Para tal, foram aplicadas várias técnicas de levantamento de requisitos:

- **Observações do comportamento:** foram realizadas visitas à biblioteca nas quais se pôde observar a organização lógica e física dos livros, bem como foram observados *in-loco* os processos de requisição e levantamento de reservas de um livro;
- **Entrevistas:** Ao longo do desenvolvimento foram realizadas algumas pequenas entrevistas às funcionárias da biblioteca. Visto estas funcionárias lidarem diariamente de perto com o sistema, possuem um bom conhecimento do mesmo e por isso foram capazes de esclarecer dúvidas que iam surgindo;
- **Pesquisas na internet:** A Internet foi útil sobretudo para o esclarecimento do significado de alguns termos relacionados com bibliografia que foram surgindo ao longo do desenvolvimento. O sítio da internet dos SDUM<sup>1</sup> foi também bastante útil, pois nele foi possível encontrar informação relativamente ao sistema. Os SDUM têm ainda outra página web<sup>2</sup> onde os utilizadores do

---

<sup>1</sup> Disponível em <http://www.sdum.uminho.pt/>

<sup>2</sup> Disponível em <http://aleph.sdum.uminho.pt/>

sistema podem fazer consultas de livros. Essa página foi também extremamente importante para perceber de que forma os livros eram pesquisados.

Após a aplicação das técnicas acima mencionadas foram considerados relevantes os seguintes requisitos:

1. Saber a designação de todas as coleções existentes.
2. Saber quantos livros cada coleção tem.
3. Saber lista de nomes dos autores dos livros da biblioteca.
4. Saber o apelido e primeiros nomes de um autor.
5. Saber lista de editoras dos livros da biblioteca.
6. Para uma dada editora, saber a sua designação.
7. Pesquisar um livro segundo: ISSN, ISBN, código de barras e título.
8. Saber os livros que um autor escreveu.
9. Saber os livros que uma editora publicou.
10. Saber os livros pertencentes a uma coleção.
11. Saber os livros que têm uma dada CDU.
12. Fazer pesquisa por título do livro, que corresponde a obter uma lista de livros que têm no seu título o conjunto de palavras indicado no campo da pesquisa.
13. Para um dado livro, saber o seu ISSN, ISBN, código de barras, título, editora, autor, edição, CDU, ano de publicação e número de exemplares.
14. Saber a localização de livros de uma certa CDU.
15. Para cada exemplar saber o estado de disponibilidade (reservado, requisitado ou não requisitável), o estado de conservação do exemplar bem como a sua localização na biblioteca (pisso, estante e prateleira).
16. Reservar exemplares de um ou mais livros.
17. Saber data de reserva e seu estado (pendente, exemplar disponível para levantamento, reserva concluída ou cancelada).
18. Cancelar reserva de exemplar.
19. Efetuar requisição de um exemplar.
20. Para uma requisição, saber o seu estado (ativa ou não), em que data foi realizada, em que data deverá ser entregue o exemplar, qual o número de renovações efetuado e qual o número máximo de renovações permitidas em vigor na data da reserva.
21. Renovar uma requisição, não excedendo o número máximo de renovações permitido.
22. Concluir requisição, que corresponde à devolução do exemplar requisitado.
23. Gerar estatísticas de número de renovações médio e saber quantos utilizadores usam o número máximo de requisições permitidas.
24. Registrar utilizadores internos ou externos como requisitantes.



25. Para um dado utilizador, saber o seu tipo, nome, email, CC, número mecanográfico e contacto alternativo.
26. Saber os utilizadores que reservaram/requisitaram determinado exemplar.

## **2.2. Identificação de Entidades**

No anexo I encontra-se a documentação das entidades, na forma de um dicionário de dados. Nesta secção pretende-se complementar o dicionário de dados com justificação face aos requisitos das entidades que se consideraram.

### **2.2.1. Livro**

Dado o domínio do problema, a modelação da base de dados dos livros da BGUM, foi natural que a entidade central e a mais indutiva a surgir tenha sido a entidade Livro. Analisando os requisitos, observa-se que a maior parte deles se encontra relacionado com livros, seja para consultas sobre os seus atributos ou a gestão de requisições e reservas feitas pelos utentes da biblioteca.

### **2.2.2 Autor e Editora**

Decidiu-se ter Autor e Editora como entidades sobretudo para prevenir duplicação de dados e evitar redundância. Foi considerado ter o Autor e a Editora como atributos da entidade “Livro”, no entanto, um autor pode escrever vários livros, assim como uma editora também pode ter muitas publicações, o que levaria a redundância de dados caso fossem considerados como atributos.

### **2.2.3 Exemplar e Utilizador**

Apesar do Livro ser a entidade central, na verdade muitos dos requisitos levam à existência de uma outra entidade, o Exemplar, que representa uma cópia de um livro. Inicialmente foi tido em consideração a entidade Livro ter um atributo que refletisse o número de cópias disponíveis, que seria decrementado a cada requisição sobre esse livro. No entanto, com essa solução não era possível expressar que cada exemplar de um livro pode ser individualmente requisitado/reservado e um mesmo livro pode estar requisitado/requisitado a vários utilizadores, uma vez que existem vários exemplares do livro. Por este motivo, surgiu naturalmente a necessidade da entidade Exemplar.

## **2.2.4 Utilizador**

O Utilizador representa um utente da biblioteca, que pode fazer consultas sobre o sistema e participar em reservas e requisições de livros. Possui atributos próprios e necessita ser identificado univocamente, o que justifica a sua existência como entidade na base de dados.

## **2.2.5 Requisições**

As requisições são também um conceito chave no sistema, pois é referida em alguns dos requisitos. A requisição tem um estado próprio associado e um identificador único, daí ter sido considerada como entidade.

## **2.2.6 Localização**

Da análise dos requisitos surgiu também a necessidade da entidade Localização, por ser necessário saber facilmente onde um exemplar se encontra no espaço da biblioteca, identificando o piso, estante e prateleira do mesmo. Esta informação é útil tanto para leitores da biblioteca que podem saber onde procurar um determinado livro bem como para as funcionárias que necessitam de saber onde determinado exemplar deve ser arrumado. Em alternativa a ser considerada entidade, a Localização poderia ser considerada como atributo de exemplar, no entanto visto existirem vários exemplares numa determinada localização, decidiu-se considerar como entidade para evitar a redundância de dados.

## **2.2.7 Colecção**

A colecção representa um conjunto de livros. Uma vez que uma colecção tem vários livros, optou-se por considerar a Colecção como uma entidade para evitar a redundância de dados caso esta fosse considerada como atributo da entidade “Livro” por exemplo.

## 2.3. Identificação de Relacionamentos

A documentação sobre os relacionamentos pode ser encontrada no anexo II, no respectivo dicionário de dados. Após serem identificadas as entidades, foram determinados os relacionamentos entre cada uma das mesmas:

### 2.3.1 Livro - Autor

Relacionamento: Livro é escrito por autor

Descrição: Identifica a escrita de um livro por um autor

Cardinalidade: Livro(N); Autor(N) - Um livro pode ter vários autores, e um autor pode escrever vários livros.

Participação: Livro(T); Autor(T) - Todos os livros são escritos por autores, e todos os autores escrevem pelo menos um livro

Atributos: Este relacionamento não tem atributos.

### 2.3.2 Livro - Editora

Relacionamento: Livro tem editora; Livro é publicado por editora

Descrição: Identifica a publicação de um livro por uma editora

Cardinalidade: Livro(N); Editora(N) - Um livro pode ter sido publicado por várias editoras, e uma editora pode publicar vários livros.

Participação: Livro(T); Editora(T) - Todos os livros são publicados por editoras, e todas as editoras publicam pelo menos um livro

Atributos:

- **Edição** - Edição de um livro por uma dada editora.
- **Ano** - Ano em que uma edição do livro foi publicada.

### 2.3.3 Livro - Exemplar

Relacionamento: Livro tem exemplar

Descrição: Identifica que um livro pode ter mais do que uma cópia

Cardinalidade: Livro(1); Exemplar(N) - Um livro pode ter vários exemplares mas um exemplar apenas diz respeito a um livro

Participação: Livro(T); Exemplar(T) – Todos os livros têm exemplares e todos os exemplares estão associados a um livro.

Atributos: Este relacionamento não tem atributos

### 2.3.4 Exemplar - Requisição

Relacionamento: Exemplar participa em requisição

Descrição: Identifica o envolvimento de um exemplar no processo de requisição

Cardinalidade: Exemplar(1); Requisição(N) - Um exemplar pode participar em várias requisições, mas ao registo de uma requisição só diz respeito um exemplar

Participação: Exemplar(P); Requisição(T) - Nem todos os exemplares são requisitados mas todas as requisições são de um exemplar.

Atributos: Este relacionamento não tem atributos

### 2.3.5 Exemplar - Utilizador

Relacionamento: Exemplar é reservado por utilizador

Descrição: Identifica a reserva de um exemplar por parte de um utilizador

Cardinalidade: Exemplar(N); Utilizador(N) - Um exemplar pode ser reservado por vários utilizadores, e um utilizador pode reservar vários exemplares.

Participação: Exemplar(P); Utilizador(P) – Nem todos os exemplares são reservados e nem todos os utilizadores efetuam reservas.

Atributos:

- **DataReserva** - Data em que o pedido de reserva foi feito
- **Estado** - Em que estado se encontra a reserva

### 2.3.6 Exemplar - Localização

Relacionamento: Exemplar tem localização; Exemplar está em localização

Descrição: Identifica a localização de um exemplar no espaço da biblioteca

Cardinalidade: Exemplar(N);Localização (1) - Um exemplar tem apenas uma localização, mas uma localização pode dizer respeito a vários exemplares

Participação: Exemplar(T);Localização (P) – Todos os exemplares estão numa determinada localização, mas nem todas as localizações têm exemplares.

Atributos: Este relacionamento não tem atributos

### 2.3.7 Requisição - Utilizador

Relacionamento: Requisição é feita por utilizador

Descrição: Identifica a participação de um utilizador numa requisição

Cardinalidade: Requisição(N); Utilizador(1) - Uma requisição é concretizada apenas por um utilizador, mas um utilizador pode realizar várias requisições

Participação: Requisição(T); Utilizador(P) – Todas as requisições são feitas por um utilizador mas nem todos os utilizadores fazem requisições.

Atributos: Este relacionamento não tem atributos

## 2.3.8 Colecção – Livro

Relacionamento: Colecção tem livro

Descrição: Descreve o facto de um livro pertencer a uma colecção

Cardinalidade: Colecção(1); Livro(N) – Uma colecção tem vários livros no entanto um livro só tem uma colecção

Participação: Colecção(P); Livro(P) – Nem todas as colecções têm que ter livros e um livro pode ou não pertencer a uma colecção.

Atributos: Este relacionamento não tem atributos

## 2.4. Identificação de Atributos

No anexo III encontra-se o dicionário de dados com a descrição dos atributos de cada entidade. Nesta secção, pretende-se complementar os dicionários de dados com algumas justificações face aos requisitos que levaram a considerar os atributos que se tem no esquema conceptual.

### 2.4.1 Atributos de Livro

Excluindo a sua chave primária, idLivro, livro tem os seguintes atributos: ISSN, ISBN, CodBarras, Título e CDU. A justificação da existência tem a ver principalmente com os requisitos que exigem que um livro seja pesquisável por estes campos. Destaca-se como atributo a Classificação Decimal Unitária (CDU) que é um identificador usado em sistemas bibliográficos para categorização de livros. No caso da nossa biblioteca, este atributo é importante sobretudo em termos de localização. Optou-se por ter como atributo e não como entidade dado a dimensão da base de dados, que não justificava ter este atributo como entidade. Este atributo é multi-valor uma vez que na realidade, uma CDU pode ser complexa. Uma CDU pode ser apenas “128.0” mas também poderá ser “128-150”, representando todas as CDU entre 128 e 150. Para evitar lidar com a complexidade que a CDU apresenta, optou-se por colocar um atributo multi-valor para tentar simular essa complexidade. A elevada complexidade foi também o motivo pelo qual não se decidiu subdividir em CDU em mais campos.

## 2.4.2 Atributos de Autor

Excluindo a sua chave primária, idAutor, o autor tem o atributo composto Nome, que inclui os primeiros nomes e o apelido. A separação dos nomes em primeiros nomes e apelido deve-se ao facto da forma como o nome do autor é apresentado nas informações bibliográficas do livro. Não só no sistema desta biblioteca como no universo bibliotecário em geral, é usual identificar o autor pelo apelido, e só depois apresentar os últimos nomes. Em alguns casos, na identificação de um autor apenas se coloca o apelido completo, seguido das iniciais dos primeiros nomes.

## 2.4.3 Atributos de Editora

Excluindo a sua chave primária idEditora, a editora tem o atributo Designação, que corresponde ao nome da editora. Pela análise dos requisitos todas as pesquisas envolviam apenas a designação da editora, pelo que se concluiu que não seria necessário guardar mais nenhuma informação para esta entidade.

## 2.4.4 Atributos de Exemplar

Excluindo a sua chave primária idExemplar, o Exemplar possui ainda os seguintes atributos:

- **Condição** – Para efeitos de renovação de exemplares estragados e responsabilização dos utilizadores pouco cuidados com os livros, é necessário manter um registo sobre a condição atual do livro.
- **Disponibilidade** – Para efeitos de requisição/reserva, é necessário sempre saber qual a disponibilidade atual do livro. A título de exemplo, os requisitos exigem que um livro cuja disponibilidade seja “apenas consulta” não pode ser requisitado nem reservado.

## 2.4.5 Atributos de Requisição

Excluindo a sua chave primária idExemplar, o Exemplar possui ainda os seguintes atributos:

- **Estado** – Representa o estado da requisição: se o livro foi entregue ou o utilizador ainda o tem. Este atributo permite manter um histórico de todas as requisições e ainda assim distinguir entre as que devem ser consideradas no momento presente para efeito de contacto com os utilizadores ou não. Por exemplo, se a data de entrega do livro já tiver passado, mas o livro já tiver sido entregue, não se deve alertar o utilizador de que tem livros em atraso.
- **NroMaxRenovacoes** – Número máximo de renovações permitidas para aquele livro pelo utilizador. Optou-se por ter este número como atributo, porque embora ele seja

igual para muitos livros, pode mudar de ano para ano, e no histórico das requisições, para fins estatísticos é útil saber o nº máximo de requisições em vigor na altura em que a requisição foi feita.

- **NroRenovações** – Número atual de renovações que o utilizador efetuou. Em conjunto com o atributo NroMaxRenovacoes permite saber se um utilizador pode continuar a renovar a requisição de um livro ou não.
- **DataEntrega** – Data em que o livro tem que ser entregue. Permite satisfazer o requisito de saber de uma forma rápida se a requisição está em atraso ou não.
- **DataRequisicao** – Data em que um livro foi requisitado. Útil sobretudo para fins estatísticos, por exemplo, saber os meses em houve mais requisições.

## 2.4.6 Atributos de Utilizador

Além da sua chave primária, idUser, o Utilizador tem os seguintes atributos:

- **Tipo** – Indicação do tipo de utilizador. Útil sobretudo para requisitos relacionados com fins estatísticos do tipo de utilizadores que utiliza os serviços da biblioteca.
- **Nome** – Nome do utilizador.
- **Email** – Principal meio de contacto do sistema para com os utilizadores.
- **CC** – Cartão de Cidadão. Identifica o utilizador enquanto cidadão, o que poderá ser usado em termos de apuração de responsabilidades caso seja
- **NroMecanografico** – Número de utilizador. Em conjunto com o tipo de utilizador, constitui uma identificação do utilizador importante no contexto do meio académico do qual a biblioteca faz parte.
- **Telefone** – Contacto telefónico alternativo ao *e-mail*, destinado a ser usado para assuntos urgentes.

## 2.4.7 Atributos de Localização

Além da sua chave primária, idLocal, a Localização tem como atributos o Piso, Estante e Prateleira. Estes atributos visam localizar facilmente e com alguma especificidade, um livro no espaço da biblioteca. Saber a localização é um requisito importante visto que bastantes pesquisas feitas aos livros da base de dados destinam-se a saber a localização dos exemplares de um livro.

## 2.4.8 Atributos de Colecção

Além da sua chave primária, idColeccao, esta entidade possui ainda outro atributo “Designacao”, que corresponde ao nome da colecção. Este foi o único atributo relevante considerado para esta entidade.

## 2.5. Identificação das chaves

### 2.5.1 Livro

**Chaves candidatas:** ISBN, ISSN, CodBarras, idLivro

**Chave primária escolhida:** idLivro

**Justificação:** Atributo de menor tamanho

**Chaves secundárias:** ISBN, ISSN, CodBarras

### 2.5.2 Autor

**Chaves candidatas:** idAutor

**Chave primária escolhida:** idAutor

**Justificação:** O único atributo que pode identificar univocamente uma instância de entidade.

**Chaves secundárias:** Nenhumas

### 2.5.3 Editora

**Chaves candidatas:** idEditora, Designacao

**Chave primária escolhida:** idEditora

**Justificação:** Chave de menor tamanho não textual

**Chaves secundárias:** Designacao

### 2.5.4 Exemplar

**Chaves candidatas:** idExemplar

**Chave primária escolhida:** idExemplar

**Justificação:** O único atributo que pode identificar univocamente uma instância de entidade.

**Chaves secundárias:** Nenhumas



### 2.5.5 Requisição

**Chaves candidatas:** idRequisicao

**Chave primária escolhida:** idRequisicao

**Justificação:** O único atributo que pode identificar univocamente uma instância de entidade.

**Chaves secundárias:** Nenhumas

### 2.5.6 Localização

**Chaves candidatas:** idLocal

**Chave primária escolhida:** idLocal

**Justificação:** O único atributo que pode identificar univocamente uma instância de entidade.

**Chaves secundárias:** Nenhumas

### 2.5.7 Utilizador

**Chaves candidatas:** idUser, CC

**Chave primária escolhida:** idUser

**Justificação:** Atributo de menor tamanho e de menor probabilidade de ser alterado.

**Chaves secundárias:** CC

### 2.5.8 Colecção

**Chaves candidatas:** idColecao

**Chave primária escolhida:** idColecao

**Justificação:** O único atributo que pode identificar univocamente uma instância de entidade.

**Chaves secundárias:** Nenhumas

## 2.6. Validação do Modelo Conceptual segundo os requisitos

Para mostrar a validade do modelo conceptual, para cada um dos requisitos apresenta-se a forma como o modelo conceptual lhes dá resposta.

### 1. Saber a designação de todas as colecções existentes

A entidade “Coleccao” guarda informação sobre a designação de cada colecção.

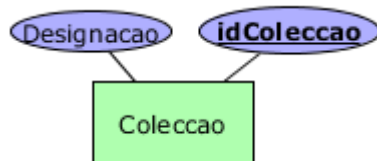


Figura 1 - Entidade Coleccao

### 2. Saber quantos livros cada colecção tem

O relacionamento “Coleccao tem livro” permite dar resposta a este requisito.

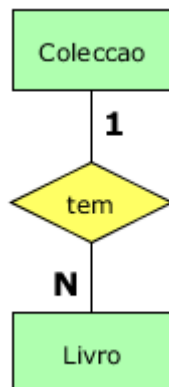


Figura 2 - Relacionamento "Coleccao tem Livro". Os atributos de cada entidade estão omissos.

### 3. Saber lista de nomes dos autores dos livros da biblioteca.

O relacionamento “Livro escrito por Autor” permite dar resposta a este requisito.



Figura 3 - Relacionamento "Livro escrito por Autor". Os atributos das entidades estão omissos.

**4. Saber o apelido e primeiros nomes de um autor.**

A entidade “Autor” guarda informação sobre o apelido e primeiros nomes dos autores.

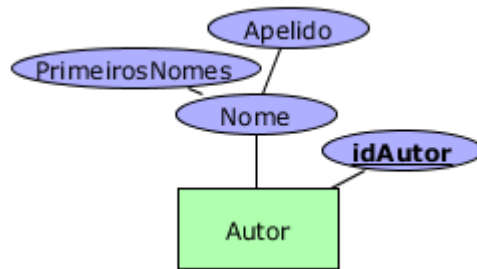


Figura 4 - Entidade "Autor"

**5. Saber lista de editoras dos livros da biblioteca.**

O relacionamento “Livro publicado por Editora” permite dar resposta a este requisito.



Figura 5 - Relacionamento "Livro publicado por Editora. Os atributos das entidades estão omissos.

**6. Para uma dada editora, saber a sua designação.**

A entidade “Editora” guarda a designação de cada editora.



Figura 6 - Entidade "Editora"

**7. Pesquisar um livro segundo: ISSN, ISBN, código de barras e título.**

A entidade “Livro” guarda informação sobre o ISSN, ISBN, código de barras e título.

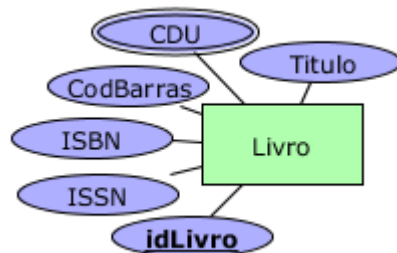


Figura 7 - Entidade "Livro"

**8. Saber os livros que um autor escreveu.**

O relacionamento “Livro escrito por Autor” permite dar resposta a este requisito. Ver Figura 3.

**9. Saber os livros que uma editora publicou.**

O relacionamento “Livro publicado por Editora” permite dar resposta a este requisito. Ver Figura 5.

**10. Saber os livros pertencentes a uma colecção.**

O relacionamento “Colecao tem Livro” permite dar resposta a este requisito.

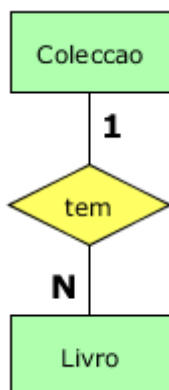


Figura 8 - Relacionamento "Colecao tem Livro". Os atributos das entidades estão omissos.

**11. Saber os livros que têm uma dada CDU.**

A entidade “Livro” guarda informação sobre a CDU de um livro. Ver figura 7.

**12. Fazer pesquisa por título do livro, que corresponde a obter uma lista de livros que têm no seu título o conjunto de palavras indicado no campo da pesquisa.**

A entidade “Livro” guarda informação sobre o título de um livro. Ver figura 7.

**13. Para um dado livro, saber o seu ISSN, ISBN, código de barras, título, editora, autor, edição, CDU, ano de publicação e número de exemplares.**

A entidade “Livro” guarda informação sobre o ISSN, ISBN, código de barras, título e CDU de um livro (Ver figura 7). A editora, edição e ano de publicação são dados pelo relacionamento “Livro publicado por Editora” (Figura 5), o autor é dado pelo relacionamento “Livro escrito por Autor” (Figura 3) e o número de exemplares pode ser obtido pelo relacionamento “Livro tem Exemplar” (Figura 9).

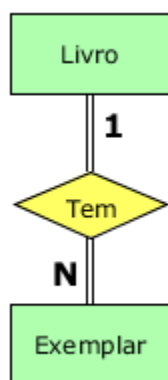


Figura 9 - Relacionamento "Livro tem Exemplar". Os atributos das entidades estão omissos.

#### 14. Saber a localização de livros de uma certa CDU.

Através do relacionamento “Livro tem Exemplar” (Figura 9) é possível saber os exemplares com determinada CDU. Através do relacionamento “Exemplar tem Localização” (Figura 10) é possível saber a localização desses exemplares.

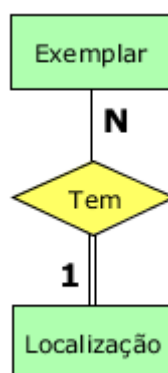


Figura 10 - Relacionamento "Exemplar tem Localizacao" . Os atributos das entidades foram omitidos.

#### 15. Para cada exemplar saber o estado de disponibilidade (reservado, requisitado ou não requisitável), o estado de conservação do exemplar bem como a sua localização na biblioteca ( piso, estante e prateleira).

A entidade “Exemplar” guarda informação sobre o estado de disponibilidade (atributo “Disponibilidade”) e conservação do exemplar (atributo “Condicao”), como pode ser visto na Figura 11. A localização do exemplar é dada pelo relacionamento “Exemplar tem Localizacao” (Figura 10).

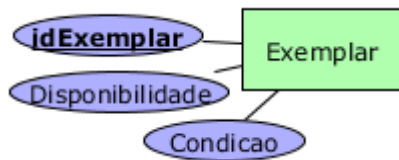


Figura 11 - Entidade "Exemplar"

**16. Reservar exemplares de um ou mais livros.**

O relacionamento "Exemplar reservado por Utilizador" garante este requisito.



Figura 12 - Relacionamento "Exemplar reservado por Utilizador". Atributos das entidades omitidos.

**17. Saber data de reserva e seu estado (pendente, exemplar disponível para levantamento, reserva concluída ou cancelada).**

O relacionamento "Exemplar reservado por Utilizador" guarda a data de reserva e respetivo estado (atributos do relacionamento). Ver Figura 12.

**18. Cancelar reserva de exemplar.**

O relacionamento "Exemplar reservado por Utilizador" guarda o estado da reserva (atributo do relacionamento). Ver Figura 12.

**19. Efetuar requisição de um exemplar.**

O relacionamento "Exemplar participa em Requisicao" em conjunto com o relacionamento "Requisicao feita por utilizador" dão resposta a este requisito.

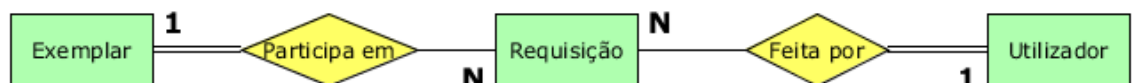


Figura 13 - Relacionamentos e entidades que traduzem a requisição de um exemplar por um utilizador.

**20. Para uma requisição, saber o seu estado (ativa ou não), em que data foi realizada, em que data deverá ser entregue o exemplar qual o número de renovações efetuado e qual o número de máximo de renovações em vigor na data da reserva.**

Os atributos da entidade "Requisicao" dão resposta a este requisito.



Figura 14 - Entidade "Requisicao"

**21. Renovar uma requisição, não excedendo o número máximo de renovações permitido.**

O atributo “NroRenovacoes” da entidade “Requisicao” dá resposta a este requisito. Ver Figura 14.

**22. Concluir requisição, que corresponde à devolução do exemplar requisitado.**

O atributo “Estado” da entidade “Requisicao” dá resposta a este requisito. Ver Figura 14.

**23. Gerar estatísticas de número de renovações médio e saber quantos utilizadores usam o número máximo de requisições permitidas.**

Os atributos “NroRenovacoes” e “NroMaxRenovacoes” da entidade “Requisicao” dão resposta a este requisito. Ver Figura 14.

**24. Registrar utilizadores internos ou externos como requisitantes.**

O atributo “Tipo” da entidade “Utilizador” permite dar resposta a este requisito.

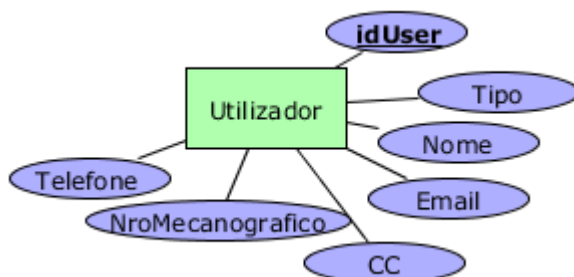


Figura 15 - Entidade "Utilizador"

**25. Para um dado utilizador, saber o seu tipo, nome, email (contacto principal), CC, número mecanográfico e contacto alternativo (telefone).**

Os atributos da entidade “Utilizador” permitem dar resposta a este requisito. Ver Figura 15.

## **26. Saber os utilizadores que reservaram/requisitaram determinado exemplar.**

O relacionamento “Exemplar reservado por Utilizador” (Figura 12) e o conjunto de relacionamentos mostrados na figura 13 dão resposta a este requisito.

## **2.7. Validação do Modelo Conceptual segundo as transações**

Nesta secção pretende-se mostrar que o modelo conceptual produzido consegue dar resposta algumas transações impostas pelos requisitos e com isso ter mais um elemento de validação do modelo. Optou-se por apresentar transações consideradas importantes dados os requisitos e descrever textualmente a forma como o modelo conceptual lhes dá resposta.

- **Qual(is) a(s) localização(ões) (pisso, estante e prateleira) dos exemplares de um livro com determinado título?**

A entidade Livro tem associado os detalhes de um livro, nomeadamente o seu título. Através do relacionamento entre Livro e Exemplar é possível saber a informação de todos os exemplares desse livro, e pelo relacionamento entre Exemplar e Localização é possível saber a localização de cada um dos exemplares, obtendo-se assim as possíveis localizações onde o exemplar do livro pode ser encontrado.

- **Efetuar uma requisição**

Antes de uma requisição poder ser registada é necessário ver se esta pode ser feita, o que implica verificar se o exemplar não possui reservas, que pode ser verificado no relacionamento “Exemplar reservado por Utilizador” e que o livro não é apenas de consulta, que pode ser visto através da entidade Exemplar, que acolhe o estado de disponibilidade de um exemplar. Caso o exemplar esteja disponível para requisição acolhe as informações de uma requisição. A entidade exemplar contém o estado de disponibilidade do mesmo, que terá que ser actualizado depois de feita a requisição. É ainda necessário verificar se a requisição corresponde ao levantamento de uma reserva, que pode ser verificado no relacionamento “Exemplar reservado por Utilizador”.

- **Efectuar uma reserva**

Antes de um utilizador poder efectuar uma reserva é necessário ver se esta pode ser feita, o que implica verificar se o exemplar pode ser requisitado, que pode ser visto através da entidade Exemplar e que actualmente não possui reservas por parte de outros utilizadores, que pode ser verificado no relacionamento “Exemplar reservado por Utilizador”. Caso a reserva possa ser efectuada, a informação da mesma é registada no relacionamento “Exemplar reservado por Utilizador”.



## 2.8 Modelo Conceptual Final

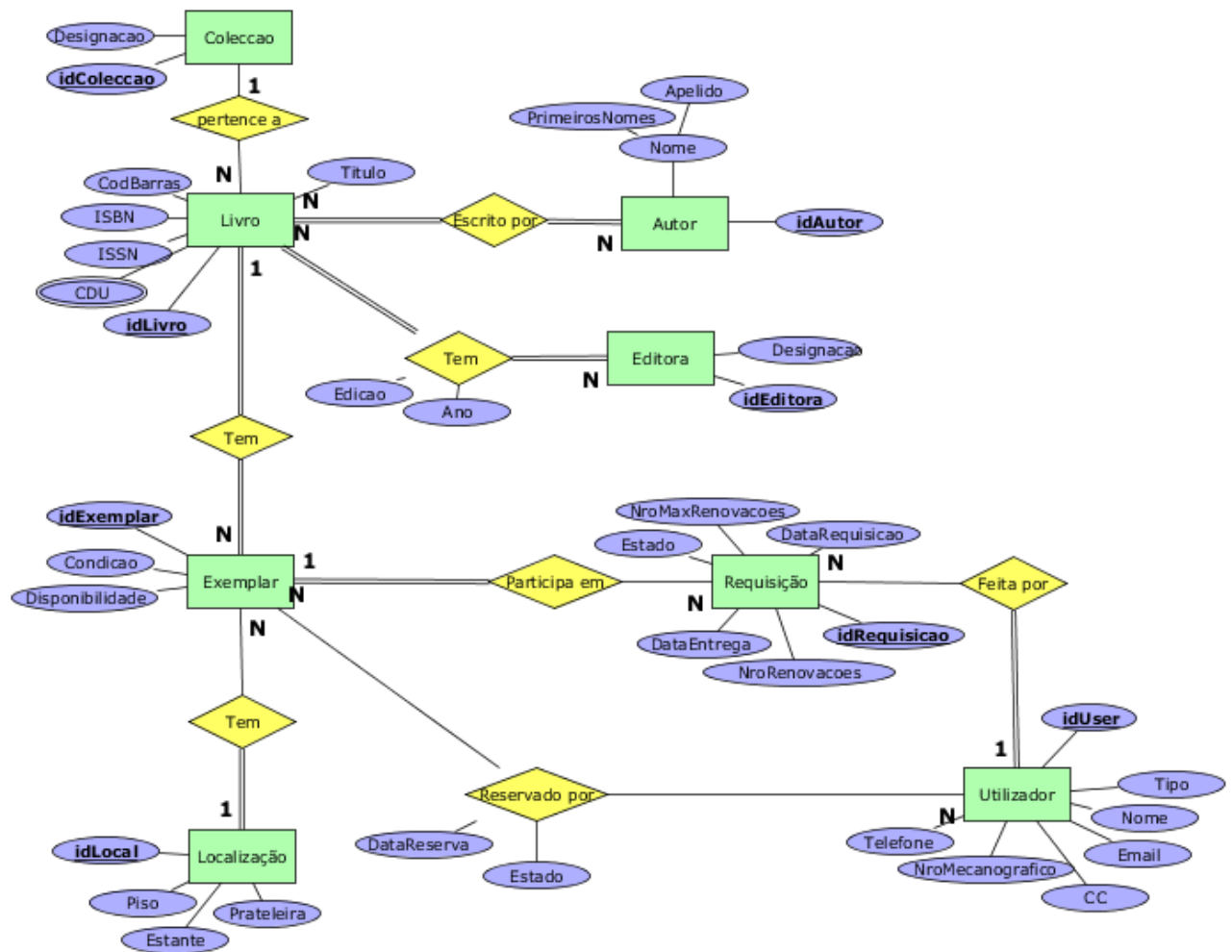


Figura 16 - Modelo conceptual final

## 3. Modelo lógico

Este capítulo trata da derivação do modelo lógico a partir do modelo conceptual apresentado no capítulo anterior. Neste capítulo vão ser explicados os passos de derivação do modelo conceptual para obtenção do modelo lógico. Apresenta-se em primeiro lugar a construção das tabelas do modelo lógico seguindo-se a sua validação.

### 3.1. Derivação dos Relacionamentos

#### 3.1.1 Derivação do Modelo para Obtenção de Tabelas

Nesta secção apresenta-se a forma como cada tabela do modelo lógico foi obtida. Para cada tabela apresentam-se os seus atributos, sendo indicadas as suas chaves primárias e estrangeiras bem como o motivo da tabela surgir no modelo lógico.

**Utilizador** = {idUser, Tipo, Nome, Email, CC, NroMecanografico, Telefone}

**Chave primária:** idUser

**Chaves estrangeiras:** Nenhuma

**Observações:** Relação de base. Utilizador é uma entidade no modelo conceptual.

**Livro** = {idLivro, Titulo, CodBarras, ISBN, ISSN, Coleccao}

**Chave primária:** idLivro

**Chaves estrangeiras:** Coleccao

**Observações:** Relação de base. Livro é uma entidade no modelo conceptual.

**CDU** = {CDU, Livro}

**Chave primária:** CDU, Livro

**Chaves estrangeiras:** Livro

**Observações:** Derivação do atributo multivalor “CDU” da entidade “Livro”.

**Autor** = {idAutor, PrimeirosNomes, Apelido}

**Chave primária:** idAutor

**Chaves estrangeiras:** nenhuma

**Observações:** Relação de base. Autor é uma entidade no modelo conceptual.

**Editora** = {idEditora, Designacao}

**Chave primária:** idEditora

**Chaves estrangeiras:** nenhuma

**Observações:** Relação de base. Editora é uma entidade no modelo conceptual.

**Requisicao** = {idRequisicao, DataRequisicao, DataEntrega, Estado,  
NroMaxRenovacoes, NrRenovacoes, Exemplar, Utilizador}

**Chave primária:** idRequisicao

**Chaves estrangeiras:** Exemplar, Utilizador

**Observações:** Relação de base. Requisicao é uma entidade no modelo conceptual.

**Exemplar** = {idExemplar, Condicao, Disponibilidade, Localizacao, Livro}

**Chave primária:** idExemplar

**Chaves estrangeiras:** Localizacao, Livro

**Observações:** Relação de base. Exemplar é uma entidade no modelo conceptual.

**Localizacao** = {IdLocal, Piso, Estante, Prateleira}

**Chave primária:** IdLocal

**Chaves estrangeiras:** Nenhuma

**Observações:** Relação de base. Localizacao é uma entidade no modelo conceptual.

**Exemplar-Reservado-Utilizador** = {Exemplar, Utilizador, DataReserva, Estado}

**Chave primária:** Exemplar, Utilizador, DataReserva (composta)

**Chaves estrangeiras:** Exemplar, Utilizador

**Observações:** Derivação do relacionamento binário Exemplar(N) – Utilizador(N)

**Livro-Publicado-Editora** = {Livro, Editora, Edicao, Ano}

**Chave primária:** Livro, Editora, Edicao (composta)

**Chaves estrangeiras:** Livro, Editora

**Observações:** Derivação do relacionamento binário Livro(N) – Editora(N)

**Autor-escreve-Livro** = {Livro, Autor}

**Chave primária:** Livro, Autor (composta)

**Chaves estrangeiras:** Livro, Autor

**Observações:** Derivação do relacionamento binário Autor(N) – Livro(N)

**Coleccao** = {idColeccao, Designacao}

**Chave primária:** idColeccao

**Chaves estrangeiras:** Nenhuma

**Observações:** Relação de base. Coleccao é uma entidade no modelo conceptual.

Obtêm-se assim 12 tabelas: 8 tabelas de base, 1 tabela resultante da derivação do atributo multivalor CDU e 3 tabelas resultantes da derivação de relacionamentos N:N.

### 3.1.2 Escolha das Chaves Estrangeiras

Na secção anterior foram indicadas as chaves estrangeiras de cada tabela. Nesta secção pretende-se complementar essa informação, justificando com base nas regras de derivação, o motivo das chaves estrangeiras surgirem em cada tabela. Assim, para cada tabela apresenta-se uma lista das suas chaves estrangeiras bem como uma explicação de como a derivação do modelo conceptual originou cada chave estrangeira.

#### Tabela Livro

- **Coleccao:** Cópia da chave primária da tabela “Coleccao”. Existe um relacionamento de 1 (Coleccao) para N (Livro). A tabela “Livro” funciona por isso como tabela filho e recebe uma cópia da chave primária da tabela “Coleccao”.

#### Tabela CDU

Esta tabela resulta da derivação do atributo multivalor “CDU” da entidade “Livro”, contendo por isso uma cópia da chave primária da entidade “Livro”.

- **Livro:** Cópia da chave primária de “Livro”.

#### Tabela Requisicao

- **Exemplar:** Cópia da chave primária da tabela “Exemplar”. Existe um relacionamento de 1 (Exemplar) para N (Requisicao). A tabela “Requisicao” funciona por isso como tabela filho e recebe uma cópia da chave primária da tabela “Exemplar”.
- **Utilizador:** Cópia da chave primária da tabela “Utilizador”. Existe um relacionamento de 1 (Utilizador) para N (Requisicao). A tabela “Requisicao” funciona por isso como tabela filho e recebe uma cópia da chave primária da tabela “Utilizador”.

### **Tabela Exemplar**

- **Localizacao:** Cópia da chave primária da tabela “Localizacao”. Existe um relacionamento de 1 (Localizacao) para N (Exemplar). A tabela “Exemplar” funciona por isso como tabela filho e recebe uma cópia da chave primária da tabela “Localizacao”.
- **Livro:** Cópia da chave primária da tabela “Localizacao”. Existe um relacionamento de 1 (Livro) para N (Exemplar). A tabela “Exemplar” funciona por isso como tabela filho e recebe uma cópia da chave primária da tabela “Livro”.

### **Tabela Exemplar-Reservado-Utilizador**

Esta tabela resulta da derivação do relacionamento N (Exemplar) para N (Utilizador). Recebe por isso uma cópia da chave primária da tabela “Exemplar” e uma cópia da chave primária da tabela “Utilizador”.

- **Exemplar:** Cópia da chave primária da tabela “Exemplar”.
- **Utilizador:** Cópia da chave primária da tabela “Utilizador”.

### **Tabela Livro-Publicado-Editora**

Esta tabela resulta da derivação do relacionamento N (Livro) para N (Editora). Recebe por isso uma cópia da chave primária da tabela “Livro” e uma cópia da chave primária da tabela “Editora”:

- **Livro:** Cópia da chave primária da tabela “Livro”.
- **Editora:** Cópia da chave primária da tabela “Editora”.

### **Tabela Autor-escreve-Livro**

Esta tabela resulta da derivação do relacionamento N (Autor) para N (Livro). Recebe por isso uma cópia da chave primária da tabela “Autor” e uma cópia da chave primária da tabela “Livro”:

- **Livro:** Cópia da chave primária da tabela “Livro”.
- **Autor:** Cópia da chave primária da tabela “Autor”.

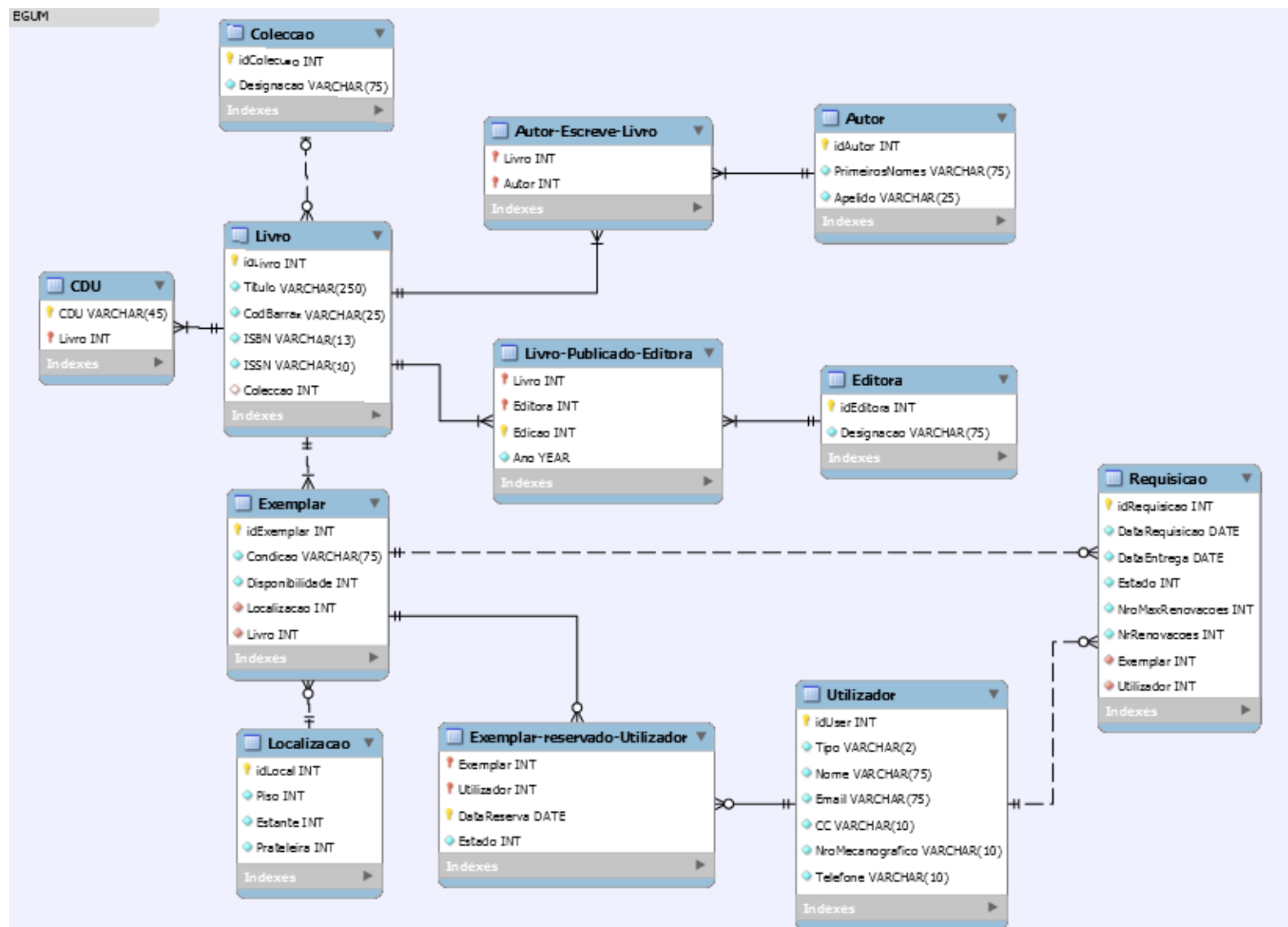


Figura 17 - Esquema Lógico Final

## 3.2 Validação segundo Regras de Normalização

Nesta secção é feita a validação do modelo lógico segundo as regras de normalização, de forma a garantir que as tabelas da base de dados têm o número mínimo e suficiente de atributos que suporte os requisitos impostos. Tal garante a menor redundância de dados possível, prevenindo inconsistências que poderão surgir na base de dados. Com recurso à análise das dependências funcionais entre os atributos de cada tabela, nesta secção apresenta-se o processo de validação segundo as regras de normalização até à forma normal Boyce-Codd.

### 3.2.1 - 1ª Forma Normal (1FN)

A existência de uma chave primária como sendo um identificador único em cada tabela garante que não há conjuntos de atributos repetidos que descrevam a mesma característica. Além disso cada atributo contém apenas valores atômicos e não há tabelas com o mesmo nome nem atributos com o mesmo nome na mesma tabela. O modelo lógico obtido encontra-se por isso na 1ª Forma Normal.

### 3.2.2 - 2ª Forma Normal (2FN)

Para verificar que as relações se encontram na 2ª Forma Normal é necessário verificar que todos os atributos têm uma dependência funcional total em relação à chave primária, eliminando-se assim dependências parciais. Para esta análise, começa-se por analisar as dependências funcionais em cada relação:

#### 3.2.2.1 Relação Livro

**Atributos da relação:** idLivro, Titulo, CodBarras, ISBN, ISSN, CDU, Coleccao

**Dependências funcionais:**

idLivro -> Titulo, CodBarras, ISBN, ISSN, CDU, Coleccao (Chave primária)

CodBarras -> Titulo, ISSN, ISBN, CDU, idLivro, Coleccao (Chave candidata)

ISBN -> Titulo, CodBarras, ISSN, CDU, idLivro, Coleccao (Chave candidata)

ISSN -> Titulo, CodBarras, ISBN, CDU, idLivro, Coleccao (Chave candidata)

Verifica-se que todos os atributos da relação têm uma dependência funcional total em relação à chave primária.

### 3.2.2.2 Relação Autor

**Atributos da relação:** idAutor, PrimeirosNomes, Apelido

**Dependências funcionais:**

idAutor -> PrimeirosNomes, Apelido (Chave primária)

Verifica-se que todos os atributos da relação têm uma dependência funcional total em relação à chave primária.

### 3.2.2.3 Relação Editora

**Atributos da relação:** idEditora, Designacao

**Dependências funcionais:**

idEditora -> Designacao (Chave primária)

Verifica-se que todos os atributos da relação têm uma dependência funcional total em relação à chave primária.

### 3.2.2.4 Relação Coleccao

**Atributos da relação:** idColeccao, Designacao

**Dependências funcionais:**

IdColeccao -> Designacao (Chave primária)

Verifica-se que todos os atributos da relação têm uma dependência funcional total em relação à chave primária.

### 3.2.2.5 Relação Exemplar

**Atributos da relação:** idExemplar, Condicao, Disponibilidade, Localizacao, Livro

**Dependências funcionais:**

idExemplar -> Condicao, Disponibilidade, Localizacao, Livro (Chave primária)

Verifica-se que todos os atributos da relação têm uma dependência funcional total em relação à chave primária.



### 3.2.2.6 Relação Requisicao

**Atributos da relação:** idRequisicao, DataRequisicao, Estado, NroMaxRenovacoes, DataEntrega, NroRenovacoes, Exemplar, Utilizador

**Dependências funcionais:**

idRequisicao -> DataRequisicao, Estado, NroMaxRenovacoes, DataEntrega, NroRenovacoes, Exemplar, Utilizador (Chave primária)

Verifica-se que todos os atributos da relação têm uma dependência funcional total em relação à chave primária.

### 3.2.2.7 Relação Utilizador

**Atributos da relação:** idUser, Nome, Tipo, Email, Telefone, CC, NroMecanografico

**Dependências funcionais:**

idUser -> Nome, Tipo, Email, Telefone, CC, NroMecanografico (Chave primária)

CC -> idUser, Nome, Tipo, Email, Telefone, NroMecanografico (Chave candidata)

NroMecanografico -> idUser, dUser, Nome, Tipo, Email, Telefone, CC (Chave candidata)

Verifica-se que todos os atributos da relação têm uma dependência funcional total em relação à chave primária.

### 3.2.2.8 Relação Localizacao

**Atributos da relação:** idLocal, Piso, Estante, Prateleira

**Dependências funcionais:**

idLocal -> Piso, Estante, Prateleira (Chave primária)

### 3.2.2.9 Relação Livro-Publicado-Editora

**Atributos da relação:** Livro, Editora, Edicao, Ano

**Dependências funcionais:**

Livro, Editora, Edicao -> Ano (Chave primária)

Verifica-se que todos os atributos da relação têm uma dependência funcional total em relação à chave primária.

### **3.2.2.10 Relação Exemplar-Reservado-Utilizador**

**Atributos da relação:** Exemplar, Utilizador, DataReserva, Estado

**Dependências funcionais:**

Exemplar, Utilizador, DataReserva -> Estado (Chave primária)

Todos os atributos de todas as relações dependem funcionalmente (de forma total) da sua chave primária. Anteriormente verificou-se que as relações estavam na primeira forma normal, e desta forma mostrou-se nesta secção que também estão na 2ª forma normal.

### **3.2.3 - 3ª Forma Normal (3FN)**

Nas secções anteriores mostrou-se que as relações estão na 2FN. De acordo com as dependências funcionais apresentadas anteriormente, também se verifica que nenhum dos atributos não chave depende de outro também não chave. Conclui-se por isso que não existem dependências transitivas e por isso o modelo encontra-se na 3FN.

### **3.2.4 Forma Normal Boyce-Codd (FNBC)**

Pelas dependências funcionais analisadas na secção 3.2.2, verifica-se que em cada tabela, todos os determinantes das dependências funcionais são chaves candidatas. Sendo que as relações já se encontravam na 3ª forma normal, conclui-se agora que elas também se encontram na forma normal Boyce-Codd.

### 3.3 Validação segundo Transações

Na secção 2.7 o modelo conceptual foi validado segundo as transações. Foram escolhidas algumas transações que se consideraram relevantes e foi analisado de que forma o modelo conceptual lhes dava resposta. Nesta secção é validado o modelo lógico através dessas mesmas transações, desenhando sobre o esquema lógico o respectivo mapa de transações.

#### 3.3.1 Qual(is) a(s) localização(ões) (pisos, estante e prateleira) dos exemplares de um livro com determinado título?

Na tabela Livro é possível verificar os livros têm um determinado título. Com a chave primária dos livros com esse título é possível saber os seus exemplares através da chave estrangeira da tabela Exemplar que referencia a tabela Livro. A tabela Exemplar por sua vez tem uma chave estrangeira para a localização, sendo assim possível saber as localizações dos exemplares.

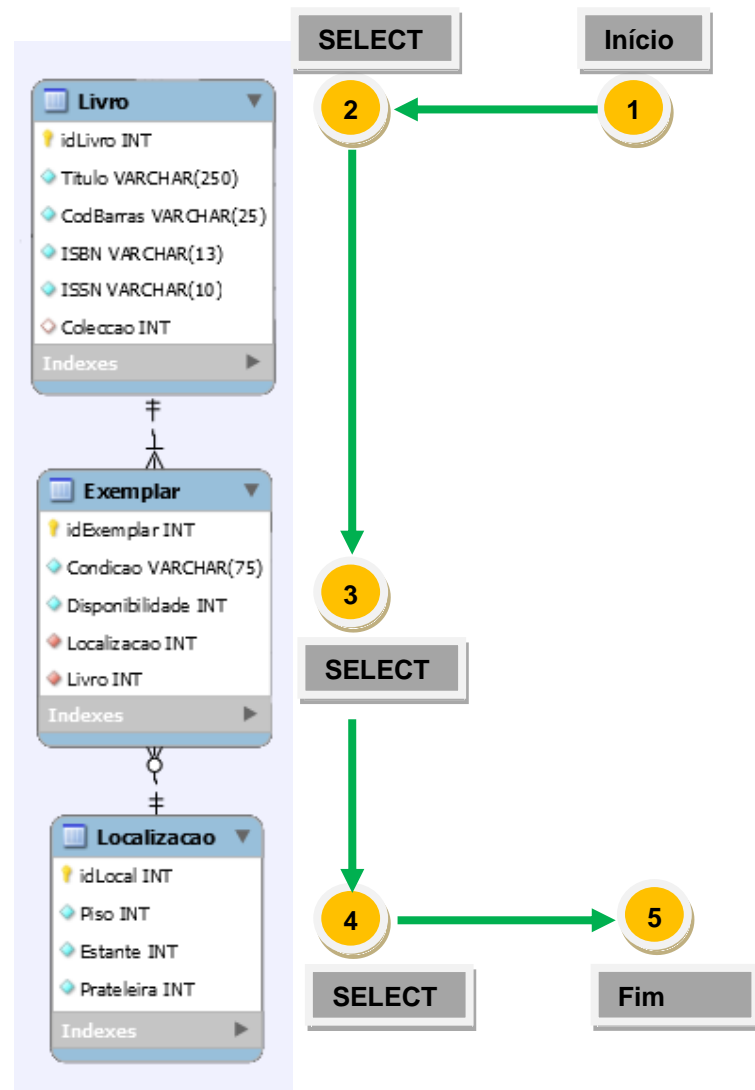


Figura 18 – Mapa de Transação – Consultar localização de exemplar por título

### 3.3.2 - Efetuar uma requisição

Antes de uma requisição poder ser registada é necessário ver se esta pode ser feita, o que implica verificar se o exemplar não possui reservas por parte de outros utilizadores, que pode ser verificado através da chave estrangeira para Utilizador na tabela “Exemplar-reservado-Utilizador”, consultando o seu atributo “Estado”. É necessário verificar ainda que o exemplar não é apenas de consulta, o que pode ser verificado na entidade Exemplar, no atributo “Disponibilidade”. Caso o exemplar esteja disponível para requisição, a informação relativa à requisição é inserida na tabela “Requisicao”. A tabela Exemplar contém o estado de disponibilidade (atributo “Disponibilidade”) de cada exemplar, que terá que ser actualizado depois de feita a requisição. É ainda necessário verificar se a requisição corresponde ao levantamento de uma reserva, e actualizar o atributo “Estado” da tabela “Exemplar-reservado-Utilizador” em caso afirmativo.

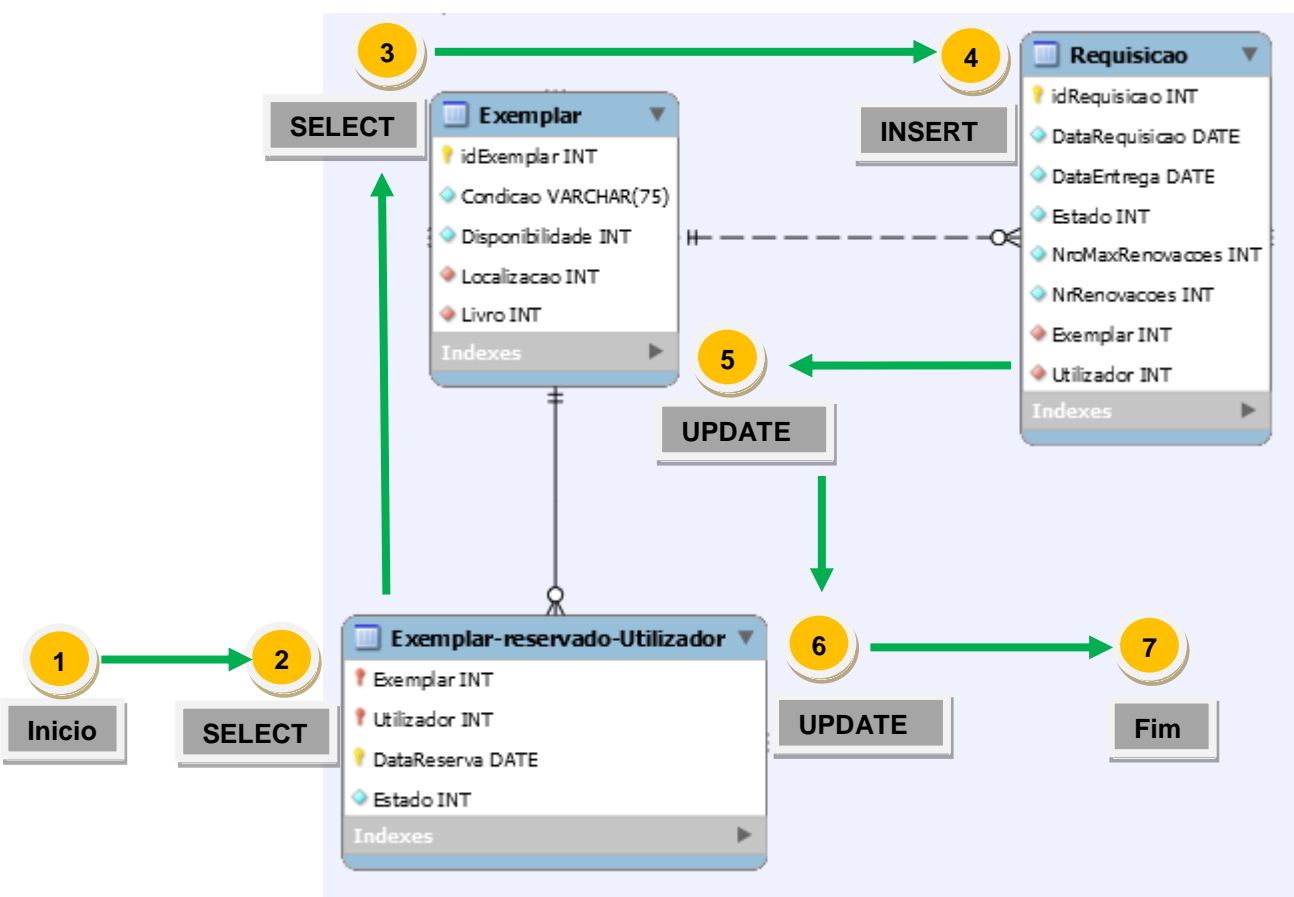


Figura 19 - Mapa de transação – Efectuar requisição

### 3.3.3 - Efectuar Reserva

Antes de um utilizador poder efectuar uma reserva é necessário ver se esta pode ser feita, o que implica verificar se o exemplar pode ser requisitado, que pode ser visto na tabela “Exemplar” no atributo “Disponibilidade”. É ainda necessário verificar se o exemplar não possui reservas por parte de outros utilizadores, uma vez que isso vai afectar o estado da reserva. Caso a reserva possa ser efectuada, a informação da mesma é registada na tabela “Exemplar-reservado-Utilizador”.

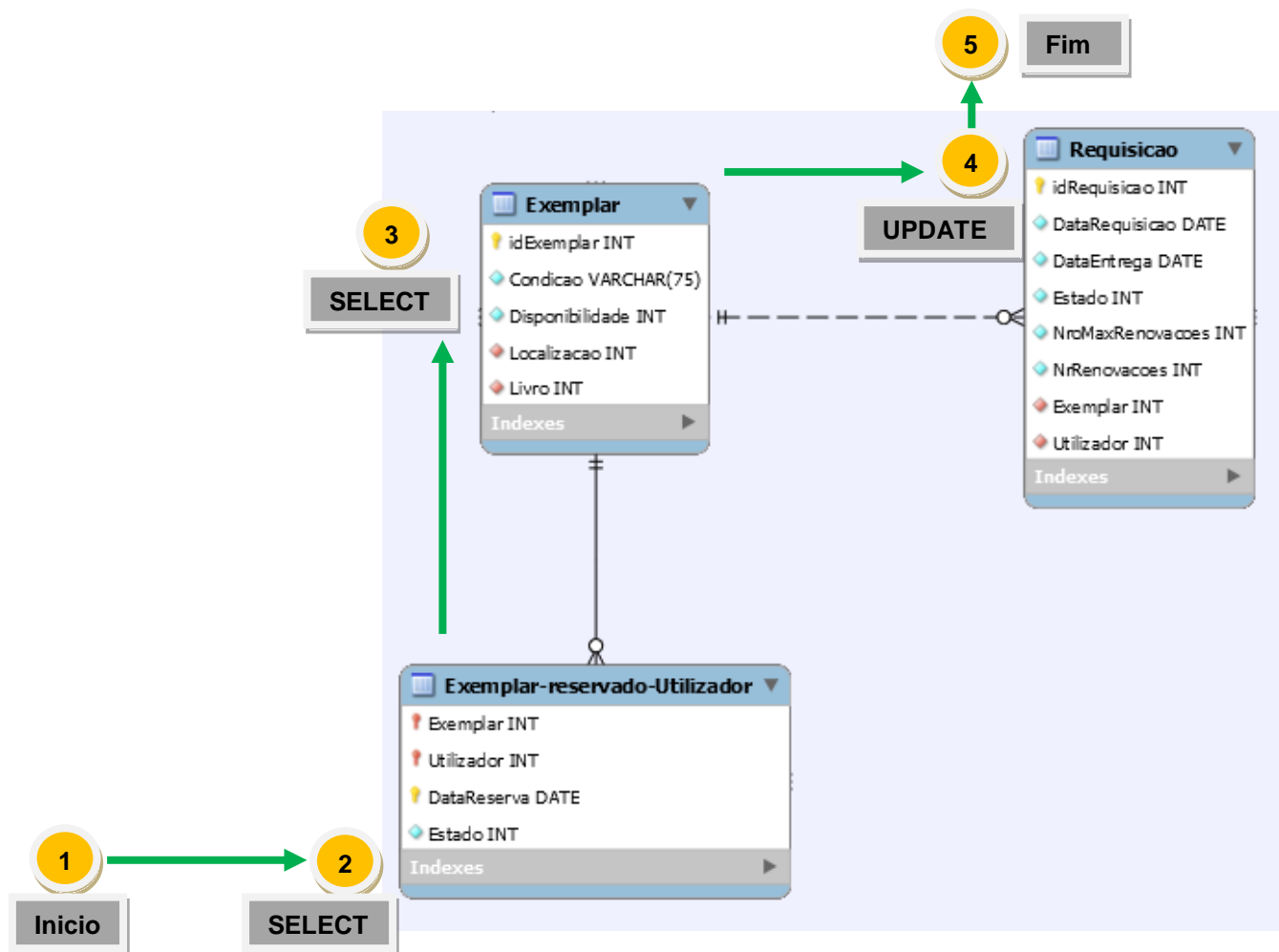


Figura 20 - Mapa de transação - Requisitar Exemplar reservado

## 3.4 - Verificação das Restrições de Integridade

Para garantir que a base de dados contém apenas dados válidos, ou seja, exatos e consistentes (de acordo com os requisitos), são impostas restrições de integridade sobre a mesma. Assim, nesta secção são analisadas restrições de domínio, regras de integridade como integridade de entidade e referencial, restrições de multiplicidade e restrições gerais aos dados.

### 3.4.1 - Integridade de Domínio

As regras de integridade de domínio dizem respeito aos valores inseridos na base de dados. Estas analisam o tipo de dados que corresponde a cada atributo, se estes podem ou não ser nulos e qual o conjunto de valores que podem tomar. Aquando da criação do dicionário de dados, estas regras já foram impostas, e podem ser verificadas no dicionário de dados dos atributos, no [anexo III](#). Atributos que podem assumir valores nulos são representados no esquema lógico por um losango vazio, sem preenchimento. Uma vez que a chave para a Coleção, na tabela Livro, é o único atributo que suporta nulo (um livro não tem obrigatoriamente que pertencer a uma coleção), não será apresentado aqui o esquema lógico para validar esta restrição, podendo este ser consultado na figura 17.

Os valores permitidos para cada atributo foram também identificados no momento da criação do dicionário de dados. Os atributos que estão restringidos por valores são, para a respetiva tabela:

- O atributo Disponibilidade, na tabela Exemplar, foi definido como um inteiro que apenas conhece três valores:
  - 0 - caso o exemplar não seja requisitável;
  - 1 - se já se encontrar requisitado;
  - 2 - caso esteja disponível.
- O atributo Estado na tabela Requisição foi definido como um inteiro que suporta dois valores:
  - 0 – caso a requisição esteja ativa, ou seja, o exemplar encontra-se com o utilizador;
  - 1 – caso o exemplar já tenha sido entregue.
- O atributo Tipo na tabela Utilizador é representado por dois caracteres que podem tomar os seguintes valores:
  - A – caso o utilizador seja um aluno de licenciatura;
  - PG/ID – caso o utilizador seja aluno de pós graduação;
  - D – se o utilizador é um docente;
  - F – se o utilizador é um funcionário;

LE – se o utilizador é um leitor externo, não estando por isso registado na universidade;

X – caso o utilizador seja aluno de Erasmus.

- O atributo Estado na tabela resultante do relacionamento Exemplar – Utilizador (Exemplar-reservado-Utilizador), é um atributo do tipo inteiro que suporta três valores:
  - 0 – caso tenha sido efetuada uma reserva sobre o exemplar;
  - 1 – se o exemplar já estiver pronto a levantar;
  - 2 – caso o exemplar já tenha sido levantado após o pedido de reserva;
  - 3 – caso a reserva tenha sido cancelada.

### 3.4.2 - Integridade de Entidade

A integridade de entidade dita que nenhum valor de chave primária pode ser nulo, nem podem existir valores de chaves primárias iguais. Esta restrição é garantidamente cumprida quando ao declarar um atributo como chave primária da tabela, o SGBD não permite que a tabela tenha dois tuplos com o mesmo valor nesse atributo, nem que o atributo chave seja nulo.

### 3.4.3 - Integridade Referencial

A integridade referencial permite garantir que todas as chaves estrangeiras são válidas. Uma chave estrangeira de uma tabela tem que coincidir com uma chave primária da sua tabela "pai" a que a chave estrangeira se refere. Sendo assim, não existem referências a valores não existentes e se um valor chave é alterado, todas as referências a ele são consistentemente alteradas em toda a base de dados.

### 3.4.4 – Restrições de Multiplicidade

Estas restrições foram impostas quando foram decididos os relacionamentos entre as entidades. Podem ser verificadas no dicionário de dados, na tabela dos relacionamentos ([tabela 2](#)).

### 3.4.5 – Restrições Gerais

Foram consideradas as seguintes restrições:

- O número de renovações de requisição de um exemplar (atributo NroRenovacoes) não pode exceder o número máximo de renovações (atributo NroMaxRenovacoes)
- Um utilizador não pode efectuar requisição se o livro não for requisitável ou tiver reservas por parte de outros utilizadores
- Uma reserva só pode ser efetuada se o exemplar for requisitável.

## 3.5 Validação de modelo lógico com utilizador

Terminada a fase de modelação lógica, o modelo foi revisto com o utilizador e confirmou-se que garantia os requisitos. Esta validação foi feita com recurso a uma nova pequena entrevista às funcionárias da biblioteca e nova verificação cuidada do modelo.

## 3.6 Tamanho inicial e Crescimento Futuro

Nesta secção apresenta-se uma estimativa para o tamanho inicial da base de dados e o seu crescimento. Esta estimativa foi obtida a partir do conhecimento obtido enquanto utentes do mesmo, complementada com alguma pesquisa adicional.

### 3.6.1 Tamanho inicial

Sobre o tamanho inicial da base de dados, considerou-se o seguinte estado inicial:

**30.000 exemplares.** Este valor baseia-se no número de pisos e numa estimativa para o número de exemplares em cada estante de cada piso.

**5.000 livros.** Este valor foi obtido a partir do número de exemplares. É frequente ver na biblioteca livros com 6 exemplares. Alguns livros têm mais e outros menos, no entanto é razoável admitir uma média de 6 exemplares por livro, obtendo-se assim  $30.000/6=5.000$  livros

**15 coleções.** Este valor foi o mais difícil de estimar devido à dificuldade de haver alguma forma de o fazer. Dado que por observação do sistema, bastantes livros pertencem a uma coleção assumiu-se 15 coleções como um valor razoável o que perfaz cerca de 333 livros por coleção.

**3.000 utilizadores.** Esta estimativa baseia-se no número de alunos, docentes e funcionários da universidade, contando também com a possibilidade de uma pequena minoria serem utilizadores externos. Este número baseia-se apenas em utilizadores com ligação ativa à universidade neste momento. Excluem-se assim do registo da base de dados ex-alunos, ex-funcionários e ex-docentes, que deverão efetuar novo registo se pretenderem usar o sistema.

**1950 localizações.** Uma localização corresponde ao piso, estante e prateleira. A biblioteca tem 3 pisos. Cada piso tem cerca de 130 estantes e cada estante tem 5 prateleiras. Assim tem-se  $10*13*5*3=1950$  localizações.

**3.000 autores.** Foi difícil encontrar informação sobre o número de autores, no entanto por observação do sistema verificou-se que a diversidade de autores era grande, mas que havia uma parte pequena dos livros que eram escritos por vários autores. Estimou-se por isso que este valor seria próximo do número de livros, mas ligeiramente inferior, pelo que se assumiu 3.000 autores.



**1.000 editoras.** A diversidade de editoras é menor que a diversidade de autores, pelo que se considerou um número inferior de editoras.

**300 requisições.** Dada a dificuldade de estimar o total de requisições feitas no passado, este valor corresponde a uma estimativa do número de requisições ativas presentes no sistema.

**10 reservas.** Dada a dificuldade de estimar o total de reservas feitas no passado, este valor corresponde a uma estimativa do número de reservas ativas presentes no sistema.

### 3.6.2 Crescimento futuro

**150 exemplares e 10 livros por ano.** A taxa de renovação de livros na biblioteca é bastante baixa, tratando-se em grande parte de renovação de exemplares em mau estado de conservação. Os livros novos resultam em grande parte de doações, que também não são abundantes.

**1 coleção por ano.** Associada à baixa taxa de renovação de livros está o número de coleções adicionadas por ano. Novos livros tendem a ser inseridos em coleções já existentes.

**3.000 utilizadores por ano.** Esta estimativa baseia-se sobretudo no número de alunos (que constituem a maioria dos utilizadores) a entrar para a universidade em cada ano.

**3 autores e editoras por ano.** O número baixo de autores e editoras deve-se também à baixa taxa de renovação de livros por ano referida anteriormente.

**50 requisições por dia.** A operação mais usada no sistema será a de requisição de livros, o que faz da tabela de requisições a mais crítica do sistema.

**10 reservas por dia.** A par das requisições, também as reservas serão a funcionalidade mais solicitada do sistema.

O espaço físico da biblioteca mantém-se praticamente constante, pelo que se considera o número de localizações invariante. Com esta análise, conclui-se que as requisições e reservas serão o principal factor de aumento de espaço da base de dados.

Relativamente ao crescimento futuro em termos do modelo, não se prevêem alterações, visto o sistema de base de dados da BGUM ter regras de negócio que se mantêm constantes.

## 4. Modelo físico

Neste capítulo é demonstrada a forma como partindo do modelo lógico apresentado no capítulo anterior, foi conseguida a sua implementação física.

### 4.1 Tradução do modelo lógico para um SGBD e consequente implementação

O primeiro passo para traduzir o modelo lógico num modelo físico foi escolher o SGBD que melhor se adequava às funcionalidades requeridas pelo sistema a implementar. Por uma questão de comodidade foi escolhido o MySQL, um sistema grátis e fácil de instalar, aceder e dar manutenção, que já tinha sido anteriormente introduzido durante as aulas da UC de Base de Dados. A linguagem do MySQL é de fácil compreensão, tornando-o por isso num sistema que permite implementações simples. Como é uma solução relativamente popular, existe toda uma comunidade que deu resposta a obstáculos que apareceram durante o processo de implementação. O MySQL também tem algumas limitações, mas dada a dimensão do problema, suporta bem as operações necessárias à gestão do mesmo. Questões de segurança e fortes mecanismos de proteção de dados foram também um dos motivos para a utilização do MySQL. Em termos de inserção de dados, este SGBD permite a criação de chaves primárias únicas, forçando a que os valores de índice sejam sempre distintos, e por isso leva a rapidez nas pesquisas e controlo de integridade. Permite também a especificação de chaves estrangeiras e domínios de atributos, e oferece suporte a transações, *triggers*, vistas, cursores, *stored procedures*, entre outros.

#### 4.1.1 Relações Base

Nesta fase decidiu-se como representar as tabelas e a informação nelas contida durante a implementação do esquema físico. Visto o modelo lógico ter sido esquematizado no Workbench do MySQL, este processo foi automático, pois durante a criação das tabelas do lógico a informação sobre as entidades, domínio dos atributos, definição de chaves primária e estrangeiras foi representada em tabelas, e estas tabelas servem de suporte ao esquema físico.

A informação corresponde àquela no dicionário de dados, e a definição de cada uma das tabelas do físico consta em seguida.

#### Legenda:

**PK** – Primary Key

**NN** – Not Null

**UQ** – Unique

**B** – Binary

**UN** – Unsigned

**ZF** – Zero-Filed

**AI** – Auto Incremental

**G** – Generated Column

**Default/Expression** – Valor por defeito

#### • Livro

Column Name	Datatype	PK	NN	UQ	B	UN	ZF	AI	G	Default/Expression
idLivro	INT	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
Titulo	VARCHAR(250)	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
CodBarras	VARCHAR(25)	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
ISBN	VARCHAR(13)	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
ISSN	VARCHAR(10)	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
Coleccao	INT	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	

Figura 21 - Tabela representativa da entidade Livro

#### • CDU

Column Name	Datatype	PK	NN	UQ	B	UN	ZF	AI	G	Default/Expression
CDU	VARCHAR(45)	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
Livro	INT	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	

Figura 22 - Tabela representativa do atributo multivalor CDU

#### • Coleccao

Column Name	Datatype	PK	NN	UQ	B	UN	ZF	AI	G	Default/Expression
idColeccao	INT	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
Designacao	VARCHAR(75)	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	

Figura 23 - Tabela representativa da entidade Coleccao

- **Autor**

Column Name	Datatype	PK	NN	UQ	B	UN	ZF	AI	G	Default/Expression
idAutor	INT	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
PrimeirosNomes	VARCHAR(75)	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
Apelido	VARCHAR(25)	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	

Figura 24 - Tabela representativa da entidade Autor

- **Autor-escreve-Livro**

Column Name	Datatype	PK	NN	UQ	B	UN	ZF	AI	G	Default/Expression
Livro	INT	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
Autor	INT	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	

Figura 25 - Tabela representativa da derivação do relacionamento Livro – Autor

- **Editora**

Column Name	Datatype	PK	NN	UQ	B	UN	ZF	AI	G	Default/Expression
idEditora	INT	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
Designacao	VARCHAR(75)	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	

Figura 26 - Tabela representativa da entidade Editora

- **Livro-publicado-Editora**

Column Name	Datatype	PK	NN	UQ	B	UN	ZF	AI	G	Default/Expression
Livro	INT	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
Editora	INT	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
Edicao	INT	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
Ano	YEAR	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	

Figura 27 - Tabela representativa da derivação do relacionamento Livro – Editora

- **Exemplar**

Column Name	Datatype	PK	NN	UQ	B	UN	ZF	AI	G	Default/Expression
idExemplar	INT	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
Condicao	VARCHAR(75)	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
Disponibilidade	INT	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
Localizacao	INT	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
Livro	INT	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	

Figura 28 - Tabela representativa da entidade Exemplar

- **Localizacao**

Column Name	Datatype	PK	NN	UQ	B	UN	ZF	AI	G	Default/Expression
idLocal	INT	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
Piso	INT	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
Estante	INT	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
Prateleira	INT	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	

Figura 29 - Tabela representativa da entidade Localizacao

- **Utilizador**

Column Name	Datatype	PK	NN	UQ	B	UN	ZF	AI	G	Default/Expression
idUser	INT	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
Tipo	VARCHAR(2)	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
Nome	VARCHAR(75)	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
Email	VARCHAR(75)	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
CC	VARCHAR(10)	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
NroMecanografico	VARCHAR(10)	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
Telefone	VARCHAR(10)	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	

Figura 30 - Tabela representativa da entidade Utilizador

- **Exemplar-reservado-Utilizador**

Column Name	Datatype	PK	NN	UQ	B	UN	ZF	AI	G	Default/Expression
Exemplar	INT	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
Utilizador	INT	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
DataReserva	DATE	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
Estado	INT	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	

Figura 31 - Tabela representativa da derivação do relacionamento Exemplar – Utilizador

- **Requisicao**

Column Name	Datatype	PK	NN	UQ	B	UN	ZF	AI	G	Default/Expression
idRequisicao	INT	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
DataRequisicao	DATE	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
DataEntrega	DATE	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
Estado	INT	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
NroMaxRenovacoes	INT	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	6
NrRenovacoes	INT	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
Exemplar	INT	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
Utilizador	INT	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	

Figura 32 - Tabela representativa da entidade Requisicao

## 4.1.2 Representação de Atributos Derivados

Nesta fase é descrita a forma como os atributos derivados foram representados aquando a implementação física do esquema lógico. Dado nenhum ter sido definido, tal documentação não é expressa.

## 4.1.3 Restrições Gerais

As restrições do modelo foram implementadas por via de métodos, dado o suporte limitado do *mysql* à instrução CHECK da linguagem SQL, que impede que algumas das verificações apresentadas pudessem estar associadas às tabelas propriamente ditas.

- **Utilizador não pode exceder máximo de renovações permitidas**

Esta restrição foi garantida no método de renovação de requisição, através do seguinte código:

```
SELECT NrRenovacoes, NroMaxRenovacoes INTO v_numRenovacoes, v_numMaxRenovacoes
      FROM requisicao
      WHERE idRequisicao = p_idRequisicao;
```

```
IF v_numRenovacoes = v_numMaxRenovacoes THEN SET mustRollback = 1; END IF;
```

- **Utilizador não pode efectuar requisição se o livro não for requisitável ou tiver reservas por parte de outros utilizadores**

Esta restrição foi garantida no método de requisição de um exemplar, através do seguinte código:

```
SELECT E.Disponibilidade INTO v_disponibilidadeExemplar
      FROM Exemplar E
      WHERE E.idExemplar = p_IdExemplar;
```

```
IF v_disponibilidadeExemplar != 2 THEN SET mustRollback = 1; END IF;
```

```
SELECT COUNT(Exemplar) INTO v_numeroReservasOutrosUtilizadores
      FROM `exemplar-reservado-utilizador` ERU
      WHERE Exemplar = p_IdExemplar AND Utilizador != p_Utilizador AND Estado = 1;
```

```
IF v_numeroReservasOutrosUtilizadores > 0 THEN SET mustRollback = 1; END IF;
```

- **Utilizador apenas pode efectuar reserva se o exemplar for requisitável**

Esta restrição foi garantida no método de reserva de um exemplar, através do seguinte código:

```
SELECT E.Disponibilidade INTO v_disponibilidade
      FROM exemplar E
      WHERE E.idExemplar = p_idExemplar;
```

```

SELECT COUNT(Exemplar) INTO v_numReservas
      FROM `exemplar-reservado-utilizador`
      WHERE Exemplar = p_idExemplar AND (Estado = 0 OR Estado = 1);

IF v_disponibilidade = 0 THEN SET mustRollback = 1; END IF;
IF v_disponibilidade = 1 OR v_numReservas > 0 THEN SET v_estadoAInserir = 0; END
IF;
IF v_disponibilidade = 2 AND v_numReservas = 0 THEN SET v_estadoAInserir = 1; END
IF;

```

## 4.2 Escolha de Índices

Considerou-se que os índices para as chaves primárias e estrangeiras são suficientes para garantir alguma eficiência nas queries.

## 4.3 Análise de Transações

Nesta secção analisa-se de que forma as transações analisadas no na validação do modelo conceptual e lógico foram implementadas usando SQL.

### 4.3.1 Qual(is) a(s) localização(ões) (pisso, estante e prateleira) dos exemplares de um livro com determinado título?

Para resposta a esta transação foi criado o método `sp_localizacao_exemplares()` que pode ser visto de seguida:

```

CREATE PROCEDURE `sp_localizacao_exemplares`(IN p_titulo VARCHAR(250))
BEGIN
SELECT DISTINCT Loc.*
      FROM (SELECT *
            FROM livro
            WHERE Titulo = p_titulo) L, Exemplar E, localizacao Loc
      WHERE L.idLivro = E.Livro AND E.Localizacao = Loc.idLocal;
END

```

### 4.3.2 Efectuar requisição

Para dar resposta a esta transação foi criado um método com a seguinte assinatura:

```
CREATE PROCEDURE sp_efectuar_requisicao(IN p_IdExemplar INT, IN p_Utilizador INT,  
IN p_DataRequisicao DATE, IN p_DataEntrega DATE, IN p_NrMaxRenovacoes INT)
```

Visto se tratar de um método algo complexo, o código será apresentado de forma faseada com respectivos comentários. Em primeiro lugar, visto o método se tratar de uma transação, além das variáveis “normais” inicializa-se também um handler para controlar a transação:

```
DECLARE mustRollback INT DEFAULT 0;  
DECLARE v_numeroReservasOutrosUtilizadores INT DEFAULT 0;  
DECLARE v_disponibilidadeExemplar INT;  
DECLARE Erro BOOL DEFAULT 0;  
DECLARE CONTINUE HANDLER FOR SQLEXCEPTION SET Erro = 1;  
  
SET autocommit = 0;  
SET SQL_SAFE_UPDATES = 0;  
  
START TRANSACTION;
```

Primeiro, é necessário verificar se o exemplar esta disponível para reserva e ativar a flag *mustRollback* caso isso não se verifique:

```
SELECT E.Disponibilidade INTO v_disponibilidadeExemplar  
FROM Exemplar E  
WHERE E.idExemplar = p_IdExemplar;  
  
IF v_disponibilidadeExemplar != 2 THEN SET mustRollback = 1; END IF;
```

É ainda necessário assegurar que nenhum utilizador efectuou reserva sobre o mesmo exemplar:

```
SELECT COUNT(Exemplar) INTO v_numeroReservasOutrosUtilizadores  
FROM `exemplar-reservado-utilizador` ERU  
WHERE Exemplar = p_IdExemplar AND Utilizador != p_Utilizador AND Estado = 1;  
  
IF v_numeroReservasOutrosUtilizadores > 0 OR Erro THEN SET mustRollback = 1; END  
IF;
```

Caso a requisição tenha sido feita a partir de uma reserva, actualizar o estado da reserva para “livro entregue”:

```
UPDATE `exemplar-reservado-utilizador`  
SET Estado = 2  
WHERE Estado = 1 AND Exemplar = p_IdExemplar AND Utilizador = p_Utilizador;
```



Por último, actualizar o estado do exemplar para “requisitado”:

```
UPDATE exemplar
    SET Disponibilidade = 1
    WHERE idExemplar = p_IdExemplar;

IF Erro OR mustRollback = 1 THEN ROLLBACK; ELSE COMMIT; END IF;
SET SQL_SAFE_UPDATES = 1;
END;
```

### 4.3.3 Efectuar uma reserva

Para dar resposta a esta transação foi criado um método com a seguinte assinatura:

```
CREATE PROCEDURE sp_efectuar_reserva(IN p_idExemplar INT, IN p_idUtilizador INT,
IN p_DataReserva DATE)
```

Visto se tratar de um método algo complexo, o código será apresentado de forma faseada com respectivos comentários. Em primeiro lugar, visto o método se tratar de uma transação, além das variáveis “normais” inicializa-se também um handler para controlar a transação:

```
DECLARE mustRollback INT DEFAULT 0;
DECLARE v_numReservas INT DEFAULT 0;
DECLARE v_disponibilidade INT;
DECLARE v_estadoAInserir INT;
DECLARE Erro BOOL DEFAULT 0;
DECLARE CONTINUE HANDLER FOR SQLEXCEPTION SET Erro = 1;

SET autocommit = 0;
SET SQL_SAFE_UPDATES = 0;
START TRANSACTION;
```

Começa-se por verificar se o exemplar é requisitável e quantas reservas existem sobre o mesmo:

```
SELECT E.Disponibilidade INTO v_disponibilidade
    FROM exemplar E
    WHERE E.idExemplar = p_idExemplar;

SELECT COUNT(Exemplar) INTO v_numReservas
    FROM `exemplar-reservado-utilizador`
    WHERE Exemplar = p_idExemplar AND (Estado = 0 OR Estado = 1);
```

Caso o exemplar não seja sequer requisitável, a variável *mustRollback* toma o valor 1. Caso alguém tenha o exemplar requisitado actualmente ou já existam reservas sobre o mesmo, definir estado da reserva a adicionar como “reserva pedida”. Se o exemplar estiver disponível e

não houver reservas sobre ele, definir estado da reserva a adicionar como “livro pronto a levantar”:

```
IF v_disponibilidade = 0 THEN SET mustRollback = 1; END IF;
IF v_disponibilidade = 1 OR v_numReservas > 0 THEN SET v_estadoAInserir = 0; END
IF;
IF v_disponibilidade = 2 AND v_numReservas = 0 THEN SET v_estadoAInserir = 1; END
IF;
```

Por fim, inserir a reserva:

```
INSERT INTO `exemplar-reservado-utilizador`
(Exemplar, Utilizador, DataReserva, Estado)
VALUES
(p_idExemplar, p_idUtilizador, p_DataReserva, v_estadoAInserir);

IF Erro OR mustRollback = 1 THEN ROLLBACK; ELSE COMMIT; END IF;
SET SQL_SAFE_UPDATES = 1;
END;
```

## 4.4 Estimativa dos Requisitos de Espaço em Disco

### 4.4.2 Povoamento e tamanho inicial

Para melhor análise dos requisitos em termos de espaço foi efectuado um povoamento da base de dados de tamanho semelhante ao indicado anteriormente na secção 2.5. Desta forma é possível ter uma ideia do espaço inicial da base de dados. Depois de efectuado o povoamento, executou-se a seguinte query SQL para conhecer os tamanhos das tabelas:

```
SELECT TABLE_NAME 'Nome Tabela', table_rows 'Nº Registos',
round(data_length/1024,2) 'Dados (KB)', round(data_length/1024
/table_rows,2) 'KB/Registo', round(index_length/1024,2) 'Indices (KB)',
round(((data_length + index_length)/1024/1024),3) 'Total MB'
FROM information_schema.TABLES WHERE table_schema = 'bgum';
```

Cujos resultados se apresentam de seguida:

	Nome Tabela	Nº Registos	Dados (KB)	KB/Registo	Indices (KB)	Total MB
▶	exemplar	19620	1552.00	0.08	768.00	2.266
	livro-publicado-editora	10089	368.00	0.04	464.00	0.813
	livro	4966	528.00	0.11	144.00	0.656
	autor-escreve-livro	5605	304.00	0.05	288.00	0.578
	cdu	5570	288.00	0.05	176.00	0.453
	utilizador	3000	288.00	0.10	0.00	0.281
	autor	3131	160.00	0.05	0.00	0.156
	localizacao	1950	96.00	0.05	0.00	0.094
	editora	1083	80.00	0.07	0.00	0.078
	exemplar-reservado-utilizador	8	16.00	2.00	32.00	0.047
	requisicao	300	16.00	0.05	32.00	0.047
	coleccao	15	16.00	1.07	0.00	0.016

Figura 33 – Nº de registos e tamanho inicial das tabelas

De seguida foi efectuada a seguinte querie para saber o espaço total ocupado pela base de dados:

```
SELECT table_schema 'DB Name', SUM(data_length + index_length) / 1024 /
1024 'DB Size in MB'
FROM information_schema.tables
WHERE table_schema = 'bgum';
```

Cujo resultado se apresenta de seguida:

	DB Name	DB Size in MB
▶	bgum	5.48437500

Conclui-se assim que a base de dados necessita de cerca de 5.5 MB inicialmente.

### 4.4.3 Crescimento Futuro

Tal como referido na secção 3.5, as requisições e reservas serão o principal factor de aumento do espaço em disco da base de dados. Por dia há cerca de 10 reservas. Ao fim de um ano há  $10 \times 365 = 3650$  reservas. Há 50 requisições por dia, o que ao fim de um ano perfaz  $50 \times 65 = 18.250$  requisições. Por ano há ainda cerca de 3.000 novos utilizadores. Usando os dados da figura 33 pode-se então prever o aumento aproximado da base de dados ao fim de um ano:

$3650 \times 2 + 18.250 \times 0,05 + 3000 \times 0,1 = 8512.5 \text{ KB} = 8.3 \text{ MB}$ .

## 4.5 Definição das Vistas de Utilização e Regras de Acesso

Não foram identificadas vistas de utilizador durante a recolha de requisitos ou durante o desenvolvimento do modelo conceptual. No entanto, poderiam ser considerados dois tipos de utilizador da base de dados, cada um com as correspondentes vistas. Esses utilizadores corresponderiam aos funcionários e aos utentes da biblioteca. Os funcionários teriam todas as permissões, exceto a manipulação (inserção, remoção ou alteração) de utilizadores da base de dados, estando essa função a cargo dos Serviços Académicos. Os utentes da biblioteca apenas poderiam efetuar consultas sobre livros, e as restantes entidades com eles relacionadas. Sabendo o seu próprio código de utilizador, os utentes também poderiam consultar informações sobre eles mesmos. Na tabela abaixo estão especificadas as permissões de cada utilizador:

Perfil de Utilizador/Tabela	Funcionário				Utilizador			
	I	U	D	S	I	U	D	S
Livro	X	X	X	X				X
CDU	X	X	X	X				X
Coleccao	X	X	X	X				X
Autor	X	X	X	X				X
Autor-escreve-Livro	X	X	X	X				X
Editora	X	X	X	X				X
Livro-publicado-Editora	X	X	X	X				X
Exemplar	X	X	X	X				X
Localizacao	X	X	X	X				X
Utilizador				X				
Exemplar-reservado-Utilizador	X	X	X	X				
Requisicao	X	X	X	X				

Tabela 1 - Permissões de cada tipo de utilizador

(I – Insert, U – Update, D- Delete, S - Select)

Para simular estas permissões, foram realizadas as seguintes queries:

```
CREATE VIEW vwUserLivro AS
SELECT
    Titulo AS "Título",
    A.PrimeirosNomes AS "Nome Autor",
    A.apelido AS "Apelido",
    C.Designacao AS "Coleção",
    ED.Designacao AS "Editora",
    LE.Edicao AS "Número da Edição",
    LE.Ano AS "Ano de Edição",
    COUNT(E.livro) AS "Número de Exemplares",
    CodBarras AS "Código de Barras",
    ISBN,
    ISSN
FROM livro AS L INNER JOIN coleccao AS C
    ON L.Coleccao = C.idColeccao
LEFT JOIN exemplar as E
    ON (L.idlivro = E.livro)
    LEFT OUTER JOIN `autor-escreve-livro` AS AL
        ON L.idLivro = AL.livro
    INNER JOIN autor AS A
        ON A.idAutor = AL.autor
    LEFT JOIN `livro-publicado-editora` as LE
        ON (L.idlivro = LE.livro)
        INNER JOIN editora as ED
            ON ED.idEditora = LE.editora
GROUP BY L.titulo;
```

A vista 'vwUserLivro' simula a visão de um utente, que assim tem acesso a toda a informação referente aos livros, inclusive o número de exemplares de cada um.

```

CREATE VIEW vwExemplares AS
SELECT
    L.Titulo AS "Título",
    condicao AS "Condição",
    Loc.Piso,
    Loc.Estante,
    Loc.Prateleira
FROM exemplar AS E INNER JOIN localizacao AS Loc
    ON E.localizacao = Loc.idLocal
    INNER JOIN livro AS L
        ON L.idLivro = E.Livro
GROUP BY L.titulo;

```

A vista 'vwExemplares' permite aos utentes da biblioteca acesso a informações sobre o exemplar de cada livro, como o seu estado de conservação e localização.

No caso dos funcionários, para garantir que estes não têm permissões para manipular os utilizadores, bastaria apenas retirar-lhes esses privilégios:

- Exemplo da criação de um Funcionário:

```

CREATE USER 'Funcionario'@'localhost'
IDENTIFIED BY 'bgum2016';

```

- Concedem-se todos os privilégios:

```

GRANT ALL PRIVILEGES
ON bgum.*
TO 'Funcionario'@'localhost';

```

- E retiram-se os privilégios de inserção, alteração e remoção na tabela Livro :

```

REVOKE INSERT, UPDATE, DELETE
ON bgum.livro
FROM 'Funcionario'@'localhost';

```

## 5. Conclusões e Trabalho Futuro

O balanço global que se faz do trabalho efetuado é positivo. Todos os passos da metodologia de desenho de uma base de dados foram seguidos com sucesso, desde a modelação conceptual até à modelação física e sua implementação num SGBD.

Nesta segunda fase do trabalho, o foco principal foi a modelação lógica e física. No entanto, antes de realizar esta segunda fase, foram efetuadas algumas melhorias ao modelo conceptual entregue na primeira fase. Os requisitos foram reavaliados e reescritos de forma mais clara. Esta reavaliação dos requisitos levou a algumas mudanças no modelo conceptual, das quais se destaca a passagem do atributo CDU para a entidade Livro (anteriormente estava associado à entidade Exemplar) e a remoção do atributo Ano no relacionamento entre Livro e Autor, pois verificou-se que não existiam requisitos para esse atributo.

A tradução do modelo conceptual para o modelo lógico revelou algumas pequenas falhas no modelo conceptual e avaliação de alguns aspetos que não tinham sido considerados na modelação conceptual. Uma dessas falhas foi a necessidade de se ter que colocar o atributo DataReserva como parte da chave primária no relacionamento correspondente à reserva de um exemplar. Esta modificação garante que os requisitos actuais são cumpridos, no entanto deixa a sugestão que a reserva porventura poderia ser na verdade uma entidade, com identificador único, em vez de estar apenas expressa por um relacionamento. Aparte desta falha, a conversão para o modelo lógico e consequente validação foi bem conseguida.

Com o modelo lógico validado e todos os requisitos revistos, a modelação física ocorreu sem dificuldades de maior.

## Bibliografia

Connolly, T. & Begg, C., 2005. *Database Systems - A practical approach to design, implementation and management*. s.l.:Pearson Education Limited.



## Lista de Siglas e Acrónimos

<b>BD</b>	Base de Dados
<b>CDU</b>	Classificação Decimal Unitária
<b>CC</b>	Cartão Cidadão
<b>BGUM</b>	Biblioteca Geral da Universidade do Minho
<b>ISSN</b>	International Standard Serial Number
<b>ISBN</b>	International Standard Book Number
<b>SGBD</b>	Sistema Gestor Base de Dados
<b>SQL</b>	Structured Query Language

## **Anexos**

# I. Dicionário de Dados – Entidades

Tabela 2 - Tabela das Entidades

Entidade	Descrição	Alias	Ocorrência
<b>Livro</b>	Obra disponível na BGUM para requisição e/ou consulta.	Obra	Um livro é escrito por um autor e publicado por uma editora. Um livro tem em geral vários exemplares, que por sua vez podem ser requisitados/reservados por utilizadores. A CDU indica a que área do conhecimento em que o livro está categorizado e o ISBN e ISSN são identificadores usados no mundo bibliográfico par identificar livros.
<b>Exemplar</b>	Cópia de um livro.	Cópia	Representa as cópias de um livro. O exemplar tem uma localização específica no espaço da biblioteca e poderá eventualmente ser requisitado/reservado por um utilizador ou apenas estar disponível para consulta. Os utilizadores podem ser penalizados por entregas de exemplares em mau estado de conservação e exemplares em mau estado de conservação deverão ser substituídos.
<b>Autor</b>	Aquele que escreve ou participa na redação de um livro.	Escritor	Um autor é referido na bibliografia pelos seus primeiros nomes e apelido, em que geralmente o apelido aparece sempre primeiro. Um autor pode ter vários livros na biblioteca.
<b>Editora</b>	Entidade responsável por gerir a publicação de um livro.	-	Uma editora publica livros e tem uma designação. Pode ser responsável pela publicação de vários livros na biblioteca.
<b>Requisição</b>	Ato de solicitar livros na biblioteca para uso pessoal por tempo limitado.	Pedido, Empréstimo, Requerimento	Requisição de um exemplar por um utilizador numa determinada data. O utilizador deve devolver o exemplar antes da data definida para o termo do empréstimo. Pode ainda renovar a requisição para estender o tempo de empréstimo, não podendo em todo o caso exceder o máximo de renovações permitidas.
<b>Localização</b>	Determina o local onde um livro pode ser encontrado dentro da BGUM.	Local, Posição	Piso e prateleira de uma estante onde o exemplar de um livro se pode encontrar.
<b>Utilizador</b>	Utente da biblioteca.	Utente, Cliente	Um utilizador pode efetuar reservas e requisições de exemplares livros. Pode ainda ser aluno da universidade ou não, tendo em todo o caso que estar registado no sistema, sendo-lhe atribuído um número mecanográfico. É contactado por correio eletrónico preferencialmente mas poderá alternativamente ser contactado via telefone.
<b>Colecção</b>	Conjunto de livros, organizados logicamente.	-	Possui uma designação, que corresponde ao nome da colecção. Agrega livros com aspetos comuns.



## II. Dicionário de Dados – Relacionamentos

Tabela 3 - Tabela de Relacionamentos

Entidade	Multiplicidade	Relacionamento	Multiplicidade	Entidade (Relacionada)
Livro	N (obrigatório)	Escrito; Redigido	N (obrigatório)	Autor
Livro	N (obrigatório)	Tem; Editado; Publicado	N (obrigatório)	Editora
Livro	1 (obrigatório)	Tem; Possui	N (obrigatório)	Exemplar
Exemplar	1(obrigatório)	Participa	N (opcional)	Requisição
Exemplar	N (opcional)	Reservado	N (opcional)	Utilizador
Exemplar	N (opcional)	Possui; Tem	1 (obrigatório)	Localização
Requisição	N (opcional)	Feita; Efetuada;	1 (obrigatório)	Utilizador
Colecção	1(opcional)	Tem: Possui	N(opcional)	Livro

### III. Dicionário de Dados – Atributos

Tabela 4 -- Tabela de atributos da tabela Livro

Nome da Entidade	Atributo	Descrição	Tipo de Dados & Comprimento/Domínio	Nulo	Tipo de Atributo	Valor por defeito
Livro	<u>idLivro</u>	Identificador único de livro	INT	Não	Simple	Auto incrementado
	Título	Título da obra	VARCHAR(250)	Não	Simple	-
	CodBarras	Código de barras do livro	VARCHAR(25)	Não	Simple	-
	ISBN	<i>International Standard Serial Number</i> - número internacional normalizado para publicações em série	VARCHAR (13)	Não	Simple	-
	ISSN	<i>International Standard Book Number</i> - número padrão internacional de identificação de um livro	VARCHAR(10)	Não	Simple	-
	CDU	Classificação Decimal Universal – identifica o assunto de um livro	VARCHAR (45)	Não	Multivalor	-
	Coleccao	Identificador da coleção a que um livro pertence	INT	Sim	Simple	-

Tabela 5 - Tabela de atributos da tabela Autor

Nome da Entidade	Atributo	Descrição	Tipo de Dados & Comprimento/Domínio	Nulo	Tipo de Atributo	Valor por defeito
Autor	<u>idAutor</u>	Identifica unicamente um autor	INT	Não	Simple	Auto incrementado
	Nome	Nome completo de um autor	-	Não	Composto (PrimeirosNomes, Apelido)	-
	PrimeirosNomes	Primeiros nomes de um autor	VARCHAR(75)	Não	Simple	-
	Apelido	Último apelido de um autor	VARCHAR (25)	Não	Simple	-

Tabela 6 -- Tabela de atributos do relacionamento Autor - Livro

Nome da Entidade	Atributo	Descrição	Tipo de Dados & Comprimento/Domínio	Nulo	Tipo de Atributo	Valor por defeito
Autor-Escreve-Livro	<u>idAutor</u>	Identificador do autor que escreveu o livro	INT	Não	Simples	-
	<u>IdLivro</u>	Identificador do livro que foi escrito	INT	Não	Simples	-

Tabela 7 - Tabela de atributos da tabela Editora

Nome da Entidade	Atributo	Descrição	Tipo de Dados & Comprimento/Domínio	Nulo	Tipo de Atributo	Valor por defeito
Editora	<u>idEditora</u>	Identifica unicamente uma editora	INT	Não	Simples	Auto incrementado
	Designacao	Nome da editora	VARCHAR(75)	Não	Simples	-

Tabela 8 -- Tabelas de atributos do relacionamento Livro - Editora

Nome da Entidade	Atributo	Descrição	Tipo de Dados & Comprimento/Domínio	Nulo	Tipo de Atributo	Valor por defeito
Livro-Publicado-Editora	<u>idLivro</u>	Identificador do livro publicado	INT	Não	Simples	-
	<u>IdEditora</u>	Identificador da editora que publicou o livro	INT	Não	Simples	-
	Edicao	Número de edição do livro	INT	Não	Simples	-
	Ano	Ano em que o livro foi publicado	YEAR	Não	Simples	-

Tabela 9 - Tabela de atributos da tabela Exemplar

Nome da Entidade	Atributo	Descrição	Tipo de Dados & Comprimento/Domínio	Nulo	Tipo de Atributo	Valor por defeito
Exemplar	<u>idExemplar</u>	Identificador único de exemplar	INT	Não	Simples	Auto incrementado
	Condicao	Estado em que se encontra o exemplar	VARCHAR(75)	Não	Simples	-
	Disponibilidade	Indica a disponibilidade de um exemplar – Não requisitável (0), Requisitado (1), Livre (2)	INT / 0,1,2	Não	Simples	-
	Localizacao	Identificador da localização de um exemplar	INT	Não	Simples	-
	Livro	Identificação do livro a que um exemplar corresponde	INT	Não	Simples	-

Tabela 10 - Tabela de atributos da tabela Requisição

Nome da Entidade	Atributo	Descrição	Tipo de Dados & Comprimento/ Domínio	Nulo	Tipo de Atributo	Valor por defeito
Requisição	<u>idRequisicao</u>	Identificador único de uma requisição	INT	Não	Simple	Auto incrementado
	DataRequisicao	Data em que a requisição ocorreu	DATE	Não	Simple	-
	DataEntrega	Data em que um livro deve ser entregue	DATE	Não	Simple	-
	Estado	Refere-se ao estado de um exemplar na requisição – 0 se estiver requisitado (requisição ativa), 1 se já foi entregue	INT / 0,1	Não	Simple	-
	NroMaxRenovacoes	Número máximo de vezes que uma requisição pode ser renovada	INT	Não	Simple	6
	NrRenovacoes	Número atual de renovações já efetuadas na requisição – não pode exceder NroMaxRenovacoes	INT	Não	Simple	-
	Exemplar	Identificador do exemplar que participa na requisição	INT	Não	Simple	-
	Utilizador	Identificado do utilizador que participa na requisição	INT	Não	Simple	-

Tabela 11 - Tabela de atributos da tabela Utilizador

Nome da Entidade	Atributo	Descrição	Tipo de Dados & Comprimento/D omínio	Nulo	Tipo de Atributo	Valor por defeito
Utilizador	<u>idUser</u>	Identifica unicamente um utilizador	INT	Não	Simple	Auto incrementado
	Tipo	Distingue os utilizadores – Aluno de licenciatura (A), Alunos de pós-graduação (PG / ID), Docente (D), Funcionário (F), Investigador (IE), Leitor externo (LE) e aluno de Erasmus (X).	VARCHAR(2) / A, PG, ID, D, F, LE, X	Não	Simple	-
	Nome	Nome completo do utilizador	VARCHAR(75)	Não	Simple	-
	Email	Endereço de Email do utilizador	VARCHAR(75)	Não	Simple	-
	CC	Número de identificação civil do utilizador	VARCHAR(10)	Não	Simple	-
	NroMecanografico	Parte numérica do número mecanográfico do utilizador	VARCHAR(10)	Não	Simple	-
	Telefone	Contacto telefónico do utilizador	VARCHAR(10)	Não	Simple	-



Tabela 12 - Tabela de atributos do relacionamento Exemplar - Utilizador

Nome da Entidade	Atributo	Descrição	Tipo de Dados & Comprimento/Domínio	Nulo	Tipo de Atributo	Valor por defeito
Reserva (Exemplar-reservado-Utilizador)	<u>idExemplar</u>	Identificador do exemplar reservado	INT	Não	Simples	-
	<u>idUser</u>	Identificador do utilizador que reservou o livro	INT	Não	Simples	-
	<u>DataReserva</u>	Data em que um livro foi reservado. Serve para responder a reservas por ordem cronológica.	DATE	Não	Simples	-
	Estado	Identifica o estado de uma reserva – Reserva pedida (0), Livro pronto a levantar (1), Livro levantado (2), Reserva cancelada (3)	INT / 0, 1, 2, 3	Não	Simples	-

Tabela 13 - Tabela de Atributos da tabela Localização

Nome da Entidade	Atributo	Descrição	Tipo de Dados & Comprimento/Domínio	Nulo	Tipo de Atributo	Valor por defeito
Localização	<u>idLocal</u>	Identificador único de localização	INT	Não	Simples	Auto incrementado
	Piso	Identifica o piso, dentro da biblioteca, onde o exemplar se encontra	INT	Não	Simples	-
	Estante	Identifica a estante em que o exemplar se encontra	INT	Não	Simples	-
	Prateleira	Identifica a prateleira em que o exemplar se encontra	INT	Não	Simples	-

Tabela 14 - Tabela dos atributos da tabela Colecção

Nome da Entidade	Atributo	Descrição	Tipo de Dados & Comprimento/Domínio	Nulo	Tipo de Atributo	Valor por defeito
Colecção	<u>idColecção</u>	Identificador único de Colecção	INT	Não	Simples	Auto incrementado
	Designacao	Nome pelo qual a colecção é conhecida	VARCHAR(75)	Não	Simples	-

Tabela 15 - Tabela dos atributos da tabela CDU

Nome da Entidade	Atributo	Descrição	Tipo de Dados & Comprimento/Domínio	Nulo	Tipo de Atributo	Valor por defeito
CDU	<u>CDU</u>	Classificação Decimal Universal – identifica o assunto de um livro	VARCHAR(45)	Não	Simples	-
	<u>Livro</u>	Identificador do livro	INT	Não	Simples	-