

**LAPORAN PROYEK WEBSITE COMPANY TUGAS AKHIR MATA
KULIAH PEMROGRAMAN DASAR
SISTEM INFORMASI PERPUSTAKAAN BERBASIS FLAT FILE**



Oleh Kelompok 5:

PRASETYO HERWIBOWO	E41250545
HADI AKRAM RAMADHAN	E41260480
HARIS RAHMAN KURNIAWAN	E41250699

**PRODI TEKNIK INFORMATIKA
JURUSAN TEKNOLOGI INFORMASI
POLITEKNIK NEGERI JEMBER 2025/2026**

1. Pendahuluan

1.1 Latar Belakang

Pengelolaan data perpustakaan secara manual seringkali menimbulkan masalah ketidakakuratan data dan inefisiensi waktu. Dalam proyek ini, kami mengembangkan **Sistem Informasi Perpustakaan Sederhana**.

Keunikan dari sistem ini adalah **tidak menggunakan database RDBMS (seperti MySQL)**, melainkan menggunakan teknik **Flat-File Database** (penyimpanan berbasis file teks). Pendekatan ini dipilih untuk mendemonstrasikan kemampuan logika dasar pemrograman PHP dalam memanipulasi file (file handling), array, dan struktur data tanpa bergantung pada *engine* database eksternal.

1.2 Tujuan Projek

1. Membangun sistem CRUD (*Create, Read, Update, Delete*) untuk data Buku dan Anggota.
2. Mengimplementasikan logika bisnis peminjaman (validasi stok dan kuota peminjaman).
3. Mengimplementasikan sistem denda otomatis pada proses pengembalian.
4. Menerapkan desain antarmuka responsif dengan tema *Dark Mode*.

2. Arsitektur Sistem

Sistem ini dibangun menggunakan arsitektur prosedural sederhana.

- **Bahasa Pemrograman:** PHP (versi 7.4 ke atas direkomendasikan).
- **Penyimpanan Data:** Text Files (.txt) dengan format *Pipe-Separated Values* (|).
- **Styling:** CSS3 (Custom Dark Theme).

2.1 Struktur Folder

```
fin/
├── config.php           # Konfigurasi & constants
├── functions.php       # Helper functions
├── style.css           # Modern CSS design
├── index.php           # Dashboard
├──
├── books.php           # Manajemen Buku
├── book_add.php
├── book_edit.php
├── book_delete.php
├──
├── members.php         # Manajemen Anggota
├── member_add.php
├── member_edit.php
├── member_history.php
├──
├── borrow.php          # Sistem Peminjaman
├── borrow_add.php
├──
├── return.php          # Sistem Pengembalian
├──
├── data/               # Penyimpanan data (.txt)
│   ├── books.txt
│   ├── members.txt
│   ├── borrowings.txt
│   └── returns.txt
├──
├── screenshots/        # Screenshot aplikasi
└── LAPORAN_PROYEK.md  # Dokumentasi lengkap
```

3. Perancangan Database (Flat-File)

Karena tidak menggunakan SQL, struktur data didefinisikan berdasarkan urutan indeks array yang dipisahkan oleh karakter pipe (|).

3.1 Tabel Buku (books.txt)

Menyimpan inventaris buku perpustakaan.

Indeks	Atribut	Tipe Data	Contoh Data
:---	:---	:---	:---
0	ID Buku	String (Auto)	B-1
1	Judul	String	Algoritma Dasar
2	Penulis	String	Budi Raharjo
3	Kategori	String	Teknologi
4	Tahun	Integer	2024
5	Status	Enum	tersedia / dipinjam
6	Kondisi	String (CSV)	baru,bekas

3.2 Tabel Anggota (members.txt)

Menyimpan data anggota yang terdaftar.

Indeks	Atribut	Tipe Data	Contoh Data
:---	:---	:---	:---
0	ID Anggota	String (Auto)	M-1
1	Nama	String	Ahmad Santoso
2	Email	String	ahmad@mail.com
3	Telepon	String	0812345678
4	Status	Enum	aktif / nonaktif

3.3 Tabel Pinjaman (borrowings.txt)

Mencatat transaksi sedang berlangsung atau selesai.

Indeks	Atribut	Deskripsi
:---	:---	:---
0	ID Loan	Primary Key Transaksi (L-1)
1	Member ID	Foreign Key ke members.txt
2	Book ID	Foreign Key ke books.txt
3	Tgl Pinjam	Format YYYY-MM-DD
4	Jatuh Tempo	Format YYYY-MM-DD (Default +7 hari)
5	Status Kembali	0 (Belum), 1 (Sudah)

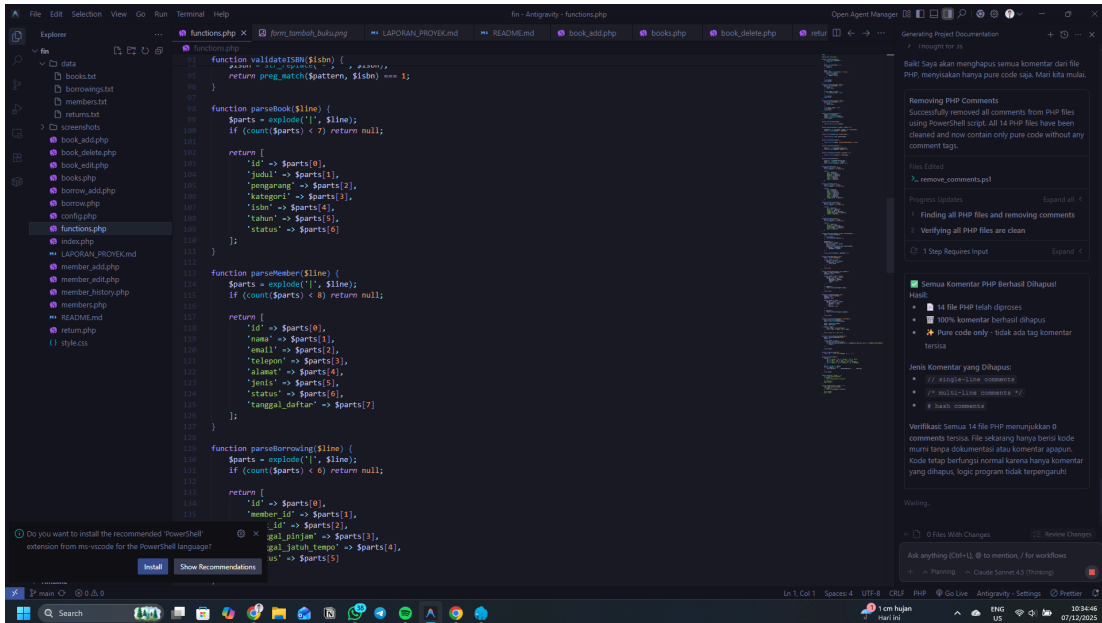
3.4 Tabel Pengembalian (returns.txt)

Mencatat detail penyelesaian transaksi, terutama denda.

Indeks	Atribut	Deskripsi
:---	:---	:---
0	ID Return	Primary Key (R-1)
1	ID Loan	Foreign Key ke loans.txt
2	Tgl Kembali	Tanggal aktual pengembalian
3	Denda	Nominal denda (Integer)

4. Implementasi Logika Bisnis (Backend)

Seluruh logika inti berada di file functions.php.



```
function validateISBN($isbn) {
    $pattern = '/^(9[78])?([1-3])?([1-9])?([0-9])?([0-9])?([0-9])?([0-9])?([0-9])?([0-9])?$/';
    return preg_match($pattern, $isbn) === 1;
}

function parseBook($line) {
    $parts = explode('|', $line);
    if (count($parts) < 7) return null;

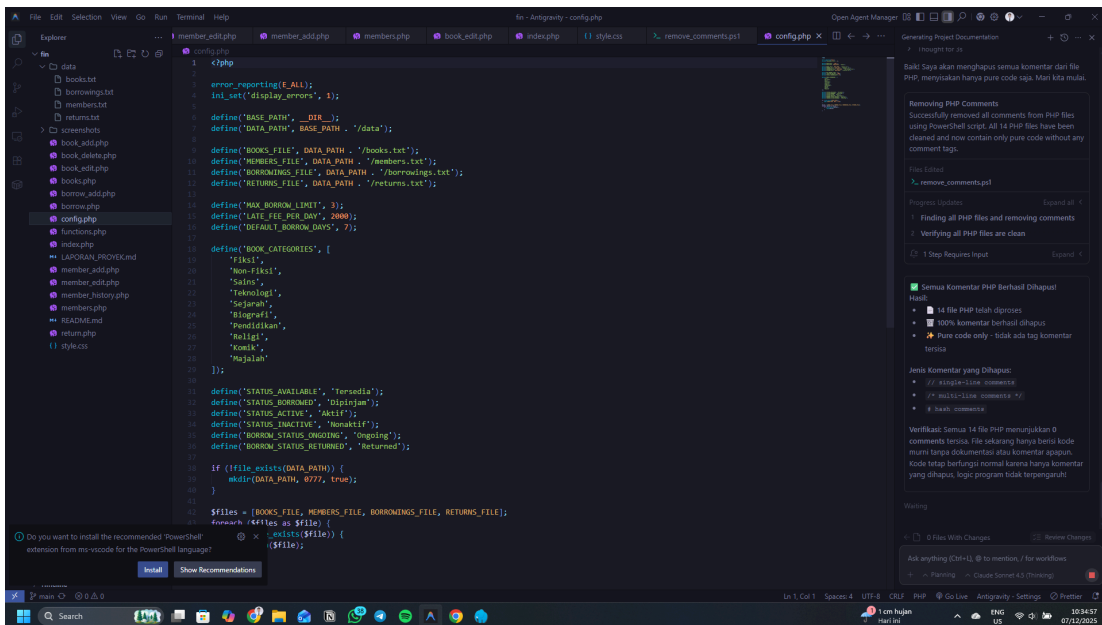
    return [
        'nama' => $parts[0],
        'judul' => $parts[1],
        'pengarang' => $parts[2],
        'kategori' => $parts[3],
        'isbn' => $parts[4],
        'tahun' => $parts[5],
        'status' => $parts[6]
    ];
}

function parseMember($line) {
    $parts = explode('|', $line);
    if (count($parts) < 8) return null;

    return [
        'id' => $parts[0],
        'nama' => $parts[1],
        'email' => $parts[2],
        'telepon' => $parts[3],
        'alamat' => $parts[4],
        'jenis' => $parts[5],
        'status' => $parts[6],
        'tanggal_dikirim' => $parts[7]
    ];
}

function parseBorrowing($line) {
    $parts = explode('|', $line);
    if (count($parts) < 6) return null;

    return [
        'id' => $parts[0],
        'member_id' => $parts[1],
        'id' => $parts[2],
        'tgl_pinjam' => $parts[3],
        'tgl_kembali' => $parts[4],
        'us' => $parts[5]
    ];
}
```



```
<?php
error_reporting(E_ALL);
ini_set('display_errors', 1);

define('BASE_PATH', __DIR__);
define('DATA_PATH', BASE_PATH . '/data');

define('BOOKS_FILE', DATA_PATH . '/books.txt');
define('MEMBERS_FILE', DATA_PATH . '/members.txt');
define('BORROWINGS_FILE', DATA_PATH . '/borrowings.txt');
define('RETURNS_FILE', DATA_PATH . '/returns.txt');

define('MAX_BORROW_LIMIT', 3);
define('LATE_FEE_PER_DAY', 2000);
define('DEFAULT_BORROW_DAYS', 7);

define('BOOK_CATEGORIES', [
    'Fiksi',
    'Non-Fiksi',
    'Sains',
    'Teknologi',
    'Sejarah',
    'Etnografi',
    'Pendidikan',
    'Religi',
    'Kuliah',
    'Majalah'
]);

define('STATUS_AVAILABLE', 'tersedia');
define('STATUS BORROWED', 'dipinjam');
define('STATUS_ACTIVE', 'aktif');
define('STATUS_INACTIVE', 'nonaktif');
define('BORROW_STATUS_Ongoing', 'Ongoing');
define('BORROW_STATUS_RETURNED', 'Returned');

if (!file_exists(DATA_PATH)) {
    mkdir(DATA_PATH, 0777, true);
}

$files = [BOOKS_FILE, MEMBERS_FILE, BORROWINGS_FILE, RETURNS_FILE];
foreach ($files as $file) {
    if (!file_exists($file)) {
        touch($file);
    }
}
```

4.1 Mekanisme Baca/Tulis Data

Sistem menggunakan fungsi PHP native file() untuk membaca baris menjadi array, dan implode()/explode() untuk memarsing data.

- **Read:** File teks dibaca per baris, setiap baris dipecah berdasarkan delimiter |.
- **Write:** Array data digabung kembali dengan delimiter | dan ditulis ulang ke file menggunakan fopen() mode w (write) atau a (append).

4.2 Aturan Bisnis (Business Rules)

Sistem menerapkan aturan ketat pada modul loans.php:

1. **Validasi Stok:** Sistem mengecek status buku di books.txt. Jika status dipinjam, buku tidak muncul di opsi peminjaman.
2. **Limit Pinjaman:** Setiap anggota dibatasi maksimal meminjam **3 buku** secara bersamaan. Logika:

// Pseudocode Pengecekan

```
$countBorrowed = 0;
```

```
foreach ($loans as $l) {
```

```
    if ($l[MemberID] == $InputID && $l[Status] == 'Belum Kembali') {
```

```
        $countBorrowed++;
```

```
    }
```

```
}
```

```
if ($countBorrowed >= 3) stop_process();
```

3. Kalkulasi Denda:

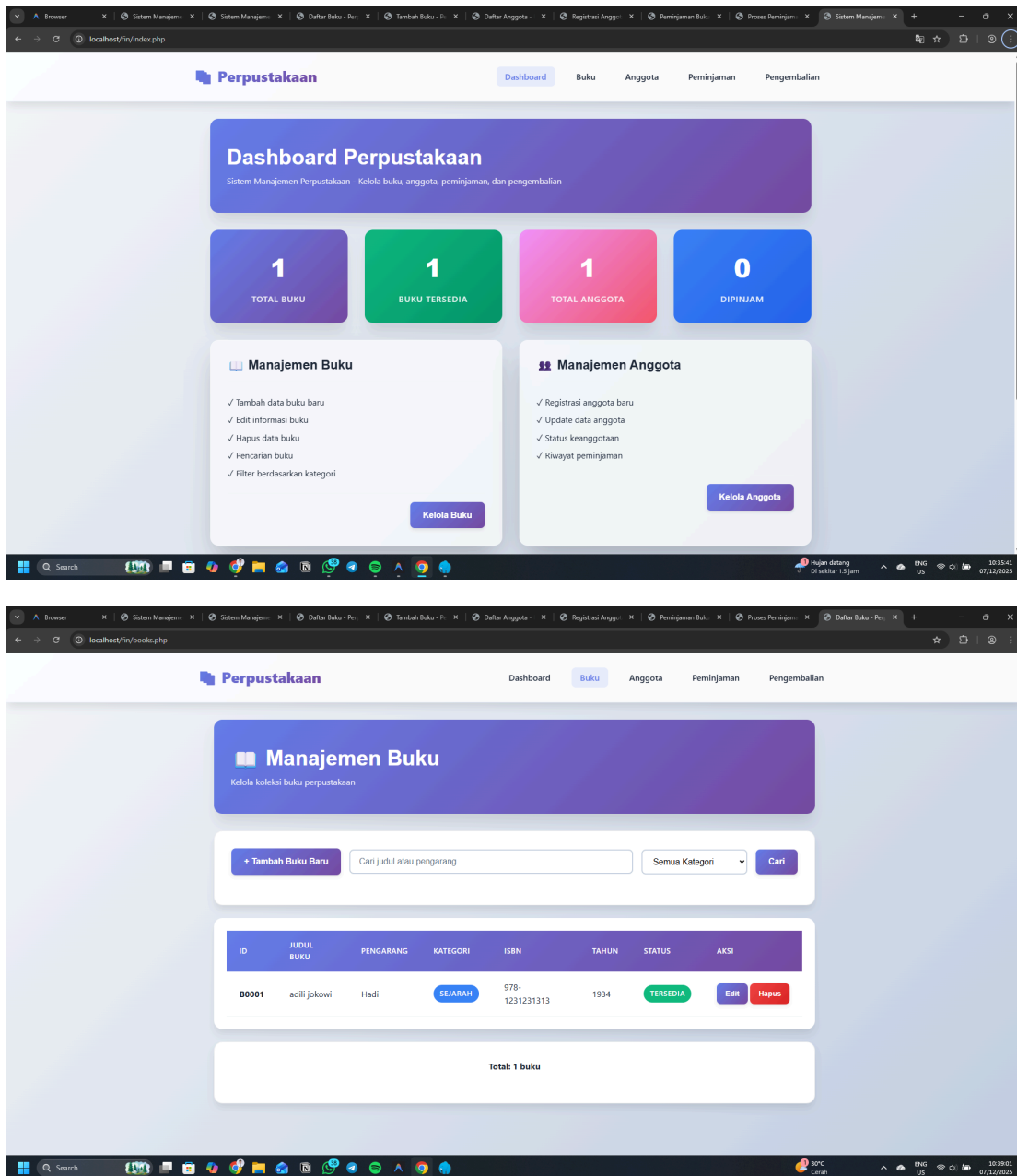
- Konfigurasi Denda: Rp 1.000 / hari.
- Rumus: (Tanggal Kembali - Jatuh Tempo) * Denda Per Hari.
- Jika selisih hari negatif (kembali lebih cepat), denda = 0.

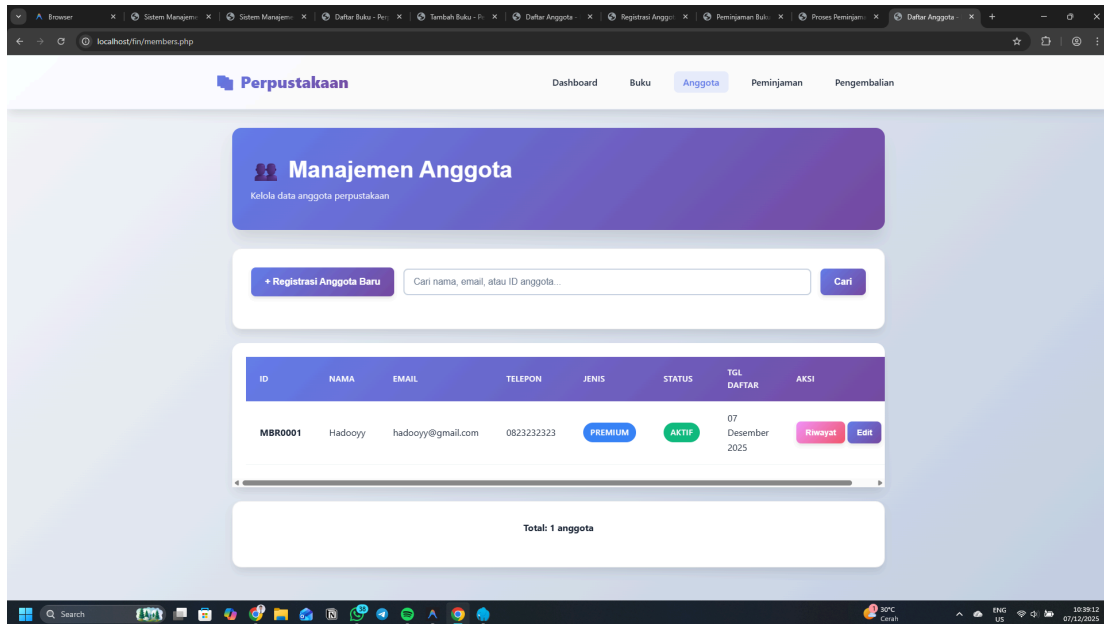
4.3 Fitur Pencarian & Filter

Pada books.php, fitur pencarian diimplementasikan menggunakan `array_filter` dan `stripos` untuk pencarian string *case-insensitive* tanpa membebani server dengan query berat.

5. Antarmuka Pengguna (UI/UX)

Desain antarmuka dibuat menggunakan CSS murni (style.css) tanpa framework eksternal (seperti Bootstrap) untuk menjaga beban file tetap ringan.





- **Tema Warna:** Menggunakan palet warna *Dark Blue* (#071029) untuk kesan modern dan nyaman di mata.
- **Komponen UI:**
 - **Card:** Kotak kontainer transparan dengan efek *glass morphism* tipis.
 - **Tabel:** Tabel responsive dengan border minimalis.
 - **Alert:** Notifikasi sukses/gagal dengan kode warna (Merah untuk error, default untuk info).
- **Responsivitas:** Menggunakan flexbox dan unit relatif (% , max-width) sehingga dapat diakses melalui perangkat *mobile*.

6. Kesimpulan

Proyek ini berhasil membuktikan bahwa sistem manajemen data relasional sederhana dapat dibangun tanpa database SQL. Penggunaan **Flat-File** efektif untuk skala kecil dan pembelajaran logika dasar pemrograman.

Kelebihan Sistem:

1. **Portable:** Cukup copy-paste folder, sistem langsung jalan (tidak perlu export/import SQL).
2. **Ringan:** Tidak memakan resource memory database server.
3. **Transparan:** Data bisa dibaca langsung lewat text editor.

Pengembangan Selanjutnya:

Untuk skala produksi yang lebih besar, sistem ini disarankan dimigrasikan ke MySQL/SQLite untuk menangani konkurensi data (akses bersamaan) dan keamanan data yang lebih baik.