Assignment 8

create a simple application to manage a collection of books and authors. The application will allow you to:

1. Define the Models

1. Define a book model:
    - `title` (String, required)
    - content (String, required)
    - `author` (String, required)
    - `publishedDate` (Date, default to the current date)
2. Define an author model:
    - `name` (String, required)
    - `bio` (String)
    - `birthDate` (Date)
    - `books` (Array of ObjectIds referencing Book model)

2. Test the Application

1. Use a Postman to test the API endpoints:
    - `POST` request to create a new book.
    - `GET` request to retrieve all books.
    - `GET` request to retrieve a single book by its ID.
    - `PATCH` request to update a book by its ID.
    - `DELETE` request to delete a book by its ID.
    - `POST` request to create a new author.
    - `GET` request to retrieve all authors.
    - `GET` request to retrieve a single author by its ID.
    - `PATCH` request to update an author by its ID.
    - `DELETE` request to delete an author by its ID.

Bonus Task

- Add pagination to the GET endpoints for retrieving all books and authors.
- Implement search functionality to filter books by title or author, and authors by name or bio.
- Add a relationship so that when retrieving an author, the response includes a list of books written by them.

Post on LinkedIn:

- A GitHub repository containing the complete code for the CRUD application.
- Screenshots or a screen recording of the API endpoints being tested using Postman.

Submission

Submit the GitHub repository link along with a postman documentation.