



Progetto Object Orientation e Basi di Dati

Lorenzo Tecchia N86004446

Mirko Marciano N86004019

10/12/2022

**Progettazione e sviluppo di un applicativo per la gestione di una
Biblioteca Digitale**

Indice

1	Descrizione e Analisi del Progetto	5
1.1	Descrizione e risoluzione sintetica	5
2	Progettazione concettuale	6
2.1	Class Diagram	6
2.2	Analisi della ristrutturazione del Class Diagram	7
2.2.1	Analisi delle ridondanze	7
2.2.2	Analisi degli identificativi	7
2.2.3	Rimozione degli attributi multipli	7
2.2.4	Rimozione degli attributi composti	7
2.2.5	Partizione/Accorpamento delle associazioni	7
2.2.6	Rimozione delle gerarchie, delle composizioni	7
2.3	Class Diagram ristrutturato	8
2.4	Dizionario delle Classi	10
2.5	Dizionario delle associazioni	10
3	Schema logico	11
3.1	Spiegazione Schema Logico	11
3.1.1	Schema	11
4	Progettazione Fisica	13
4.1	Creazione delle Tabelle	13
5	Caso d'uso e manuale	14

Elenco delle figure

2.1	Class Diagram	6
2.2	Class Diagram Ristrutturato	8
2.3	Diagramma Entity Relation	9

Elenco delle tabelle

2.1	Dizionario delle Classi	10
2.2	Tabella delle Associazioni	10

Capitolo 1

Descrizione e Analisi del Progetto

1.1 Descrizione e risoluzione sintetica

Il progetto consiste in una base di dati relazionale unito ad un'interfaccia grafica costruita in Java per la gestione di una libreria digitale. Si è pensato di implementare una base di dati relazionali con server in locale tramite il dialetto PostgreSQL.

Si è pensato di articolare la base dati in cinque classi di oggetti per tenere traccia di articoli, libri, autori, collane e riviste.

E' stato messo in risalto il tracciamento per autori e argomenti tramite l'utilizzo di viste.

Si è pensato di controllare gli inserimento delle tuple nel database tramite implementazioni di trigger e domini che controllano ad ogni inserimento la validità degli elementi caratteristici degli oggetti consumabili all'interno della biblioteca, quali ISBN e ISSN per libri e collane rispettivamente. Per articoli e riviste vengono controllati gli inserimenti corretti di DOI e ISSN rispettivamente.

Tutti i tipi di codici menzionati sono stati scelti come vincoli di chiavi primarie per i rispettivi oggetti.

Capitolo 2

Progettazione concettuale

2.1 Class Diagram

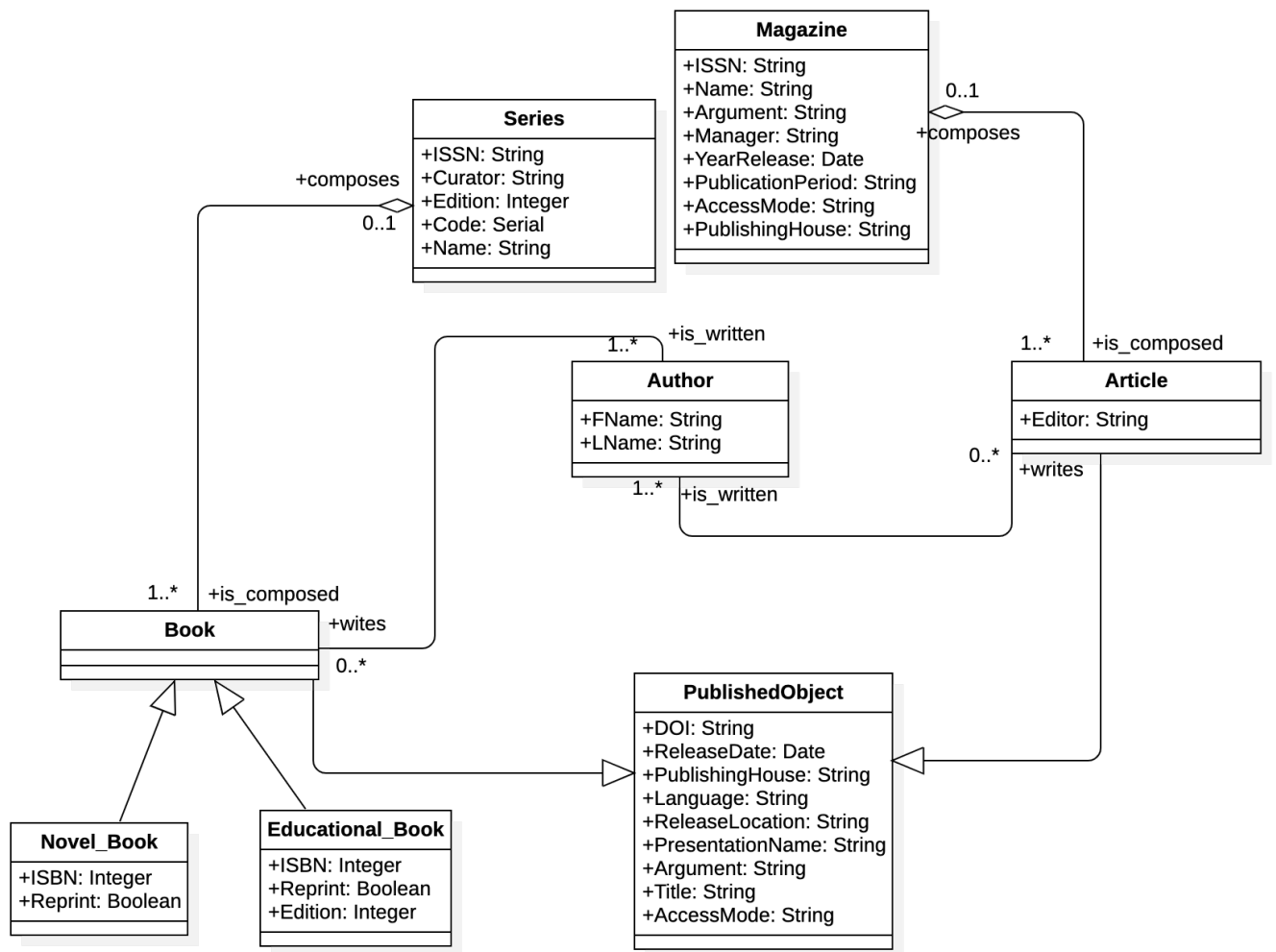


Figura 2.1: Class Diagram

2.2 Analisi della ristrutturazione del Class Diagram

2.2.1 Analisi delle ridondanze

Le ridondanze individuate nel class diagram sono presenti nella specializzazione della classe **Book** in: **Novel_Book** e **Educational_Book**.

Gli attributi: **ISBN** e **Reprint** di entrambe le specializzazioni vengono inserite all'interno della classe **Book**.

2.2.2 Analisi degli identificativi

Identificativo primario per la classe **Book** è l'attributo **DOI**, viene attribuito all'attributo **ISBN** il vincolo di chiave candidata. Per la classe **Author** viene implementato un attributo aggiuntivo: **ID_Author** che sarà chiave primaria. Per la classe **Article** viene usato l'attributo **DOI** come chiave primaria. Per le classi **Series** e **Magazine** viene usato come chiave primaria l'attributo **ISSN**.

2.2.3 Rimozione degli attributi multipli

Non sono presenti attributi multipli all'interno del Class Diagram.

2.2.4 Rimozione degli attributi composti

Non sono presenti attributi composti all'interno del Class Diagram.

2.2.5 Partizione/Accorpamento delle associazioni

In questo Class Diagram non sono presenti associazioni 1..1 da eliminare.

2.2.6 Rimozione delle gerarchie, delle composizioni

Nel Class Diagram vengono incorporati nella classe **Book** le specializzazioni **Novel_Book** e **Educational_Book** assieme ai rispettivi attributi. All'attributo **Argument** nel Class Diagram ristrutturato, viene specificato l'argomento educativo (Filosofia, Economia ecc.) oppure all'attributo viene assegnato il valore: **Romanzo**. Dato l'interesse di tracciamento delle classi **Book** e **Article** in modo separato, viene eliminata la classe **PublishedObject** e i suoi attributi vengono inseriti all'interno delle classi menzionate precedentemente. Le composizioni che riguardano le classi **Series** e **Magazine**, vengono eliminate e sostituite da una semplice associazione.

2.3 Class Diagram ristrutturato

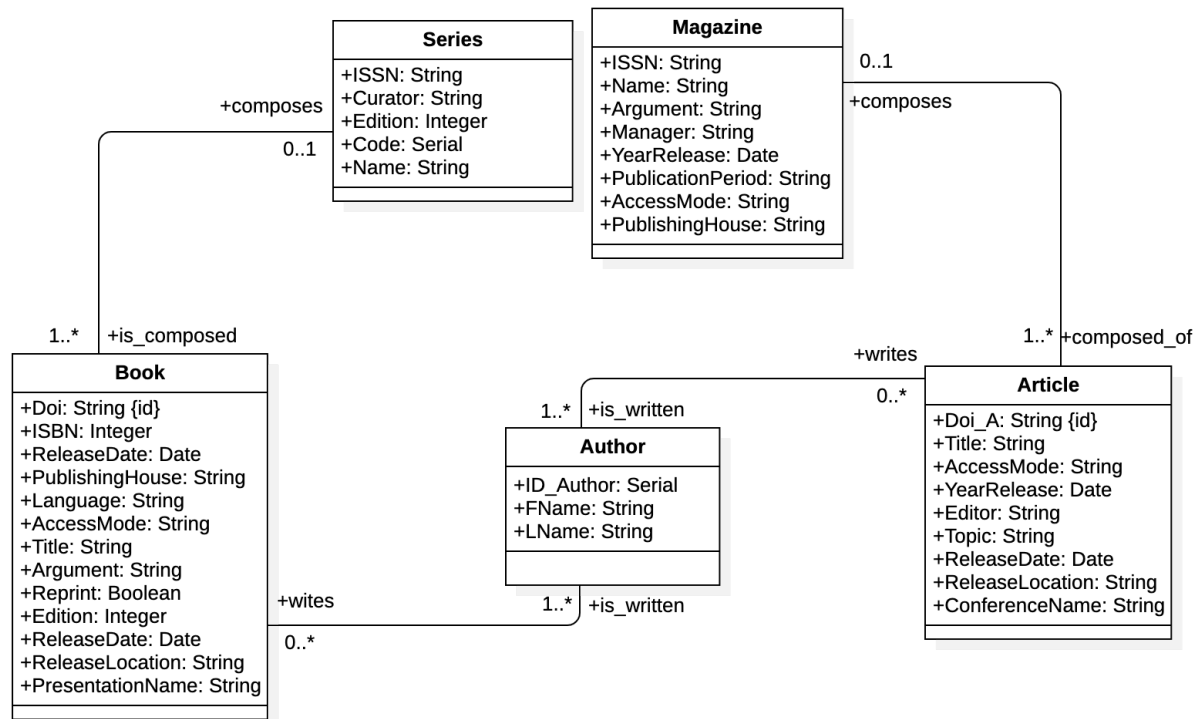


Figura 2.2: Class Diagram Ristrutturato

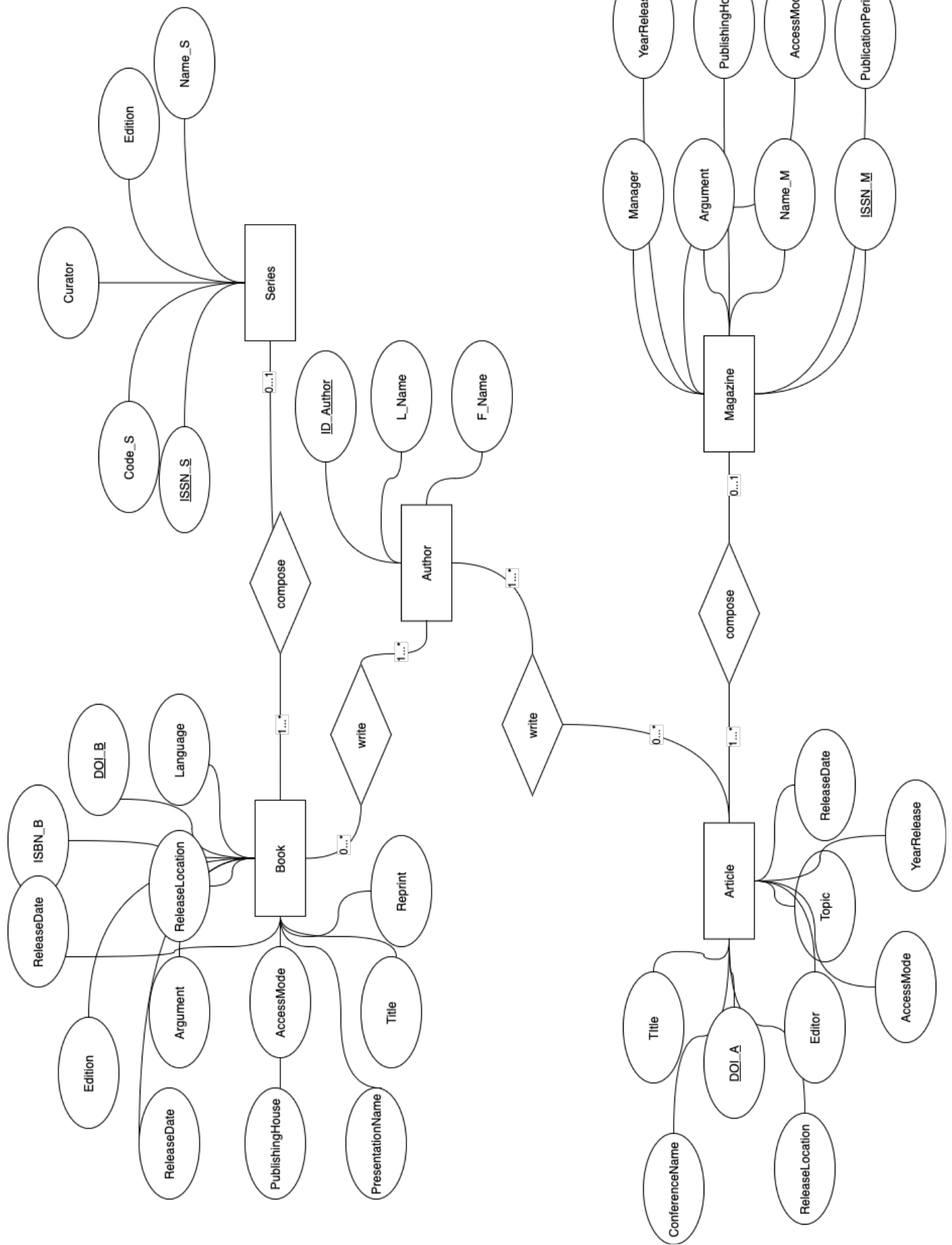


Figure 2.3: Diagramma Entity Relation

2.4 Dizionario delle Classi

Tabella 2.1: Dizionario delle Classi

Classe	Spiegazione	Attributi
Author	Autore di libri o articoli	ID_Author (Serial): Identificazione dell'autore. FirstName (String): Nome dell'autore. LastName (String): Cognome dell'autore.
Books	Oggetti leggibili, romanzi o d'educazione	ISBN (Integer): Classificazione numerica di un libro. Edition (Integer): Numero d'edizione. AccessMode (AccessMode): Modo di fruizione. ReleaseDate (Date): Data di pubblicazione. PublishingHouse (String): Casa editrice che ha stampato il libro. ReleaseLocation (String): Luogo di pubblicazione. Language (String): Lingua in cui è scritto il libro. Title (String): Titolo del libro. Argument (String): Argomento del libro. Reprint (Boolean): Parametro che identifica se il libro è una ristampa. PresentationName (String): Nome della presentazione alla quale il libro è presentato.
Series	Insieme di libri	ISSN (String): Numero internazionale che identifica le collane. Edition (Integer): Numero dell'edizione. Curator (String): Curatore della collana. Code (Serial): Codice affidato alla collana. Name (String): Nome della collana.
Magazine	Insieme di Articoli	ISSN(Integer): numero internazionale che identifica le riviste. Name (String): Nome della rivista. Argument (String): Argomento della rivista. Manager (String): Manager della rivista. YearRelease (Timestamp): Anno di pubblicazione. PublicationPeriod (String): Periodicità della rivista. AccessMode (AccessMode): Modo di fruizione.
Article	Articoli di ricerca Scientifica	DOI (String): Digital object Identifier dell'articolo. Title (String): Titolo dell'articolo. AccessMode (AccessMode): Metodo di fruizione. Editor (String): Editore dell'articolo. ReleaseDate (Timestamp): Data di pubblicazione. ReleaseLocation (String): Luogo di pubblicazione. ConferenceName (String): Nome di conferenza in cui è presentato/discusso l'articolo.

2.5 Dizionario delle associazioni

Tabella 2.2: Tabella delle Associazioni

Nome	Descrizione	Classi Coinvolte
composes/is_composed	Una collana è composta da uno o più libri/ Un libro può comporre oppure no una collana	Series/Book
writes/is_written	Un libro è scritto da uno o più autori/ Un autore scrive molti oppure nessun libro	Book/Author
is_written/writes	Un autore scrive molti oppure nessun articolo/ Un articolo è scritto da uno o più autori	Author/Article
composes/is_composed	Un articolo può comporre oppure no una rivista/ Una rivista è composta da uno o più articoli	Article/Magazine

Capitolo 3

Schema logico

3.1 Spiegazione Schema Logico

Il seguente schema logico riassume la composizione del database, evidenziando le chiavi primarie (in grassetto) e le chiavi esterne per il collegamento di una tabella con l'altra (sottolineate).

3.1.1 Schema

- Author

(**IDAuthor**, FName, LName)

- Book

(**DOI_B**, ISBN_B, PublishingHouse, Language, AccessMode, Title, Argument, Reprint, Edition, ReleaseDate, ReleaseLocation, PresentationName, FK_Author, FK_Series)

FK_Author → *Author*(*ID_Author*)

FK_Series → *Series*(*ISSN_S*)

- Article

(**DOI_A**, Title, AccessMode, YearRelease, Editor, Topic, ReleaseDate, ReleaseLocation, ConferenceName, FK_Author, FK_Magazine)

FK_Author → *Author*(*ID_Author*)

FK_Magazine → *Series*(*ISSN_M*)

- Series

(**ISSN_S**, Curator, Edition, Code, Name)

- Magazine

(**ISSN_M**, Name, Argument, Manager, YearRelease, PublicationPeriod, AccessMode, PublishingHouse)

- Author_Books

(AuthorsFK, BooksFK)

AuthorsFK → *Author*(*CodAuthor*)

BooksFK → *Book*(*ISBN_B*)

- Author_Article
(AuthorsFK, ArticlesFK)

AuthorsFK \rightarrow *Author*(*CodAuthor*)

ArticlesFK \rightarrow *Article*(*DOI_A*)

Capitolo 4

Progettazione Fisica

4.1 Creazione delle Tabelle

Capitolo 5

Caso d'uso e manuale