

# **Course Project: Database Design and Implementation**

Final Project Report (Week 5)

## **Library Borrowing Tracker System**

**Group 12**

**Mahardika Ramadhana** (24/538247/PA/22831)

**Hammam Muhammad Yazid** (24/534894/PA/22687)

**Daffa M. Siddiq** (24/533358/PA/22569)

Department of Computer Science, Universitas Gadjah Mada

December 28, 2025

# Contents

<b>1</b>	<b>Introduction</b>	<b>2</b>
1.1	Comprehensive Problem Background . . . . .	2
1.2	Project Objectives and Rationalization . . . . .	2
1.3	System Scope and Business Logic . . . . .	2
<b>2</b>	<b>Database Design and Architecture</b>	<b>2</b>
2.1	Analytical Entity Relationship Diagram (ERD) . . . . .	2
<b>3</b>	<b>Application Implementation and UI Analysis</b>	<b>3</b>
3.1	User Interface Visualization and Functionality . . . . .	3
<b>4</b>	<b>Testing, Results, and Quality Assurance</b>	<b>6</b>
4.1	API Documentation (Swagger Integration) . . . . .	6
4.2	Functional Testing via Command Line (Curl) . . . . .	7
<b>5</b>	<b>Conclusion and Personal Reflection</b>	<b>8</b>
5.1	Final Project Conclusion . . . . .	8
5.2	Reflective Experience and Lessons Learned . . . . .	8

# 1 Introduction

## 1.1 Comprehensive Problem Background

Dalam ekosistem perpustakaan modern, pengelolaan data sirkulasi buku yang presisi merupakan fondasi utama operasional. Banyak institusi masih terjebak dalam metode manual yang menyebabkan tingginya risiko ketidakkonsistenan data dan kesulitan dalam melacak posisi fisik buku secara real-time. Masalah utama muncul saat penghitungan denda dilakukan secara manual, yang sering memicu kesalahan manusia. Sistem ini hadir untuk mengotomatisasi seluruh alur tersebut dengan infrastruktur digital yang kokoh dan terintegrasi secara relasional.

## 1.2 Project Objectives and Rationalization

Tujuan utama dari proyek ini adalah membangun arsitektur basis data relasional yang ternormalisasi (3NF) untuk menjamin integritas data jangka panjang. Kami menargetkan implementasi sistem yang mampu mengelola siklus hidup peminjaman mulai dari registrasi anggota hingga otomasi perhitungan penalti keuangan. Dengan sistem ini, transparansi data antara pihak pengelola perpustakaan dan anggota dapat ditingkatkan secara signifikan melalui dashboard yang informatif dan akurat.

## 1.3 System Scope and Business Logic

Sistem ini menerapkan aturan bisnis yang ketat: batas waktu standar peminjaman selama 14 hari kalender dan denda otomatis sebesar Rp1.000 untuk setiap hari keterlambatan per buku. Meskipun admin dapat mencari buku berdasarkan nama penulis atau judul, sistem memaksakan penggunaan ID unik (ID Anggota dan ID Buku) dalam setiap transaksi peminjaman untuk menghindari ambiguitas data yang sering terjadi pada pencatatan manual.

# 2 Database Design and Architecture

## 2.1 Analytical Entity Relationship Diagram (ERD)

Rancangan ERD merupakan representasi logis dari kebutuhan penyimpanan data. Kami memastikan referential integrity terjaga melalui pemanfaatan Primary Key dan Foreign Key yang memungkinkan pelacakan data lintas tabel tanpa kehilangan konteks relasionalnya.

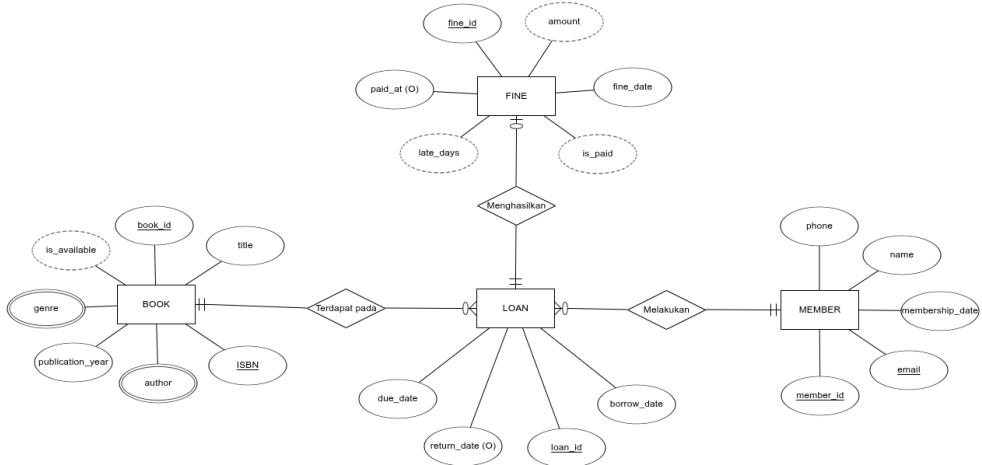


Figure 1: Rancangan ERD Database Perpustakaan (3NF)

## 3 Application Implementation and UI Analysis

### 3.1 User Interface Visualization and Functionality

Antarmuka dibangun menggunakan React dan Tailwind CSS untuk memberikan pengalaman pengguna yang intuitif bagi admin dalam mengelola inventaris secara efisien.

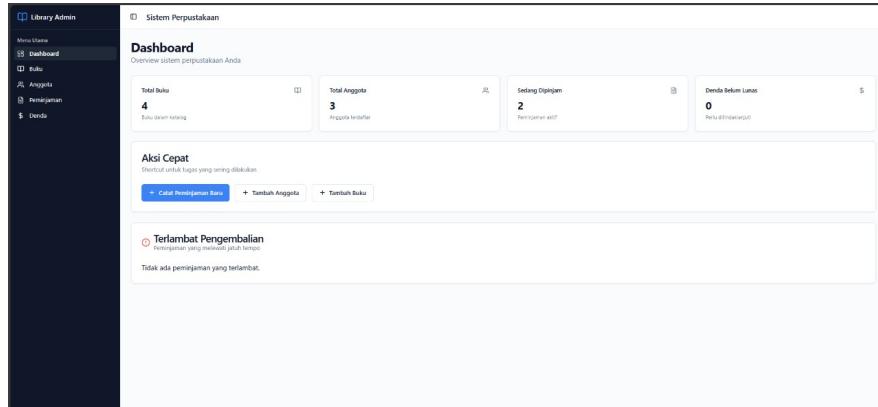


Figure 2: Dashboard Admin: Ringkasan Statistik Real-time

Sistem Perpustakaan							
Manajemen Buku							
Ketela katalog buku perpustakaan							
<a href="#">+ Tambah Buku</a>							
<b>Daftar Buku</b>		Can dan filer buku berdasarkan kategori					
<input type="text"/> Can berdasarkan judul atau penulis...		<input type="button"/> Simpan		<input type="button"/> Semua Genre		<input type="button"/> Semua Status	
Judul Buku	Penulis	Tahun Terbit	Genre	Status	Aksi		
Database	Daniel	2022	Computer Science	<input checked="" type="radio"/> Teredia	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Laut Bercerita	Lelia S. Chudori	2020	History	<input checked="" type="radio"/> Dijemput	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
OS	Matematika	2009	Horror	<input checked="" type="radio"/> Dijemput	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
TPSC	Tufel Barik Bahri	2024	Fiction	<input checked="" type="radio"/> Teredia	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Figure 3: Manajemen Katalog Buku: Menampilkan Seluruh Koleksi

Sistem Perpustakaan							
Manajemen Buku							
Ketela katalog buku perpustakaan							
<a href="#">+ Tambah Buku</a>							
<b>Daftar Buku</b>		Can dan filer buku berdasarkan kategori					
<input type="text"/> Can berdasarkan judul atau penulis...		<input type="button"/> Simpan		<input type="button"/> Semua Genre		<input type="button"/> Semua Status	
Judul Buku	Penulis		Status	Aksi			
Database	Daniel		<input checked="" type="radio"/> Teredia	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
Laut Bercerita	Lelia S. Chudori		<input checked="" type="radio"/> Dijemput	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
OS	Matematika	2009	<input checked="" type="radio"/> Dijemput	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
TPSC	Tufel Barik Bahri	2024	<input checked="" type="radio"/> Teredia	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	

Figure 4: Modul Penambahan Inventaris Buku Baru

Sistem Perpustakaan							
Manajemen Anggota							
Ketela data anggota perpustakaan							
<a href="#">+ Tambah Anggota</a>							
<b>Daftar Anggota</b>		Can anggota berdasarkan nama atau email					
<input type="text"/> can berdasarkan nama atau email...		<input type="button"/> Simpan		<input type="button"/> Semua		<input type="button"/> Aktif	
Name	Email	No. Telepon	Tanggal Bergabung	Aksi			
dafa mud	defu@gmail.com	085390087545	15/11/2025	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
dika ramadhana	dika@gmail.com	085312458745	24/11/2025	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
marco ramadhan	marco@gmail.com	085478948543	25/11/2025	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	

Figure 5: Basis Data Anggota Perpustakaan yang Aktif

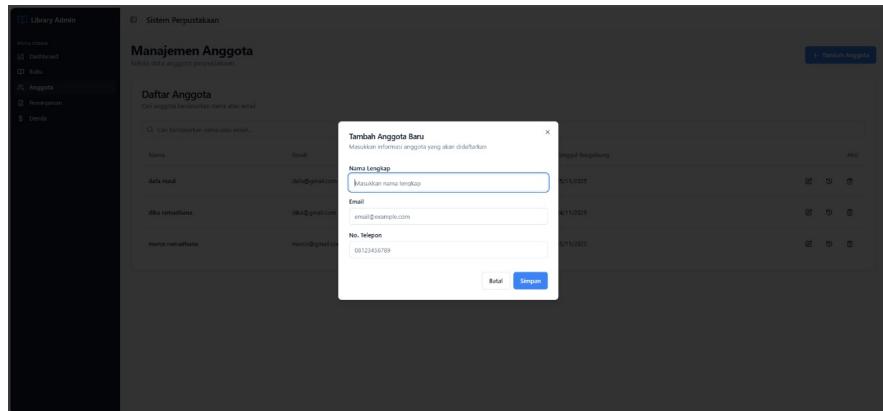


Figure 6: Modul Registrasi dan Penambahan Anggota Baru

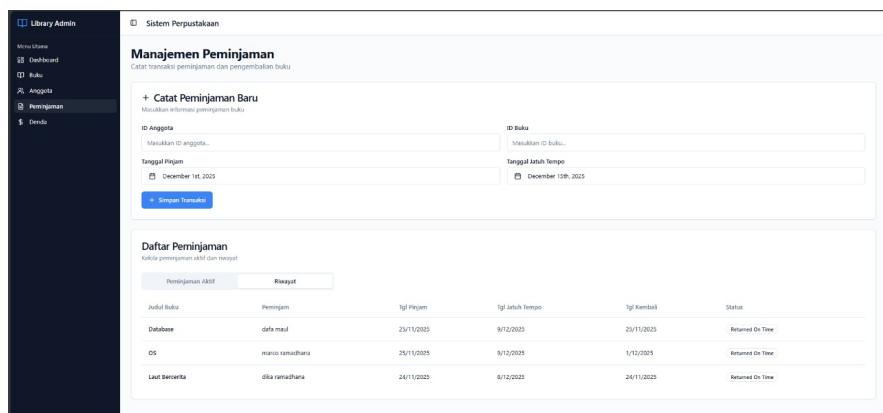


Figure 7: Sistem Manajemen Transaksi Peminjaman Buku

## 4 Testing, Results, and Quality Assurance

### 4.1 API Documentation (Swagger Integration)

Dokumentasi API memastikan komunikasi lancar antara frontend dan backend. Setiap endpoint didefinisikan secara transparan, mencakup parameter masukan dan format respon yang diharapkan.

The screenshot shows the Swagger UI for the Library Borrowing API. The title is "Library Borrowing API". Below it, a subtitle says "API untuk sistem perpustakaan (books, members, loans, fines)". The main content is organized into sections: "books" and "members".

**books**

- POST /api/books
- GET /api/books
- GET /api/books/{id}
- PATCH /api/books/{id}
- DELETE /api/books/{id}

**members**

- POST /api/members
- GET /api/members
- GET /api/members/{id}
- PATCH /api/members/{id}
- DELETE /api/members/{id}

Figure 8: Endpoint API Bagian 1: Manajemen Books dan Members

The screenshot shows the Swagger UI for the Library Borrowing API. The main content is organized into sections: "loans" and "fines".

**loans**

- POST /api/loans
- GET /api/loans
- GET /api/loans/{id}
- DELETE /api/loans/{id}
- PATCH /api/loans/{id}/return

**fines**

- GET /api/fines
- GET /api/fines/{id}
- PATCH /api/fines/{id}/status

Figure 9: Endpoint API Bagian 2: Manajemen Loans dan Fines

```

Schemas

CreateBookDto ✕ {
    title*           > [...]
    author*          > [...]
    genre            > [...]
    publication_year > [...]
}

UpdateBookDto ✕ {
    title           > [...]
    author          > [...]
    genre           > [...]
    publication_year > [...]
}

CreateMemberDto ✕ {
    name*           > [...]
    email*          > [...]
    phone           > [...]
    membership_date > [...]
}

UpdateMemberDto ✕ {
    name           > [...]
    email          > [...]
    phone          > [...]
    membership_date > [...]
}

CreateLoanDto ✕ {
    member_id*      > [...]
    book_id*        > [...]
}

ReturnLoanDto ✕ {
    return_date      > [...]
}

UpdateFineStatusDto ✕ {
    is_paid          > [...]
}

```

Figure 10: Skema Struktur Data (Data Transfer Object)

## 4.2 Functional Testing via Command Line (Curl)

Untuk memverifikasi keandalan backend, kami melakukan pengujian fungsional menggunakan tool curl yang mencakup skenario pembuatan data (POST) dan penghapusan data (DELETE).

```

Curl
curl -X 'POST' \
  'http://localhost:3000/api/books' \
  -H 'Accept: */*' \
  -H 'Content-Type: application/json' \
  -d '{
    "title": "string",
    "author": "string",
    "genre": "string",
    "publication_year": 2009
}'

```

Request URL  
<http://localhost:3000/api/books>

Server response

Code	Details
201	Response body

```
{
  "book_id": "853dfa0f-243c-4cc3-b664-b28dd9917870",
  "title": "string",
  "author": "string",
  "genre": "string",
  "publication_year": 2009,
  "is_available": true,
  "created_at": "2025-12-01T09:05:37.071Z",
  "updated_at": "2025-12-01T09:05:37.071Z"
}
```

Download

Response headers

```

access-control-allow-origin: http://localhost:3000
connection: keep-alive
content-length: 202
content-type: application/json; charset=utf-8
date: Mon, 01 Dec 2025 16:05:34 GMT
etag: W/c2-jlhZl8B6LXy0eV/NKscrvYOGba"
keep-alive: timeout=5
vary: Origin
x-powered-by: Express

```

Responses

Code	Description	Links
201		No links

Figure 11: Verifikasi Backend: Uji Coba POST Success (Status 201)

```

Curl
curl -X 'DELETE' \
  'http://localhost:3000/api/books/853dfa0f-243c-4cc3-b664-b28dd9917870' \
  -H 'Accept: */*'

```

Request URL  
<http://localhost:3000/api/books/853dfa0f-243c-4cc3-b664-b28dd9917870>

Server response

Code	Details
200	Response headers

```

access-control-allow-origin: http://localhost:3000
connection: keep-alive
content-length: 0
date: Mon, 01 Dec 2025 16:10:14 GMT
keep-alive: timeout=5
vary: Origin
x-powered-by: Express

```

Responses

Code	Description	Links
200		No links

Figure 12: Verifikasi Backend: Uji Coba DELETE Success (Status 200)

## 5 Conclusion and Personal Reflection

### 5.1 Final Project Conclusion

Proyek ini berhasil mencapai seluruh objektif awal yang ditetapkan. Sistem yang dibangun mampu menangani kompleksitas manajemen perpustakaan mulai dari normalisasi data tingkat lanjut hingga otomasi denda harian sebesar Rp1.000 dengan performa yang stabil.

### 5.2 Reflective Experience and Lessons Learned

Tantangan teknis utama berupa sinkronisasi zona waktu pada objek tanggal berhasil diatasi dengan penerapan standarisasi format ISO-8601 di seluruh lapisan aplikasi. Hal

ini memastikan perhitungan jatuh tempo peminjaman 14 hari berjalan secara akurat. Pengalaman ini memberikan pelajaran berharga mengenai pentingnya desain database yang matang sebelum tahap implementasi kode dilakukan.