



Department of Digital Business

Journal of Artificial Intelligence and Digital Business (RIGGS)

Homepage: <https://journal.ilmudata.co.id/index.php/RIGGS>

Vol. 4 No. 4 (2026) pp: 11648-11662

P-ISSN: 2963-9298, e-ISSN: 2963-914X

Perancangan Aplikasi Peminjaman Alat Rumah Tangga Dengan Metode V-Model Berbasis Web

Aditya Alviansyah*, Aris Munandar, Muhammad Ilham Ali
Sistem Informasi, Fakultas Ilmu Komputer, Universitas Pamulang
*adityaalviansyah05@gmail.com

Abstrak

Transformasi digital telah menjadi kebutuhan krusial untuk meningkatkan tata kelola aset komunitas di tingkat RT/RW. Penelitian ini bertujuan merancang dan mengimplementasikan sistem informasi peminjaman alat rumah tangga berbasis web guna mengatasi permasalahan sistem manual, seperti inefisiensi pencatatan, ketidakakuratan data, risiko kehilangan aset, dan kurangnya transparansi. Pengembangan sistem mengadopsi pendekatan V-Model yang dikombinasikan dengan pola iteratif, memastikan setiap tahapan mulai dari analisis kebutuhan, perancangan, hingga implementasi diverifikasi dan divalidasi secara sistematis. Perancangan konseptual menggunakan Use Case Diagram dan Entity Diagram (ERD), sementara implementasi teknis memanfaatkan stack teknologi PHP, Framework CodeIgniter 3, MySQL, dan Bootstrap 5 untuk membangun aplikasi yang stabil, aman, dan responsif. Pengujian fungsional menyeluruh dengan metode black-box terhadap 23 skenario fitur utama meliputi pengelolaan data barang, peminjaman, pengembalian, dan administrasi pengguna menunjukkan bahwa semua fungsi beroperasi dengan sukses sesuai ekspektasi. Evaluasi kinerja sistem kerangka PIECES (Performance, Information, Economy, Control, Efficiency, Service) mengonfirmasi adanya peningkatan signifikan dalam hal akurasi informasi, efisiensi waktu pengelolaan, pengendalian aset, dan kualitas pelayanan kepada warga. Aplikasi yang dihasilkan tidak hanya menyelesaikan masalah administratif namun juga berperan sebagai fondasi menuju tata kelola komunitas yang modern dan akuntabel. Untuk pengembangan lebih lanjut, studi masa depan dapat mengintegrasikan fitur notifikasi otomatis, memperkuat aspek keamanan siber, dan melakukan integrasi dengan platform digital lain di tingkat kelurahan guna menciptakan ekosistem layanan publik yang terpadu dan berkelanjutan.

Kata kunci: Sistem Informasi, Peminjaman Barang, RT/RW, V-Model, Aplikasi Berbasis Web.

1. Latar Belakang

Perkembangan teknologi informasi dan komunikasi pada era revolusi industri 4.0 telah mentransformasi tata kelola administrasi masyarakat, termasuk pada tingkat komunitas terkecil seperti RT dan RW. Pemanfaatan teknologi web terbukti mampu meningkatkan efisiensi, transparansi, dan akuntabilitas pelayanan publik di level komunitas [1]. Fenomena ini semakin menguat seiring dengan penetrasi internet di Indonesia yang terus meluas dan meningkatnya penggunaan perangkat mobile.

Dalam konteks kehidupan sosial komunitas di Kota Tangerang, muncul kebutuhan mendesak akan digitalisasi aktivitas kolektif, khususnya dalam pengelolaan peminjaman alat rumah tangga bersama. Alat-alat seperti kursi plastik, tenda, dan perlengkapan dapur untuk acara hajatan menjadi aset komunitas yang vital. Namun, praktik pengelolaan yang masih mengandalkan sistem manual melalui pencatatan konvensional telah memunculkan permasalahan sistemik berupa ketidakakuratan data, keterlambatan pengembalian, hingga kehilangan aset bersama [2];[3].

Sistem informasi berperan sebagai tulang punggung transformasi digital dalam pengelolaan aset komunitas, dengan fungsi mengelola data menjadi informasi bernilai melalui proses input, proses, output, penyimpanan, dan kontrol [4]. Dalam konteks peminjaman alat rumah tangga di lingkungan RT/RW, sistem informasi tidak hanya mendukung operasional harian tetapi juga berperan strategis dalam meningkatkan transparansi dan efisiensi pelayanan [5].

Pengembangan sistem ini mengadopsi metode V-Model yang menekankan hubungan sistematis antara tahapan pengembangan dengan proses verifikasi dan validasi [6]. Keunggulan utama metode ini terletak pada perencanaan pengujian sejak dini yang meminimalisasi kesalahan dan menjamin kualitas produk akhir [7]. Meskipun kurang fleksibel terhadap perubahan kebutuhan, V-Model dinilai tepat untuk proyek dengan kebutuhan jelas seperti sistem peminjaman alat rumah tangga.

Evaluasi sistem menggunakan framework PIECES yang mencakup enam aspek: Performance, Information, Economy, Control, Efficiency, dan Service [8]. Kerangka kerja ini memungkinkan penilaian komprehensif terhadap peningkatan kinerja sistem digital dibandingkan sistem manual, khususnya dalam akurasi data, penghematan biaya, dan kualitas pelayanan.

Implementasi teknologi web berbasis PHP, MySQL, dan Bootstrap dipilih karena kemudahan implementasi, biaya rendah, dan aksesibilitas tinggi [9,10]. Kombinasi teknologi ini menjamin sistem yang responsif, aman, dan mudah digunakan oleh masyarakat dengan variasi tingkat literasi digital.

Penelitian terdahulu oleh Nabilah et al. [11] dalam "Rancangan Sistem Peminjaman Barang Digital Berbasis Web" berhasil mengidentifikasi bahwa sistem manual menyebabkan hilangnya data, ketidaksesuaian jadwal peminjaman, dan kesulitan melacak barang. Namun, penelitian tersebut masih terbatas pada lingkungan kampus dan belum mengakomodasi karakteristik khusus komunitas RT/RW. Beberapa penelitian lain [12];[13] lebih berfokus pada konteks institusi pendidikan atau perkantoran, sehingga belum secara spesifik menjawab kebutuhan peminjaman alat rumah tangga di level komunitas.

Berdasarkan analisis celah tersebut, pengembangan aplikasi peminjaman alat rumah tangga berbasis web untuk lingkungan RT/RW memiliki urgensi dan kebaruan yang kuat. Pendekatan pengembangan sistem yang terstruktur melalui metode V-Model dipilih karena menekankan hubungan sistematis antara setiap tahapan pengembangan dengan proses verifikasi dan validasi yang ketat [14]. Integrasi pertimbangan keamanan sejak tahap desain menjadi aspek kritis mengingat kerentanan aplikasi web dapat dieksploitasi jika aspek keamanan tidak dipertimbangkan secara matang [15].

2. Metode Penelitian

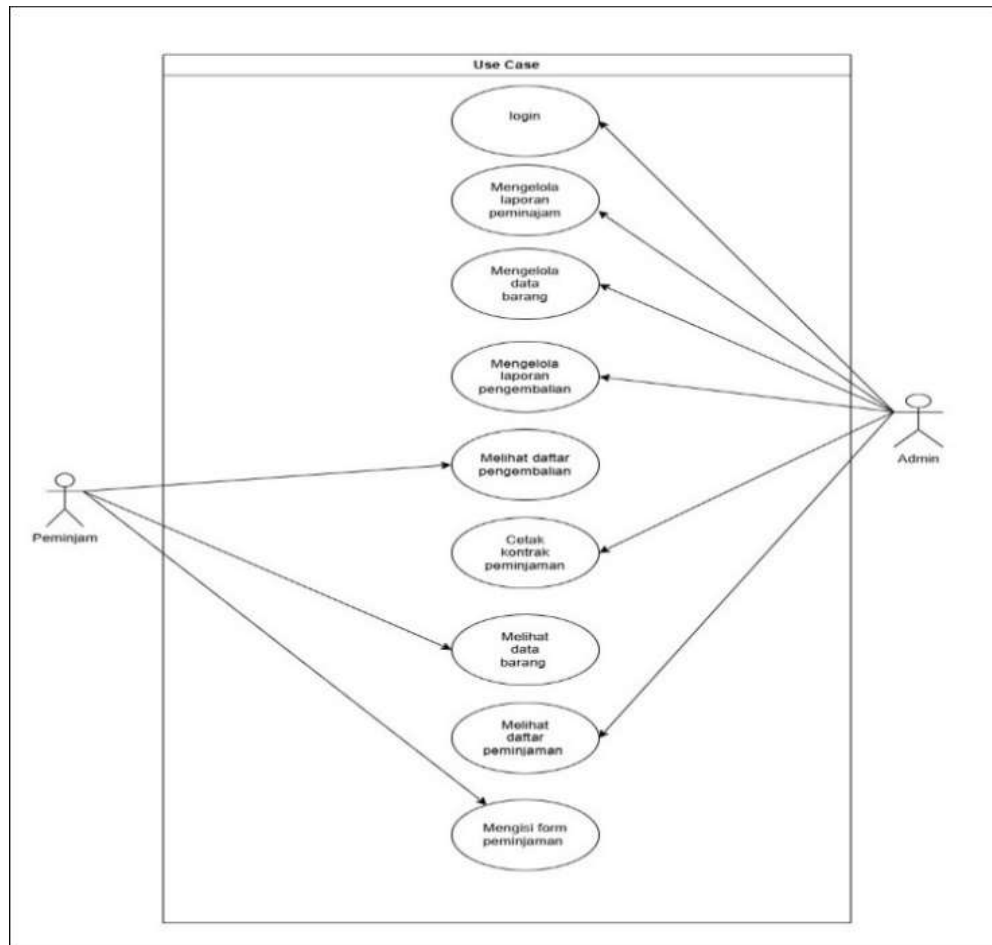
Penelitian ini menggunakan pendekatan kualitatif dengan metode studi kasus yang diterapkan pada lingkungan RT/RW di Kota Tangerang. Pendekatan ini dipilih untuk memahami secara mendalam proses pengelolaan peminjaman alat rumah tangga serta permasalahan yang muncul dalam praktiknya. Pengembangan sistem dilakukan menggunakan pendekatan V-Model yang dikombinasikan dengan pola iteratif berbasis sprint, sehingga setiap tahapan pengembangan dapat diverifikasi dan divalidasi secara berkelanjutan.

Hasil analisis kebutuhan menunjukkan bahwa sistem yang berjalan sebelumnya masih bersifat manual, sehingga menimbulkan berbagai kendala seperti rendahnya efisiensi kerja, risiko kehilangan data, keterbatasan transparansi, serta potensi kesalahan pencatatan. Oleh karena itu, sistem yang diusulkan dirancang berbasis web dengan kemampuan pengolahan data secara real-time, penyimpanan terpusat, serta antarmuka yang responsif dan mudah digunakan.

2.1 Perancangan Sistem

Tahap perancangan bertujuan menerjemahkan kebutuhan pengguna ke dalam bentuk rancangan teknis yang terstruktur. Proses ini memastikan bahwa sistem yang dikembangkan tidak hanya mampu menyelesaikan permasalahan, tetapi juga mudah digunakan, stabil, dan sesuai dengan kebutuhan operasional. Perancangan dilakukan melalui pemodelan visual untuk menggambarkan alur sistem dan interaksi pengguna sebelum implementasi dilakukan.

Salah satu model yang digunakan adalah *use case diagram*, yang berfungsi untuk menggambarkan hubungan interaksi antara pengguna dan sistem. Diagram ini menunjukkan aktivitas yang dapat dilakukan oleh setiap aktor serta respons sistem terhadap aktivitas tersebut. Melalui pendekatan ini, kebutuhan fungsional dapat diidentifikasi secara jelas sehingga meminimalkan kesalahan perancangan serta mempermudah proses evaluasi dan pengembangan lanjutan [16].



Gambar 1. Use Case Diagram

Pada Gambar 1 ditunjukkan bahwa terdapat dua aktor utama, yaitu admin dan peminjam. Peminjam memiliki hak untuk mengisi formulir peminjaman, melihat daftar barang yang tersedia, serta memantau status peminjaman. Sementara itu, admin memiliki kewenangan lebih luas, seperti mengelola data barang, memproses peminjaman dan pengembalian, mengelola laporan, serta mencetak dokumen peminjaman. Pembagian peran ini bertujuan untuk menjaga keamanan data dan keteraturan sistem.

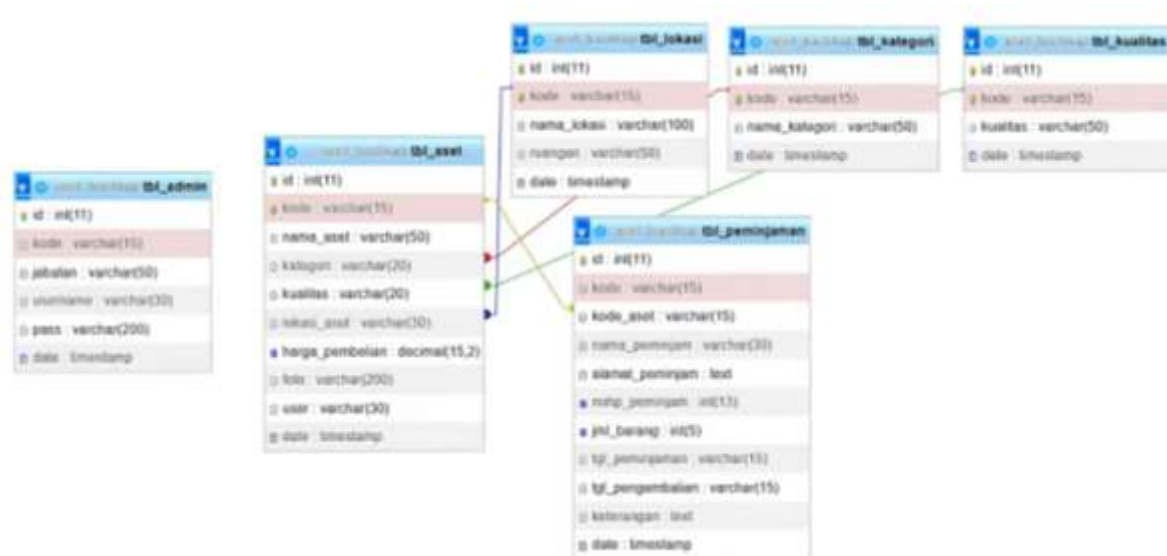
2.2 Perancangan Basis Data

Perancangan basis data dilakukan menggunakan *Entity Relationship Diagram (ERD)* sebagai model konseptual untuk menggambarkan hubungan antarentitas dalam sistem. ERD membantu memvisualisasikan struktur data serta hubungan logis antar tabel sehingga memudahkan pengelolaan data secara terorganisir dan konsisten [17].

Entitas utama yang digunakan meliputi User, Alat, Peminjaman, Detail_Peminjaman, Pengembalian, dan Laporan. Setiap entitas memiliki fungsi spesifik dalam mendukung alur proses peminjaman dan pengelolaan aset.

Tabel 1. Perancangan Basis Data (ERD)

Entitas	Atribut Utama	Keterangan
User	id_user	Menyimpan data pengguna sistem, baik admin maupun warga.
Alat	id_alat	Menyimpan data alat yang dapat dipinjam.
Peminjaman	id_peminjaman	Mencatat transaksi peminjaman alat.
Detail_Peminjaman	id_detail	Menghubungkan data peminjaman dan alat (many-to-many).
Pengembalian	id_pengembalian	Mencatat data pengembalian dan kondisi barang.
Laporan	id_laporan	Menyimpan rekap transaksi berdasarkan periode tertentu.



Gambar 2. Entity Relationship Diagram

Gambar 2 menampilkan struktur hubungan antarentitas dalam sistem peminjaman alat. Tabel *tbl_admin* berfungsi menyimpan data pengguna yang memiliki hak akses pengelolaan sistem. Tabel *tbl_aset* menyimpan informasi aset, seperti nama, kategori, kualitas, lokasi, dan harga pembelian. Untuk menjaga konsistensi data, informasi kategori, kualitas, dan lokasi dikelola melalui tabel terpisah, yaitu *tbl_kategori*, *tbl_kualitas*, dan *tbl_lokasi*.

Data peminjaman dicatat pada tabel *tbl_peminjaman*, yang memuat informasi peminjam, jumlah barang, serta tanggal peminjaman dan pengembalian. Relasi antar tabel memungkinkan sistem melakukan pelacakan aset secara akurat dan terintegrasi. Struktur ini mendukung pengelolaan data yang efisien, meminimalkan kesalahan pencatatan, serta meningkatkan akurasi informasi.

2.3 Implementasi Sistem

Tahap implementasi merupakan proses penerapan rancangan sistem ke dalam bentuk aplikasi yang dapat digunakan secara nyata. Pada tahap ini, seluruh kebutuhan infrastruktur dipersiapkan, baik perangkat keras maupun perangkat lunak. Sistem dikembangkan menggunakan bahasa pemrograman PHP, basis data MySQL, serta framework CodeIgniter 3 untuk memastikan struktur kode yang rapi, aman, dan mudah dikembangkan [18]; [19].

Penggunaan kombinasi teknologi tersebut bertujuan untuk menjamin stabilitas sistem, kemudahan pemeliharaan, serta kinerja aplikasi yang optimal. Hasil implementasi ditunjukkan melalui potongan *source code* yang merepresentasikan proses utama sistem, seperti pengelolaan data aset dan peminjaman.

Tabel 2. Data barang

```

<?php
namespace App\Models;
use Illuminate\Database\Eloquent\Model;
class Aset extends Model
{
    protected $table = 'aset';
    public $timestamps = false; // karena pakai date, bukan created_at
    protected $fillable = [
        'kode',
        'nama_aset',
        'kategori',
        'kualitas',
        'lokasi_aset',
        'harga_pembelian',
        'foto',
        'user',
    ];
}
    
```

```
public function kategori()
{
    return $this->belongsTo(Kategori::class, 'kategori', 'kode');
}
public function kualitas()
{
    return $this->belongsTo(Kualitas::class, 'kualitas', 'kode');
}
public function lokasi()
{
    return $this->belongsTo(Lokasi::class, 'lokasi_aset', 'kode');
}
public function peminjaman()
{
    return $this->hasMany(Peminjaman::class, 'kode_aset', 'kode');
}
}
```

Tabel 2 menyajikan struktur data barang yang digunakan dalam sistem peminjaman. Implementasi tabel ini direpresentasikan melalui *model* Aset yang berfungsi untuk mengelola data aset secara terintegrasi dalam basis data. Model tersebut menggunakan tabel aset sebagai sumber penyimpanan utama dan memuat sejumlah atribut penting, seperti kode aset, nama barang, kategori, kualitas, lokasi penyimpanan, harga pembelian, foto barang, serta informasi pengguna yang melakukan input data.

Relasi antarentitas dibangun untuk menjaga konsistensi dan keterkaitan data. Model Aset terhubung dengan tabel Kategori, Kualitas, dan Lokasi melalui relasi *belongsTo*, sehingga setiap aset dapat diklasifikasikan secara jelas berdasarkan kategori, mutu, dan lokasi penyimpanannya. Selain itu, relasi *hasMany* dengan tabel Peminjaman memungkinkan sistem mencatat riwayat peminjaman setiap aset secara terstruktur.

Secara fungsional, Tabel 1 digunakan untuk menampilkan informasi barang dalam bentuk tabel yang terdiri atas kolom nomor, kode barang, nama barang, kategori, kualitas, lokasi barang, harga pembelian, toko pembelian, foto barang, serta opsi pengelolaan data. Struktur ini dirancang untuk memudahkan proses pengelolaan, pencarian, dan pemantauan aset, sekaligus mendukung akurasi data dalam sistem peminjaman barang.

Tabel 3. Tambah data barang

```
<?php
namespace App\Http\Controllers;
use App\Models\Aset;
use App\Models\Kategori;
use App\Models\Kualitas;
use App\Models\Lokasi;
use Illuminate\Http\Request;
use Illuminate\Support\Facades\Storage;
class AsetController extends Controller
{
    public function index()
    {
        $data = Aset::with(['kategori','kualitas','lokasi'])
            ->orderBy('id','desc')
            ->get();
        return view('aset.index', compact('data'));
    }
    public function create()
    {
        $lastAset = Aset::orderBy('id', 'desc')->first();
        if ($lastAset) {
            $lastNumber = (int) str_replace('AST-', '', $lastAset->kode);
            $newKode = 'AST-' . ($lastNumber + 1);
        } else {
            $newKode = 'AST-1';
        }
        return view('aset.create', [
            'kategori' => Kategori::all(),
            'kualitas' => Kualitas::all(),
            'lokasi'   => Lokasi::all(),
            'kode'     => $newKode,
        ]);
    }
}
```

```
}  
public function store(Request $request)  
{  
    $request->validate([  
        'nama_aset' => 'required',  
        'kategori' => 'required',  
        'kualitas' => 'required',  
        'lokasi_aset' => 'required',  
        'foto' => 'nullable|image|max:2048',  
    ]);  
    $lastAset = Aset::where('kode', 'like', 'AST-%')  
        ->orderBy('id', 'desc')  
        ->first();  
    if ($lastAset) {  
        $lastNumber = (int) str_replace('AST-', '', $lastAset->kode);  
        $kode = 'AST-' . ($lastNumber + 1);  
    } else {  
        $kode = 'AST-1';  
    }  
    $data = $request->only([  
        'nama_aset',  
        'kategori',  
        'kualitas',  
        'lokasi_aset',  
        'harga_pembelian',  
    ]);  
    $data['kode'] = $kode;  
    $data['user'] = auth()->user()->username ?? 'admin';  
    if ($request->hasFile('foto')) {  
        $data['foto'] = $request->file('foto')->store('aset','public');  
    }  
    Aset::create($data);  
    return redirect()  
        ->route('aset.index')  
        ->with('success','Aset berhasil ditambahkan');  
}  
public function edit($id)  
{  
    return view('aset.edit', [  
        'aset' => Aset::findOrFail($id),  
        'kategori' => Kategori::all(),  
        'kualitas' => Kualitas::all(),  
        'lokasi' => Lokasi::all(),  
    ]);  
}  
public function update(Request $request, $id)  
{  
    $aset = Aset::findOrFail($id);  
    $request->validate([  
        'kode' => 'required|unique:aset,kode,' . $aset->id,  
        'nama_aset' => 'required',  
        'kategori' => 'required',  
        'kualitas' => 'required',  
        'lokasi_aset' => 'required',  
        'foto' => 'nullable|image|max:2048',  
    ]);  
    $data = $request->only([  
        'kode',  
        'nama_aset',  
        'kategori',  
        'kualitas',  
        'lokasi_aset',  
        'harga_pembelian',  
    ]);  
    if ($request->hasFile('foto')) {  
        if ($aset->foto) {  
            Storage::disk('public')->delete($aset->foto);  
        }  
        $data['foto'] = $request->file('foto')->store('aset','public');  
    }  
    $aset->update($data);  
    return redirect()  
}
```

```
->route('aset.index')
->with('success','Aset berhasil diperbarui');
}
public function destroy($id)
{
    $aset = Aset::findOrFail($id);
    if ($aset->foto) {
        Storage::disk('public')->delete($aset->foto);
    }
    $aset->delete();
    return redirect()
        ->route('aset.index')
        ->with('success','Aset berhasil dihapus');
}
}
```

Tabel 3 menggambarkan proses penambahan data barang ke dalam sistem melalui modul *controller* yang berfungsi mengelola seluruh aktivitas input, penyimpanan, dan pembaruan data aset. Proses ini diawali dengan pengambilan data referensi seperti kategori, kualitas, dan lokasi yang digunakan sebagai pilihan pada formulir input. Sistem secara otomatis menghasilkan kode aset baru berdasarkan data terakhir yang tersimpan, sehingga memastikan keunikan setiap kode barang.

Pada tahap penyimpanan, sistem melakukan validasi terhadap data yang dimasukkan, meliputi nama aset, kategori, kualitas, lokasi, serta file gambar apabila disertakan. Data yang telah lolos validasi kemudian disimpan ke dalam basis data, termasuk informasi harga pembelian dan identitas pengguna yang melakukan input. File gambar aset dikelola melalui penyimpanan terpisah untuk menjaga efisiensi dan keamanan data. Selain fungsi penambahan data, modul ini juga mendukung proses pengeditan dan penghapusan data aset. Setiap perubahan data akan diperbarui secara langsung pada basis data, termasuk pengelolaan ulang file gambar apabila terjadi pembaruan. Dengan mekanisme ini, sistem mampu menjaga konsistensi data serta mendukung pengelolaan aset secara terstruktur dan berkelanjutan.

Tabel 4. Tambah data peminjam

```
<?php
namespace App\Http\Controllers;
use App\Models\Peminjaman;
use App\Models\Aset;
use Illuminate\Http\Request;
use Barryvdh\DomPDF\Facade\Pdf;
class PeminjamanController extends Controller
{
    public function index()
    {
        $data = Peminjaman::with('aset')
            ->orderBy('id', 'desc')
            ->get();
        return view('peminjaman.index', compact('data'));
    }
    public function create()
    {
        $last = Peminjaman::orderBy('id', 'desc')->first();
        if ($last) {
            $lastNumber = (int) str_replace('PMJ-', '', $last->kode);
            $kode = 'PMJ-' . ($lastNumber + 1);
        } else {
            $kode = 'PMJ-1';
        }
        return view('peminjaman.create', [
            'aset' => Aset::all(),
            'kode' => $kode
        ]);
    }
    public function store(Request $request)
    {
        $request->validate([
            'kode_aset' => 'required',
            'nama_peminjam' => 'required',
            'jml_barang' => 'required|numeric|min:1',
        ]);
    }
}
```

```
'tgl_peminjaman' => 'required|date',
'tgl_pengembalian' => 'nullable|date',
]);
// ambil kode terakhir
$last = Peminjaman::orderBy('id', 'desc')->first();
$nextNumber = $last
    ? ((int) substr($last->kode, 4)) + 1
    : 1;
$kode = 'PMJ-' . str_pad($nextNumber, 3, '0', STR_PAD_LEFT);
Peminjaman::create([
    'kode' => $kode,
    'kode_aset' => $request->kode_aset,
    'nama_peminjam' => $request->nama_peminjam,
    'alamat_peminjam' => $request->alamat_peminjam,
    'nohp_peminjam' => $request->nohp_peminjam,
    'jml_barang' => $request->jml_barang,
    'tgl_peminjaman' => $request->tgl_peminjaman,
    'tgl_pengembalian' => $request->tgl_pengembalian,
    'keterangan' => $request->keterangan,
]);
return redirect()
    ->route('peminjaman.index')
    ->with('success', 'Peminjaman berhasil disimpan');
}
public function edit($id)
{
    return view('peminjaman.edit', [
        'peminjaman' => Peminjaman::findOrFail($id),
        'aset' => Aset::all()
    ]);
}
public function update(Request $request, $id)
{
    $peminjaman = Peminjaman::findOrFail($id);
    $request->validate([
        'kode_aset' => 'required',
        'nama_peminjam' => 'required',
        'jml_barang' => 'required|numeric|min:1',
        'tgl_peminjaman' => 'required|date',
    ]);
    // kode TIDAK diupdate
    $peminjaman->update([
        'kode_aset' => $request->kode_aset,
        'nama_peminjam' => $request->nama_peminjam,
        'alamat_peminjam' => $request->alamat_peminjam,
        'nohp_peminjam' => $request->nohp_peminjam,
        'jml_barang' => $request->jml_barang,
        'tgl_peminjaman' => $request->tgl_peminjaman,
        'tgl_pengembalian' => $request->tgl_pengembalian,
        'keterangan' => $request->keterangan,
    ]);
    return redirect()
        ->route('peminjaman.index')
        ->with('success', 'Peminjaman berhasil diperbarui');
}
public function destroy($id)
{
    Peminjaman::findOrFail($id)->delete();
    return redirect()
        ->route('peminjaman.index')
        ->with('success', 'Peminjaman berhasil dihapus');
}
public function cetak(Request $request)
{
    $request->validate([
        'ids' => 'required|array|min:1'
    ]);
    $data = Peminjaman::with('aset')
        ->whereIn('id', $request->ids)
        ->get();
    if ($data->isEmpty()) {
        return back()->with('error', 'Data tidak ditemukan');
```

```

    }
    $pdf = Pdf::loadView('peminjaman.cetak', compact('data'))
    ->setPaper('A4', 'portrait');
    return $pdf->stream('surat-peminjaman.pdf');
  }
}

```

Tabel 4 menampilkan proses pengelolaan data peminjaman barang yang diimplementasikan melalui modul *controller* pada sistem. Modul ini berfungsi untuk mengatur seluruh aktivitas pencatatan peminjaman, mulai dari penambahan data, pembaruan, hingga penghapusan data peminjam. Sistem secara otomatis menghasilkan kode peminjaman unik berdasarkan data terakhir yang tersimpan, sehingga memastikan setiap transaksi memiliki identitas yang berbeda dan mudah ditelusuri.

Pada tahap input, pengguna mengisi data peminjam yang meliputi nama peminjam, alamat, nomor telepon, jumlah barang yang dipinjam, tanggal peminjaman, tanggal pengembalian, serta keterangan tambahan. Sistem juga mengaitkan data peminjaman dengan aset yang tersedia untuk memastikan keterpaduan informasi antar tabel. Validasi diterapkan untuk menjamin kelengkapan dan ketepatan data sebelum disimpan ke dalam basis data.

Selain fungsi penambahan data, modul ini mendukung proses pengeditan dan penghapusan data peminjaman secara terkontrol. Sistem juga menyediakan fitur pencetakan laporan peminjaman dalam format dokumen digital untuk keperluan dokumentasi. Dengan mekanisme tersebut, proses pengelolaan data peminjaman menjadi lebih tertata, akurat, dan mendukung transparansi dalam pengelolaan aset komunitas.

2.4 Pengujian Sistem

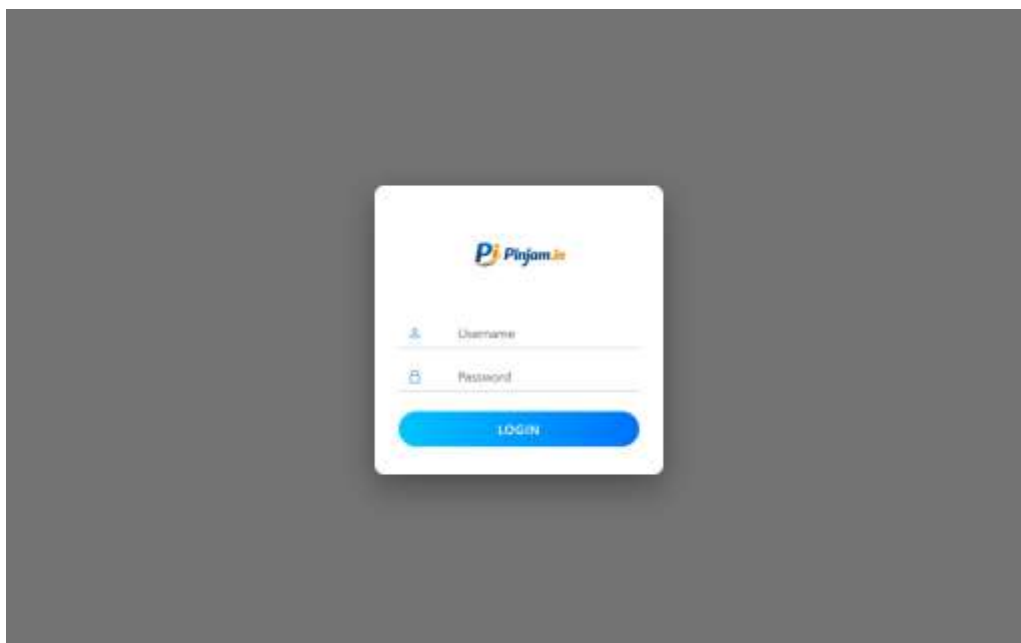
Pada Tahap pengujian sistem dilakukan untuk memastikan semua fitur telah memenuhi kebutuhan pengguna dan berfungsi sesuai spesifikasi. Metode *black-box testing* digunakan untuk menguji fungsionalitas sistem berdasarkan input dan output tanpa memeriksa kode sumber. Pendekatan ini mengevaluasi kinerja fitur, seperti validasi data dan pengelolaan informasi, dari sudut pandang pengguna, sehingga potensi kesalahan dapat diidentifikasi lebih awal sebelum sistem dioperasikan secara luas.

Tabel 5. *Blackbox-testing*

NO	Fitur	Skenario Pengujian	Hasil Yang Diharapkan	Status
1	Menu Login	Masukan Username dan Password yang sesuai	Dapat masuk ke halaman dashboard web	Sukses
2	Menu Logout	Berhasil Keluar dari website	Kembali ke menu login	Sukses
3	Tambah Barang	Menambahkan data barang baru	Data Barang Berhasil ditambahkan	Sukses
4	Edit Barang	Mengedit data barang yang sudah ada	Data barang berhasil diperbarui	Sukses
5	Hapus Barang	Menghapus data barang yang sudah ada	Data barang berhasil di hapus	Sukses
6	Pinjam Barang	Menambahkan data peminjam baru	Peminjam baru berhasil ditambahkan	Sukses
7	Edit Data Peminjam	Mengedit data peminjam yang sudah ada	Data peminjam berhasil diperbarui	Sukses
8	Detail Data Peminjam	Menampilkan informasi data dari peminjam	Data peminjam berhasil ditampilkan	Sukses
9	Hapus Data Peminjam	Menghapus data peminjam yang sudah ada	Data peminjam berhasil dihapus	Sukses
10	Cetak Surat Peminjam	Mencetak surat peminjaman bentuk PDF	Surat peminjaman berhasil diperbarui	Sukses
11	Pencarian Data Peminjam	Mencari data peminjam yang ingin di cari	Hanya barang sesuai input yang ditampilkan	Sukses

12	Tambah Kategori Barang	Menambahkan data kategori barang baru	Data kategori barang berhasil ditambahkan	Sukses
13	Edit Kategori	Mengedit data kategori yang sudah ada	Data kategori berhasil diperbarui	Sukses
14	Hapus Kategori	Menghapus data kategori yang sudah ada	Data kategori berhasil dihapus	Sukses
15	Tambah Kualitas Barang	Menambahkan data kualitas barang baru	Data kualitas barang berhasil ditambahkan	Sukses
16	Edit Kualitas	Mengedit data kualitas yang sudah ada	Data kualitas berhasil diperbarui	Sukses
17	Hapus Kualitas	Menghapus data kualitas yang sudah ada	Data kualitas berhasil dihapus	Sukses
18	Tambah Lokasi Barang	Menambahkan data lokasi barang baru	Data lokasi barang berhasil ditambahkan	Sukses
19	Edit Lokasi	Mengedit data lokasi yang sudah ada	Data lokasi berhasil diperbarui	Sukses
20	Hapus Lokasi	Menghapus data lokasi yang sudah ada	Data lokasi berhasil dihapus	Sukses
21	Tambah Admin	Menambahkan data admin yang baru	Data admin berhasil ditambahkan	Sukses
22	Edit Admin	Mengedit user dan password admin yang sudah ada	Data admin berhasil diperbarui	Sukses
23	Hapus Admin	Menghapus data admin yang sudah ada	Data admin berhasil dihapus	Sukses

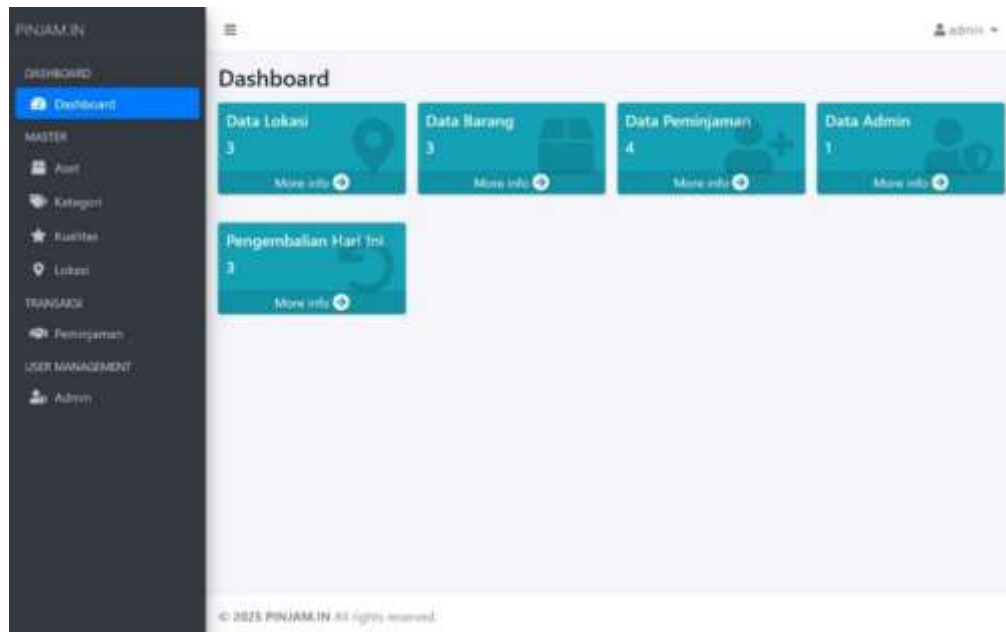
3. Hasil dan Diskusi



Gambar 3. Halaman Login

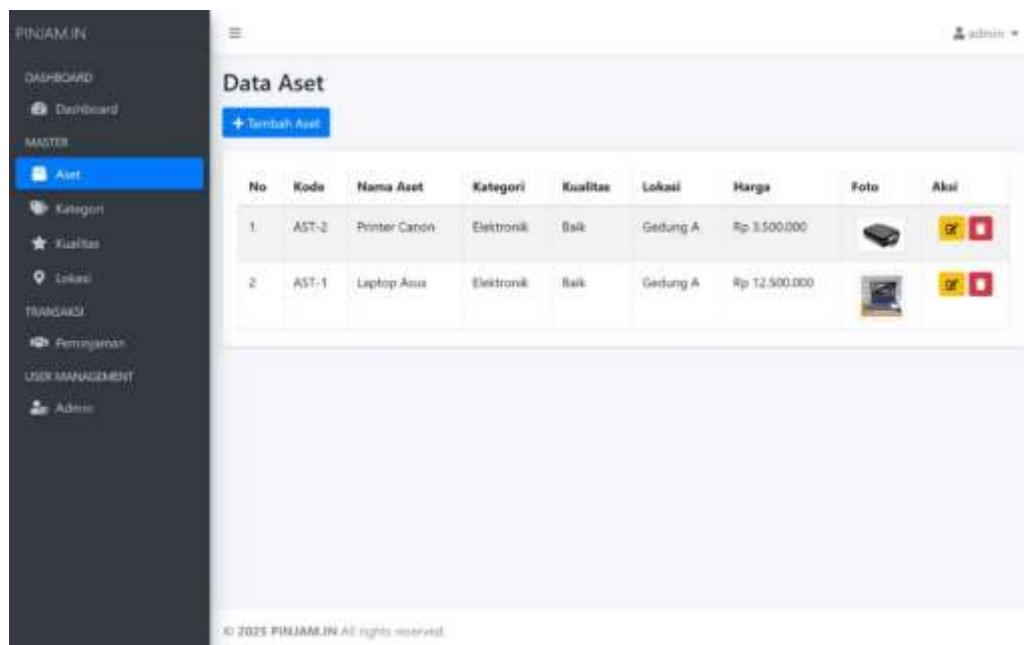
Gambar 3 menampilkan halaman awal sistem yang digunakan sebagai gerbang autentikasi pengguna sebelum mengakses fitur utama aplikasi. Halaman ini berfungsi untuk membatasi akses sistem agar hanya dapat digunakan

oleh pengguna yang memiliki hak otorisasi. Penerapan autentikasi ini bertujuan untuk menjaga keamanan data serta memastikan bahwa setiap aktivitas dalam sistem tercatat berdasarkan identitas pengguna yang sah.



Gambar 4. Halaman Dashboard

Gambar 4 menunjukkan tampilan dashboard yang menyajikan ringkasan informasi penting secara visual dan terstruktur. Pada halaman ini, pengguna dapat melihat gambaran umum aktivitas sistem serta mengakses fitur utama dengan cepat. Dashboard dirancang untuk meningkatkan efisiensi penggunaan sistem dengan menyediakan navigasi yang intuitif dan informasi yang mudah dipahami.



Gambar 5. Halaman Data Barang

Gambar 5 menampilkan halaman pengelolaan data barang yang berisi daftar inventaris aset. Informasi yang ditampilkan meliputi kode barang, nama barang, kategori, kualitas, lokasi penyimpanan, harga pembelian, serta foto barang. Pada halaman ini tersedia fitur untuk menambah, mengubah, menghapus, dan melakukan peminjaman

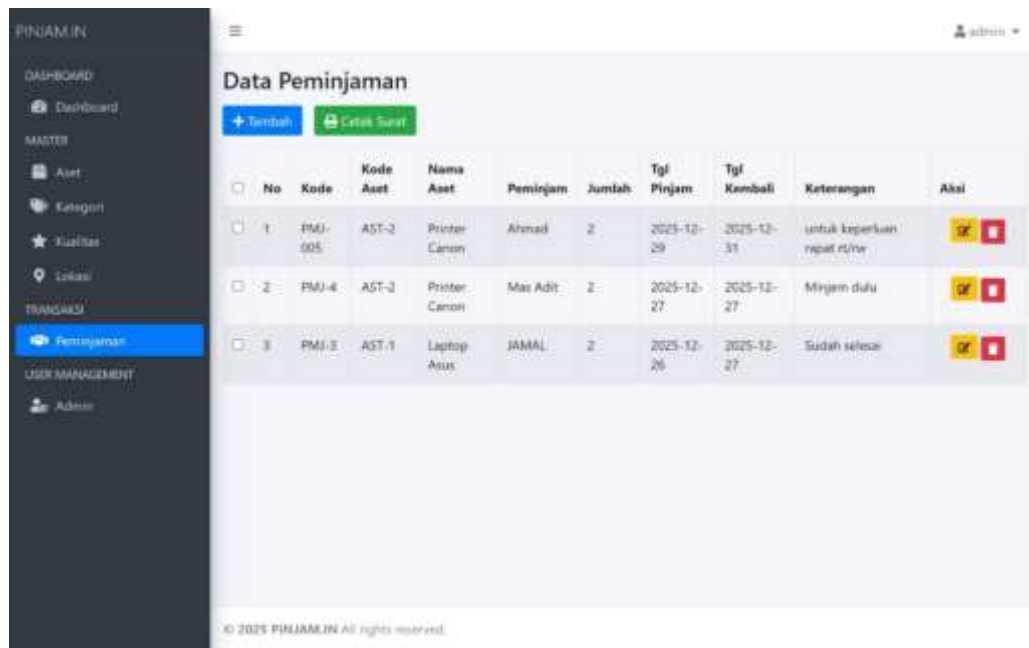
barang. Tampilan ini dirancang untuk memudahkan pengguna dalam memantau dan mengelola aset secara terstruktur dan efisien.

Gambar 6. Halaman Input Data Barang

Gambar 6 menunjukkan tampilan formulir input data barang yang digunakan untuk menambahkan data baru ke dalam sistem. Formulir ini memuat informasi penting seperti kode barang, nama barang, kategori, kualitas, lokasi penyimpanan, harga pembelian, serta unggahan foto barang. Data yang telah diinput akan tersimpan dan ditampilkan pada halaman data barang, sehingga mendukung proses pencatatan aset secara sistematis.

Gambar 7. Halaman Input Data Peminjam

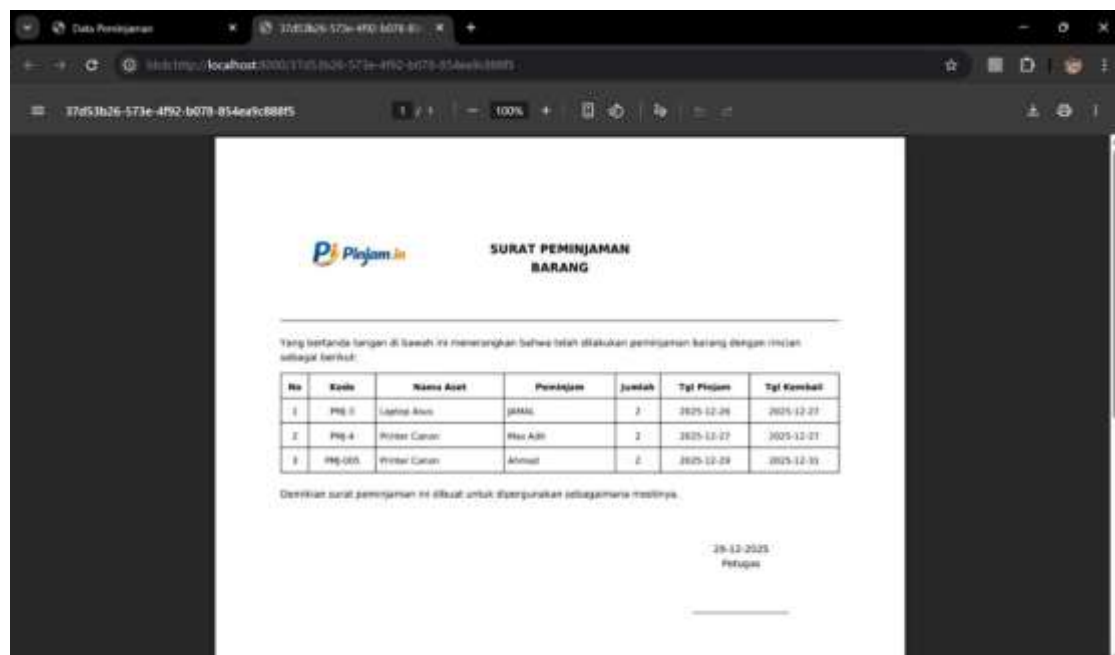
Gambar 7 menampilkan formulir pengisian data peminjam yang mencakup informasi nama peminjam, alamat, nomor telepon, tanggal peminjaman, tanggal pengembalian, serta keterangan tambahan. Data yang dimasukkan pada halaman ini akan tersimpan dalam sistem dan digunakan sebagai dasar pencatatan transaksi peminjaman barang.



No	Kode	Kode Asat	Nama Asat	Peminjam	Jumlah	Tgl Pinjam	Tgl Kembali	Keterangan	Aksi
1	PMJ-005	AST-2	Printer Canon	Ahmadi	2	2025-12-29	2025-12-31	untuk keperluan rapat rt/rw	 
2	PMJ-4	AST-2	Printer Canon	Max Adit	2	2025-12-27	2025-12-27	Minjam dulu	 
3	PMJ-3	AST-1	Laptop Asus	JAMAL	2	2025-12-26	2025-12-27	Sudah selesai	 

Gambar 8. Halaman Data Peminjaman Barang

Gambar 8 menyajikan tabel data peminjaman barang yang berisi informasi lengkap, seperti kode peminjaman, kode barang, nama peminjam, nomor kontak, alamat, jumlah barang, tanggal peminjaman, tanggal pengembalian, serta status peminjaman. Selain itu, tersedia fitur untuk mengedit data, melihat detail transaksi, dan memperbarui status peminjaman, sehingga memudahkan proses monitoring dan pengelolaan data secara menyeluruh.



SURAT PEMINJAMAN BARANG

Yang bertanda tangan di bawah ini menerangkan bahwa telah dilakukan peminjaman barang dengan rincian sebagai berikut:

No	Kode	Nama Asat	Peminjam	Jumlah	Tgl Pinjam	Tgl Kembali
1	PMJ-0	Laptop Asus	JAMAL	2	2025-12-26	2025-12-27
2	PMJ-4	Printer Canon	Max Adit	2	2025-12-27	2025-12-27
3	PMJ-005	Printer Canon	Ahmadi	2	2025-12-29	2025-12-31

Dengan surat peminjaman ini dibuat untuk dipergunakan sebagaimana mestinya.

28-12-2025
Petugas

Gambar 9. Output Cetak Surat Peminjaman

Gambar 9 memperlihatkan hasil keluaran berupa dokumen cetak surat peminjaman barang. Dokumen ini berisi informasi lengkap terkait transaksi peminjaman, termasuk identitas peminjam, data barang, serta periode peminjaman. Fitur ini berfungsi sebagai bukti administratif sekaligus arsip resmi yang mendukung transparansi dan akuntabilitas dalam pengelolaan peminjaman barang.

4. Kesimpulan

Berdasarkan hasil perancangan, implementasi, dan pengujian sistem, dapat disimpulkan bahwa pengembangan sistem informasi peminjaman alat rumah tangga berbasis web di lingkungan RT/RW berhasil memberikan solusi terhadap permasalahan pengelolaan aset yang sebelumnya dilakukan secara manual. Sistem ini mampu meningkatkan efisiensi pengelolaan data, meminimalkan kesalahan pencatatan, serta mendukung transparansi dan akuntabilitas dalam proses peminjaman barang. Penerapan metode V-Model terbukti efektif dalam memastikan setiap tahapan pengembangan sistem berjalan terstruktur, teruji, dan sesuai dengan kebutuhan pengguna. Hasil pengujian menggunakan metode black-box menunjukkan bahwa seluruh fitur sistem berfungsi dengan baik dan sesuai dengan skenario penggunaan yang dirancang. Sistem juga mampu mengelola data aset, peminjaman, serta laporan secara terintegrasi dan *real-time*. Pemanfaatan teknologi web berbasis PHP, MySQL, dan *framework CodeIgniter* memberikan kemudahan dalam pengembangan, pemeliharaan, serta penggunaan sistem oleh masyarakat dengan tingkat literasi digital yang beragam. Dengan demikian, sistem ini tidak hanya berfungsi sebagai alat administrasi, tetapi juga sebagai sarana pendukung tata kelola komunitas yang lebih modern, efisien, dan berkelanjutan. Pengembangan sistem ke depan disarankan untuk menambahkan fitur notifikasi otomatis guna meningkatkan jadwal pengembalian barang, sehingga dapat meningkatkan kedisiplinan pengguna. Selain itu, aspek keamanan sistem perlu ditingkatkan melalui penerapan enkripsi data, pengelolaan hak akses yang lebih rinci, serta pencatatan aktivitas pengguna untuk meminimalkan potensi penyalahgunaan. Pengembangan integrasi dengan sistem lain di tingkat RT/RW juga dapat dipertimbangkan agar pengelolaan data menjadi lebih terpusat dan efisien. Di samping itu, penyempurnaan antarmuka pengguna dan penambahan fitur pelaporan berbasis visual dianjurkan untuk meningkatkan kenyamanan penggunaan serta mendukung pengambilan keputusan yang lebih akurat dan berbasis data.

Referensi

- [1] Minardi, J., & Abdullah, S. (2023). *Digital transformation in community governance: Best practices and challenges*. Penerbit Teknokrat. <https://journal.ipm2kpe.or.id/index.php/INTECOM/article/view/17776>
- [2] Aulia, L. (2025). *Perancangan sistem peminjaman ruangan berbasis web*. JUTISI, 11(1), 45–56. <https://ojs.stmik-banjarbaru.ac.id/index.php/jutisi/article/view/2594>
- [3] Nugroho, W. B. (2025). *Perancangan sistem informasi peminjaman ruang laboratorium komputer berbasis web*. JUKOMTEK, 13(2), 67–78. <https://jurnal-cahayapatriot.org/index.php/jukomtek/article/view/415>
- [4] Prehanto, B. (2020). *Buku ajar konsep sistem informasi*. Telkom University Press. <https://books.google.co.id/books?id=0OriDwAAQBAJ>
- [5] Laudon, K. C., & Laudon, J. P. (2016). *Management information systems* (14th ed.). Pearson Education. https://books.google.co.id/books/about/Management_Information_Systems.html?id=SZSpxAEACAAJ
- [6] Sommerville, I. (2011). *Software engineering* (9th ed.). Addison-Wesley. <https://engineering.futureuniversity.com/BOOKS%20FOR%20IT/Software-Engineering-9th-Edition-by-Ian-Sommerville.pdf>
- [7] Pressman, R. S., & Maxim, B. R. (2015). *Software engineering: A practitioner's approach* (8th ed.). McGraw-Hill Education. https://books.google.co.id/books/about/Software_Engineering.html?id=bL7QZHtWvaUC
- [8] Ward, J., & Peppard, J. (2002). *Strategic planning for information systems* (3rd ed.). Wiley. https://books.google.co.id/books/about/Strategic_Planning_for_Information_Syste.html?id=Y-djKt6DaV8C
- [9] Nixon, R. (2012). *Learning PHP, MySQL, JavaScript, and CSS* (3rd ed.). O'Reilly Media. http://projanco.com/Library/Leaming%20PHP,%20MySQL,%20JavaScript,%20CSS%20_%20HTML5,%20Third%20Edition-%20A%20Step-by-Step%20Guide%20to%20Creating%20Dynamic%20Websites.pdf
- [10] Spurlock, J. (2015). *Bootstrap: Responsive web development*. O'Reilly Media. <https://books.google.co.id/books?id=LZm7Cxgi3aQC>
- [11] Nabilah, M. Z., Ramadhani, A. A., Nazzarin, Z. I., Anastasya, A., Butsainah, N., & Sebayang, F. A. (2025). Rancangan sistem peminjaman barang digital berbasis web. *JATI (Jurnal Mahasiswa Teknik Informatika)*, 9(1), 1737–1743. <https://www.ejournal.itn.ac.id/jati/article/download/12347/7108>
- [12] Saputra, S. M. T. (2025). Pembuatan aplikasi peminjaman barang inventaris berbasis website dengan teknologi RFID. *REMik*, 9(1), 78–90. <https://repository.wicida.ac.id/6222/>
- [13] Maysaroh, A. (2025). Implementasi aplikasi peminjaman dan pengembalian buku perpustakaan berbasis web. *JSSR*, 8(1), 78–90. <https://www.jurnal.goretanpena.com/index.php/JSSR/article/view/4212/2085>
- [14] Rachmawati, R. (2024). Implementasi sistem peminjaman ruang berbasis web untuk efisiensi manajemen dengan evaluasi PIECES. *Jurnal Infokom*, 18(2), 89–102. https://www.researchgate.net/publication/391586557_Implementasi_Sistem_Peminjaman_Ruang_Berbasis_Web_Untuk_Efisiensi_Manajemen_Dengan_Evaluasi_Pieces
- [15] Abairahman, M. B. (2025). Perancangan dan implementasi sistem peminjaman barang berbasis web pada PT. Gresik Migas. *REMik*, 9(2), 112–125. <https://jurnal.polgan.ac.id/index.php/remik/article/view/14365>
- [16] Syahid Pebriadi, M., Salman, P., Kemal Fattah, T., & Negeri Banjarmasin Jl Brigjen Hasan Basri, P. H. (2023). Implementasi Use Case Diagram Dan Activity Diagram Dalam Perancangan Aplikasi Kalkulator Pajak Bagi UMKM. *Multitek Indonesia: Jurnal Ilmiah*, 1, 1907–6223. <http://journal.umpo.ac.id/index.php/multitek>
- [17] Ramadhani Mukhlis, I., Hermansyah, D., & Meilisa Lantang, V. (2023). Rancangan Basis Data Transaksi Pada PT.Bank Perkreditan Rakyat ABC Menggunakan MySQL Dengan Model Entity Relationship Diagram (ERD) dan Physical Data Model (PDM). *Journal of Advances in Information and Industrial Technology*, 5(1), 1–10. <https://doi.org/10.52435/jaiit.v5i1.305>

- [18] Hasibuan, M. A., Harahap, S. Z., Nasution, M., Informasi, S., Sains, F., Teknologi, D., & Labuhanbatu, U. (2024). Analisis Sistem Informasi Pengelolaan Data Alumni MAN Labuhanbatu Berbasis Codeigniter PHP Framework. *Muhammad Adlin Hasibuan*, 1(3). <https://doi.org/10.36987/informatika.v12i3.6157>
- [19] Suryadi, O., Ahmad, L., Studi Manajemen Informatika, P., Indonesia Banda Aceh, S., Banda Aceh, K., & Aceh, P. (2024). Jurnal Sistem Komputer (SISKOM) Perancangan Sistem Informasi Penerimaan Peserta Didik Baru Dan Ujian Online Dengan Php Dan Codeigniter Di Smk Negeri 1 Sigli. *Jurnal Sistem Komputer (SISKOM)*, 4(2). <https://doi.org/10.35870/siskom.v4i2.816>