**Book Sharing**

Relatório de Projeto no âmbito da Unidade Curricular de **Bases de Dados**

******

Mestrado Integrado em Engenharia Informática e Computação

Ano letivo 2017/2018

**| Grupo 4 da Turma 3 |**

César Medeiros **|** 201605344

Margarida Silva **|** 201606214

Pedro Costa **|** 201605125

**Índice**

[**Descrição do contexto** 2](#_Toc511485634)

[**Diagrama de Classes UML** 4](#_Toc511485635)

[**Modelo relacional e Dependências Funcionais** 5](#_Toc511485636)

[**Análise de dependências funcionais: Formas Normais** 7](#_Toc511485637)

[**Restrições** 8](#_Toc511485638)

[Lista 8](#_Toc511485639)

[Implementação 9](#_Toc511485640)

# **Descrição do contexto**

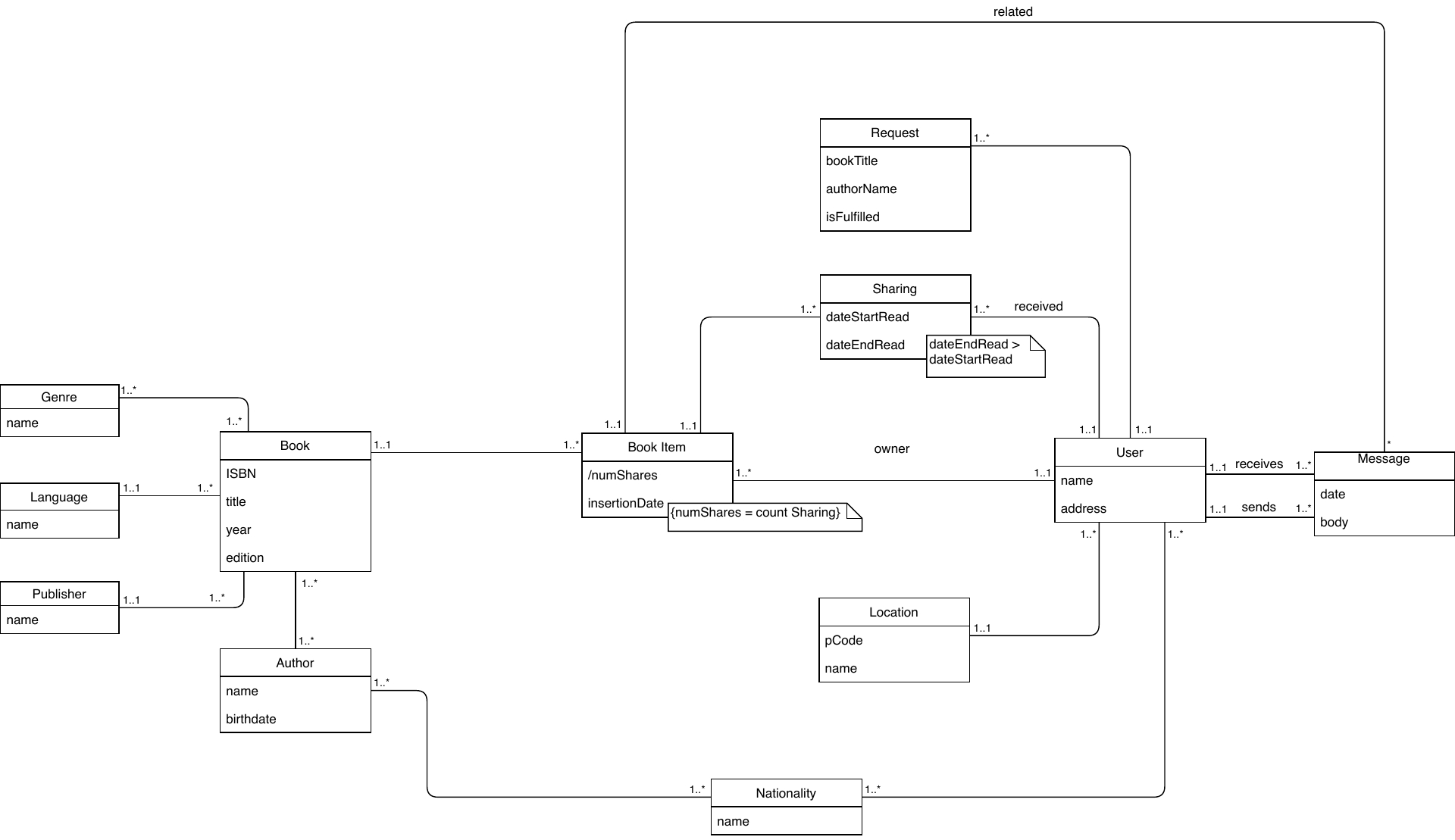
A nossa base de dados tem como fim o suporte de uma aplicação de **Book Sharing.**

Na aplicação os utilizadores podem partilhar e requisitar livros, trocando mensagens entre si para facilitar o processo de troca.

Um utilizador acede à aplicação e acede a um conjunto de Book Items que estão disponíveis para lhe serem emprestados. Cada Book Item está associado a um Book, que por sua vez está associado ao autor, linguagem, género e editora.

Por outro lado, é oferecida a possibilidade do utilizador inserir Requests de livros. Estes servem apenas para dar a indicação de que um determinado utilizador anda à procura de um determinado livro e ainda não conseguiu agendar uma troca (seja por falta de stock na base de dados (ninguém tem o livro), incompatibilidades de tempo/localização com os users, etc). Quando vê a sua necessidade do livro chegar ao fim, preenche o campo isFulfilled – seja porque já trocou o livro, ou simplesmente já não tem mais a necessidade de obter aquele livro em específico.

No processo de Sharing, é associado um Book Item e um User a quem vai ser emprestado o livro, sendo guardada a data de troca, e a data do fim da leitura. Este último atributo está nulo enquanto o empréstimo decorre, sendo apenas preenchido quando o user dá a leitura por terminada. Nessa altura, o Book Item volta a estar disponível para que outros utilizadores o possam pedir emprestado também. O processo de partilha é acordado via mensagens entre utilizadores. Como cada user tem uma localização associada, é possível que os utilizadores vão procurando uns aos outros em localizações próximas, estabelecendo contactos.

**Diagrama de Classes UML**

# **Modelo relacional e Dependências Funcionais**

* ***Genre*** (id, name);
  1. {id}→{name}
* ***Language*** (id, name);
  1. {id}→{name}
* ***Publisher*** (id, name);
  1. {id}→{name};
* ***Nationality*** (id, name);
  1. {id}→name;
* ***Book*** (id, ISBN, title, year, edition, language → Language, publisher → Publisher);
  1. {id}→{ISBN}
  2. {ISBN}→{title, year, edition, language, publisher};
* ***BookGenre*** (idB → Book, idG → Genre);
* ***BookAuthor*** (idA → Author, idB → Book);
* ***Author*** (id, name, birthdate);
  1. {id} → {name, birthdate};

* ***AuthorNationality*** (author→ Author, nationality→ Nationality);
* ***Location*** (id, pCode, name);
  1. {id} → {pCode};
  2. {pCode} → {name};
* ***User*** (id, name, address, location→ Location);
  1. {id} → {name, address, location};
* ***UserNationality*** (user→ Nationality, nationality→ Nationality);
* ***BookItem*** (id, insertionDate, book→Book, owner→User);
  1. {id} → {insertionDate, book, owner};
* ***Sharing*** (id, startDate, endDate, book→ BookItem, receives → User);

1. {id} → {startDate, endDate, book, receives};

* ***Request*** (id, bookTitle, authorName, isFulfilled, request→ User);
  1. {id} → {bookTitle, authorName, isFulfilled, request};
* ***Message*** (id, date, body, receives → User, sender→ User,context→BookItem);
  1. {id} → {date, body, receives, sender, context};

# **Análise de dependências funcionais: Formas Normais**

As violações à terceira forma normal e à forma normal de Boyce-Codd estão presentes nas relações:

* ***Book***

Na dependência ii), o lado esquero, {ISBN}, não é nem uma superkey, nem é um atributo primo (a chave candidata é apenas {id}).

* ***Location***

Motivo análogo ao enunciado em Book.

Todas as outras relações cumprem todas as condições das formas normais. Como só possuem uma dependência funcional, o lado esquerdo será obrigatoriamente uma chave, neste caso, chave candidata.

# **Restrições**

## Lista

## Implementação