



## PENGEMBANGAN SISTEM INFORMASI TERPADU UNTUK PENGELOLAAN ASET, IT DAN LABORATORIUM DI SEKOLAH KEJURUAN

Refi Riduan Achmad<sup>1</sup>, Dwi Fiqkri Hermawanto<sup>2</sup>, Rafi Agung<sup>3</sup>

<sup>1</sup>refiriduanachmad@unukaltim.ac.id, <sup>2</sup>dwifiqkri@gmail.com, <sup>3</sup>rafilapalulu@gmail.com

<sup>1,2,3</sup>Universitas Nahdlatul Ulama Kalimantan Timur

### Abstrak

Pengelolaan sarana prasarana, teknologi informasi, dan laboratorium di sekolah kejuruan memerlukan sistem yang mampu mengelola data secara terintegrasi dan terstruktur. Pada praktiknya, pengelolaan aset di sekolah masih banyak dilakukan menggunakan beberapa file dan aplikasi terpisah, sehingga berpotensi menimbulkan duplikasi data, kesalahan pencatatan, dan kesulitan dalam monitoring. Penelitian ini bertujuan untuk mengembangkan sistem informasi terpadu berbasis web yang mendukung pengelolaan aset, teknologi informasi, dan laboratorium sebagai studi kasus di SMK Tamansiswa Kota Mojokerto. Metode pengembangan sistem yang digunakan adalah *Rational Unified Process* (RUP) karena bersifat iteratif dan sistematis. Sistem dikembangkan menggunakan *framework* CodeIgniter 4 dengan arsitektur *Model-View-Controller* (MVC) serta basis data MySQL. Pengujian sistem dilakukan melalui *Black Box Testing* untuk memastikan fungsi sistem berjalan sesuai kebutuhan, serta *User Acceptance Testing* (UAT) untuk mengetahui tingkat penerimaan pengguna. Hasil pengujian menunjukkan bahwa seluruh fungsi utama sistem dapat berjalan dengan baik dan UAT memperoleh tingkat kepuasan pengguna sebesar 92% dengan kategori Sangat Baik. Dengan demikian, sistem yang dikembangkan mampu mendukung pengelolaan aset sekolah secara terintegrasi dan diterima dengan baik oleh pengguna.

**Kata kunci:** Sistem Informasi Terpadu, Manajemen Aset, Sekolah Kejuruan, RUP, CodeIgniter.

### Abstract

*The management of facilities, information technology, and laboratories in vocational schools requires an information system that can handle data in an integrated and structured manner. In practice, asset management in schools is often conducted using separate files and applications, which may lead to data duplication, recording errors, and difficulties in monitoring. This study aims to develop a web-based integrated information system to support the management of assets, information technology, and laboratories, using SMK Tamansiswa Kota Mojokerto as a case study. The system was developed using the Rational Unified Process (RUP) methodology due to its iterative and systematic nature. The implementation utilized the CodeIgniter 4 framework with a Model-View-Controller (MVC) architecture and a MySQL database. System testing was conducted through Black Box Testing to verify functional requirements and User Acceptance Testing (UAT) to measure user acceptance. The results indicate that all main system functions operate properly, and the UAT achieved a user satisfaction score of 92%, categorized as very good. Therefore, the developed system supports integrated asset management in schools and is well accepted by users.*

**Keywords:** Integrated Information System, Asset Management, Vocational School, RUP, CodeIgniter.

## 1. Pendahuluan

Sistem informasi saat ini menjadi komponen penting dalam mendukung proses administrasi dan pengolahan data di berbagai lembaga, termasuk institusi pendidikan. Pengelolaan sarana prasarana, teknologi informasi, dan laboratorium di SMK Tamansiswa Kota Mojokerto masih dilakukan menggunakan beberapa file dan aplikasi terpisah, seperti lembar kerja *Spreadsheet* (Excel) dan pencatatan digital yang tidak saling terintegrasi. Kondisi tersebut menyebabkan proses pencatatan aset, peminjaman, dan pelaporan berjalan secara terfragmentasi, sehingga berpotensi menimbulkan duplikasi data, kesalahan pencatatan, serta kendala dalam proses monitoring dan penyusunan laporan secara menyeluruh.

Kondisi pengelolaan data yang terpisah dan tidak terintegrasi tersebut menunjukkan perlunya penerapan sistem informasi yang mampu mengelola data aset secara terstruktur dan terpusat. Dengan

adanya sistem informasi yang terintegrasi, proses pencatatan, peminjaman, dan pelaporan aset diharapkan dapat dilakukan dengan lebih sistematis, mudah dipantau, serta mendukung kebutuhan administrasi sekolah secara lebih efektif.

Penerapan sistem informasi berbasis teknologi telah banyak dikembangkan untuk meningkatkan efisiensi proses administrasi. Salah satu contohnya adalah sistem pelacakan dokumen berbasis teknologi mobile dan QR Code yang mampu mempercepat proses pencarian dan pengawasan dokumen [1]. Selain itu, pengelolaan sarana dan prasarana yang terstruktur juga menjadi faktor penting dalam mendukung layanan dan aktivitas operasional di suatu institusi Pendidikan [2]. Tanpa sistem yang terintegrasi, pengelolaan aset berpotensi menimbulkan ketidakteraturan data dan kesulitan dalam proses monitoring.

Keberhasilan penerapan sistem informasi tidak hanya ditentukan oleh fungsionalitas sistem, tetapi juga oleh kemudahan penggunaan. Penerapan prinsip Interaksi Manusia dan Komputer pada antarmuka sistem informasi terbukti dapat meningkatkan efektivitas sistem serta kenyamanan [3]. Beberapa penelitian terdahulu menunjukkan bahwa pengembangan sistem informasi di lingkungan pendidikan mampu meningkatkan kualitas layanan administrasi dan akurasi pengelolaan data [1], [2]. Selain itu, sistem monitoring berbasis web menegaskan pentingnya pengawasan data secara *real-time* untuk mendukung proses evaluasi dan pengambilan keputusan [4], serta perancangan antarmuka UI/UX yang baik agar sistem mudah digunakan oleh pengguna akhir [5].

Secara teoritis, konsep Interaksi Manusia dan Komputer berperan dalam merancang sistem yang mudah dipahami oleh pengguna [6], sedangkan prinsip rekayasa perangkat lunak menjadi dasar dalam pengembangan sistem yang berkualitas dan mudah dipelihara [7]. Dalam konteks pengembangan sistem informasi, sistem pelacakan dokumen berbasis teknologi mobile dan QR Code telah dikembangkan untuk mendukung keteraturan pengelolaan data [1]. Selain itu, pengelolaan sarana dan prasarana yang terstruktur menjadi faktor penting dalam mendukung aktivitas institusi pendidikan [2]. Pengembangan sistem monitoring berbasis web juga menunjukkan pentingnya pengawasan data secara terpusat [4]. Namun, penelitian-penelitian tersebut umumnya masih berfokus pada fungsi tertentu dan belum mengintegrasikan pengelolaan aset, teknologi informasi, dan laboratorium dalam satu sistem terpadu. Oleh karena itu, penelitian ini mengembangkan sistem informasi terpadu berbasis web yang mengintegrasikan pengelolaan aset, IT, dan laboratorium dalam satu platform sebagai studi kasus di SMK Tamansiswa Kota Mojokerto dengan menggunakan metode *Rational Unified Process* (RUP). Berdasarkan hal tersebut, penelitian ini mengembangkan sistem informasi terpadu berbasis web untuk pengelolaan aset, IT, dan laboratorium di Sekolah Menengah Kejuruan guna mendukung pengelolaan fasilitas secara terintegrasi.

## 2. Metode

Penelitian ini menggunakan metode pengembangan perangkat lunak *Rational Unified Process* (RUP). Metode ini dipilih karena bersifat iteratif dan sistematis, sehingga memungkinkan proses pengembangan sistem dilakukan secara bertahap sesuai dengan kebutuhan pengguna. Selain itu, RUP sesuai untuk pengembangan sistem informasi terintegrasi yang memiliki beberapa modul utama, karena setiap tahapannya mendukung proses analisis, perancangan, implementasi, dan pengujian secara terstruktur.

Penelitian ini mengembangkan sebuah sistem informasi manajemen aset berbasis web yang dirancang untuk mendukung proses pencatatan, peminjaman, monitoring, hingga pengaduan aset. Sistem dikembangkan menggunakan metode *Model-View-Controller* (MVC) pada *framework* CodeIgniter 4 sehingga struktur sistem menjadi lebih terorganisir, modular, dan mudah dipelihara. Untuk memastikan validitas, keamanan, dan kualitas sistem, dilakukan serangkaian aktivitas mulai dari pemodelan kebutuhan pengguna, perancangan antarmuka dan alur data, implementasi fungsi utama, hingga pengujian sistem menggunakan *Black Box Testing* dan *User Acceptance Testing* (UAT). Aktivitas ini dirancang agar sistem tidak hanya memenuhi kebutuhan fungsional, tetapi juga aspek efisiensi dan kemudahan penggunaan bagi pengguna akhir [5], [8].

Pendekatan pengembangan sistem menggunakan *Rational Unified Process* (RUP) yang bersifat iteratif dan menekankan pengembangan sistem secara bertahap sesuai kebutuhan pengguna. Tahap *Inception* dilakukan untuk mengidentifikasi kebutuhan dan permasalahan pengguna melalui wawancara, observasi langsung pada divisi sarana prasarana, IT, dan laboratorium, serta studi dokumentasi terkait pencatatan aset, peminjaman, dan pengaduan. Analisis *stakeholder* dilakukan untuk menentukan prioritas kebutuhan dan ekspektasi pengguna sehingga sistem dapat menjawab permasalahan operasional secara tepat [9], [10]. Hasil dari tahap *Inception* menjadi dasar dalam menentukan ruang lingkup sistem serta arah

pengembangan sistem informasi yang akan dibangun. Informasi kebutuhan pengguna yang diperoleh pada tahap ini kemudian digunakan sebagai acuan pada tahap-tahap selanjutnya dalam metodologi RUP, sehingga pengembangan sistem dapat dilakukan secara terarah dan sesuai dengan permasalahan yang dihadapi oleh pengguna. Dengan pendekatan ini, setiap tahapan pengembangan saling berkaitan dan membentuk alur kerja yang sistematis, mulai dari identifikasi kebutuhan hingga kesiapan sistem untuk diimplementasikan. Secara keseluruhan, tahapan pengembangan sistem berdasarkan metodologi *Rational Unified Process* (RUP), pada penelitian ini disusun dalam suatu alur pengembangan yang terstruktur sebagaimana ditunjukkan pada Gambar 1.



Gambar 1. Alur Pengembangan Sistem Berdasarkan Metodologi *Rational Unified Process* (RUP)

Tahap *Elaboration* berfokus pada analisis kebutuhan secara rinci dan perancangan sistem. Aktivitas meliputi pembuatan *use case diagram*, *activity diagram*, *Physical Data Model* (PDM), dan *wireframe* antarmuka. *Mockup* interaktif diuji oleh pengguna untuk mengidentifikasi permasalahan *usability* sebelum implementasi. Validasi alur data dilakukan agar proses informasi antar modul berjalan lancar tanpa duplikasi pencatatan, sementara analisis risiko sistem digunakan untuk mengantisipasi potensi kesalahan input, keamanan data, dan *downtime server* [3], [10].

Tahap *Construction* merupakan implementasi sistem berdasarkan rancangan yang telah disetujui dengan menggunakan *CodeIgniter 4*, *PHP*, dan *MySQL*. Sistem dikembangkan untuk mendukung modul pencatatan aset, peminjaman, pengaduan, dan monitoring. Validasi input serta mekanisme keamanan diterapkan untuk mencegah kesalahan data dan akses tidak sah, sementara integrasi modul dilakukan agar sistem berjalan secara terpadu dan mampu menghasilkan laporan otomatis guna mendukung pengambilan keputusan manajemen terkait sarana prasarana dan laboratorium [9], [10].

Tahap *Transition* memastikan sistem siap digunakan oleh pengguna melalui *User Acceptance Testing* (UAT) untuk mengevaluasi fungsi sistem dan tingkat kepuasan pengguna. Pelatihan pengguna dilakukan agar sistem dapat dimanfaatkan secara optimal, disertai monitoring dan evaluasi pasca-implementasi untuk mengidentifikasi kendala dan melakukan perbaikan sistem secara berkelanjutan. Prosedur *backup*, *maintenance*, dan pembaruan sistem dijalankan agar sistem tetap handal, aman, dan adaptif terhadap perubahan kebutuhan di masa depan [5], [6].

Selain itu, penelitian ini menekankan iterasi dan evaluasi berkala pada setiap *fase* RUP. Review internal dan sesi umpan balik pengguna dilakukan di setiap tahap untuk mendeteksi permasalahan sejak awal sehingga kesalahan implementasi dapat diminimalkan. Pengumpulan data tambahan melalui kuesioner dan *Focus Group Discussion* (FGD) memastikan seluruh kebutuhan operasional tercakup, sementara dokumentasi sistem dan panduan penggunaan mendukung pelatihan pengguna dan monitoring pasca-implementasi [5], [8].

### 3. Hasil dan Pembahasan

#### 3.1. Pemodelan Sistem

Pada penelitian ini dikembangkan sebuah sistem informasi manajemen aset berbasis web yang dirancang untuk mendukung seluruh proses pencatatan, peminjaman, monitoring, hingga pengaduan aset. Sistem dikembangkan menggunakan metode *Model-View-Controller* (MVC) pada framework CodeIgniter 4, sehingga struktur aplikasi menjadi lebih terorganisir, modular, dan mudah dipelihara. Untuk memastikan validitas, keamanan, dan kualitas sistem, dilakukan serangkaian aktivitas mulai dari pemodelan kebutuhan pengguna, perancangan antarmuka dan alur data, hingga implementasi fungsi utama dan integrasi modul. Sistem juga diuji menggunakan *Black Box Testing* dan *User Acceptance Testing* (UAT) untuk mengevaluasi kinerja setiap fitur, mendeteksi potensi kesalahan, serta mengukur kepuasan pengguna terhadap fungsionalitas dan kemudahan penggunaan.

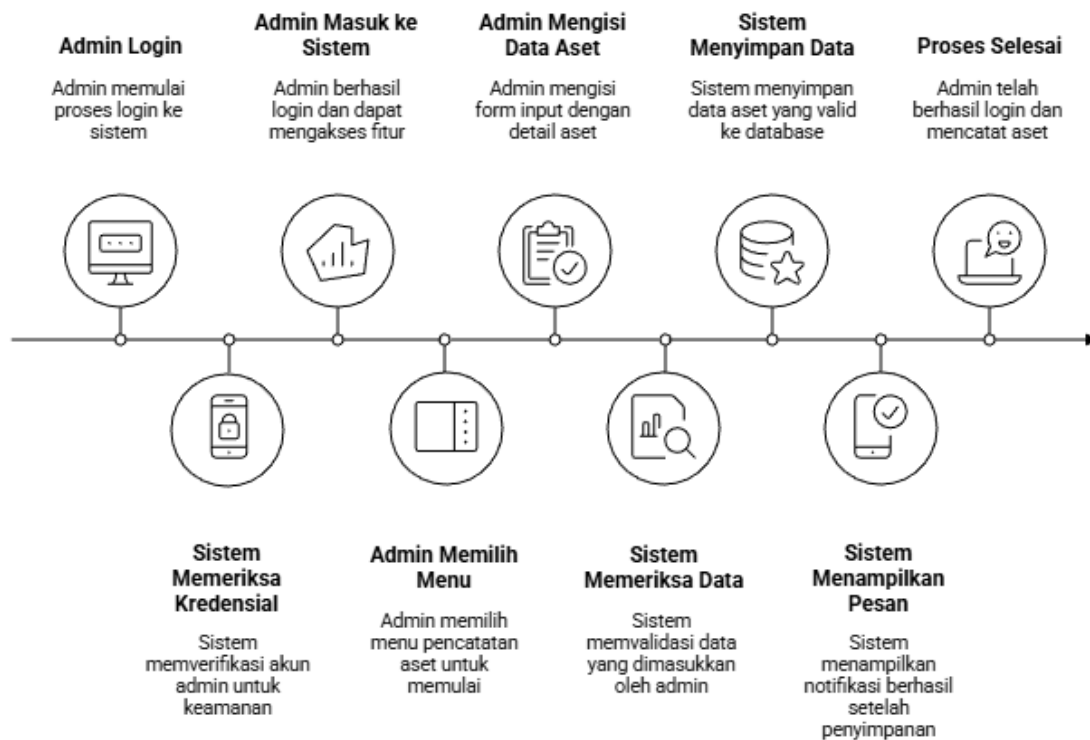
Selain itu, dilakukan iterasi dan evaluasi berkala di setiap tahap pengembangan untuk memperbaiki rancangan sistem dan menyesuaikan kebutuhan pengguna yang muncul selama pengujian. Aktivitas ini mencakup validasi alur data, pengujian modul secara internal, dan pengumpulan *feedback* dari pengguna, sehingga setiap modul yang diterapkan mampu berjalan secara optimal, aman, dan sesuai dengan tujuan pengelolaan aset. Dengan pendekatan ini, penelitian tidak hanya menekankan pengembangan sistem, tetapi juga memastikan sistem yang dihasilkan mampu meningkatkan efisiensi operasional dan meminimalkan kesalahan pencatatan.



Gambar 2. Use Case Diagram

*Use Case Diagram* pada Gambar 2 menggambarkan interaksi antara aktor dan Sistem Informasi Aset. Diagram ini melibatkan tiga aktor utama, yaitu Admin, Guru, dan Siswa. Admin memiliki hak akses untuk mengelola data aset dan memproses peminjaman aset, sedangkan Guru dan Siswa dapat melihat informasi aset serta mengajukan peminjaman aset melalui sistem. Diagram ini juga menunjukkan batasan sistem (*system boundary*) yang memisahkan fungsi sistem dari aktor eksternal, sehingga memberikan gambaran fungsi utama sistem sesuai dengan peran masing-masing aktor.

Setelah tahap pemodelan kebutuhan, dibuat *Activity Diagram* untuk menggambarkan alur operasional sistem, khususnya pada proses yang dianggap paling kritis, yaitu pencatatan aset dan peminjaman aset. *Activity Diagram* membantu memvisualisasikan bagaimana data mengalir dari awal proses hingga selesai.



Gambar 3. Activity Diagram Pencatatan Aset

Berdasarkan Gambar 3, *activity diagram* pencatatan aset menunjukkan alur proses yang dimulai dari admin melakukan login ke dalam sistem. Sistem memeriksa kredensial admin untuk memastikan keamanan akses sebelum admin dapat mengakses fitur pencatatan aset melalui menu yang tersedia.

Setelah memilih menu pencatatan, admin mengisi data aset melalui formulir input. Sistem kemudian melakukan validasi terhadap data yang dimasukkan untuk memastikan kelengkapan dan kesesuaian format sebelum data disimpan ke dalam basis data. Apabila data tidak valid, sistem akan menampilkan pesan kesalahan untuk dilakukan perbaikan.

Setelah data berhasil disimpan, sistem menampilkan notifikasi sebagai tanda bahwa proses pencatatan aset telah selesai. Alur ini menunjukkan bahwa pencatatan aset dilakukan secara terstruktur dan terdokumentasi, sehingga dapat meminimalkan kesalahan input serta duplikasi data.

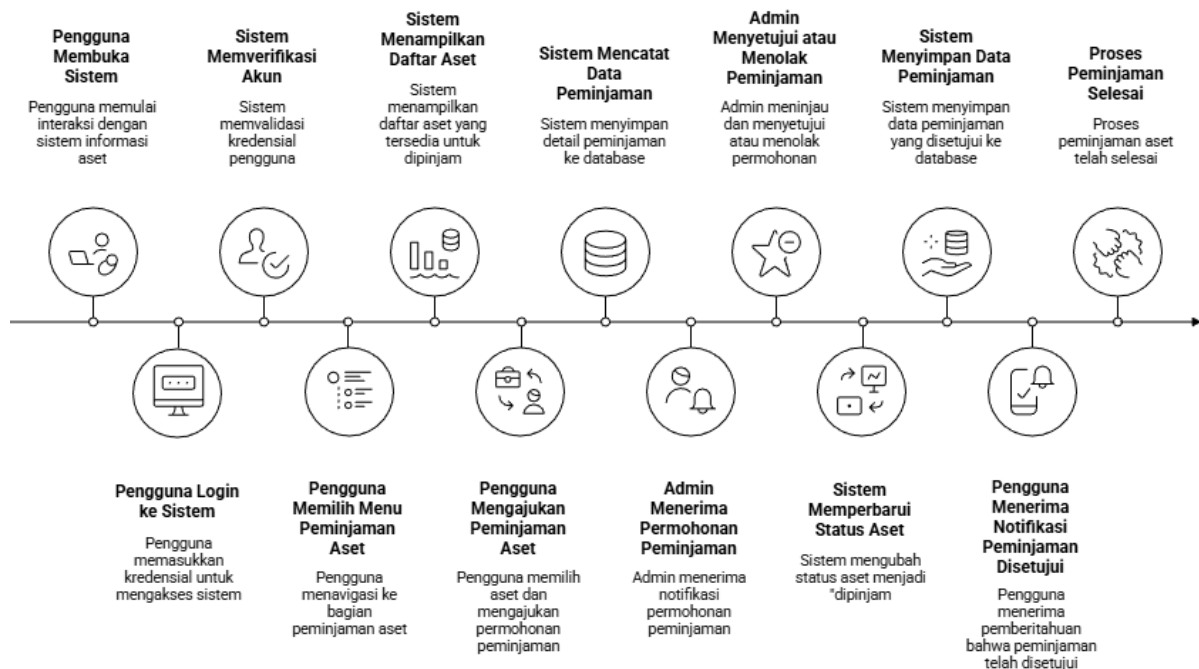
Setelah pengguna mengirimkan permintaan peminjaman melalui sistem, alur pada diagram aktivitas menunjukkan bahwa sistem terlebih dahulu melakukan validasi kelengkapan data permintaan. Validasi ini mencakup pengecekan identitas peminjam, aset yang dipilih, serta ketersediaan aset pada tanggal yang diajukan. Apabila data tidak lengkap, sistem akan mengembalikan pesan kesalahan dan meminta pengguna memperbaiki input.

Jika semua data dinyatakan valid, permintaan diteruskan ke admin untuk proses persetujuan. Admin memiliki dua opsi, yaitu menerima atau menolak permintaan. Pada saat admin menyetujui, sistem secara otomatis mengubah status aset menjadi dipinjam dan mencatat transaksi pada tabel peminjaman, sehingga mencegah terjadinya peminjaman ganda. Apabila admin menolak permintaan, sistem mengirimkan notifikasi kepada pengguna dengan mencantumkan alasan penolakan.

Setelah aset dipinjam, aktivitas berlanjut pada fase pengembalian. Pengguna mengembalikan aset secara fisik kepada admin, dan admin kemudian memperbarui status aset menjadi tersedia atau perlu perbaikan apabila ditemukan kerusakan. Sistem mencatat seluruh proses tersebut untuk kebutuhan audit dan pelacakan riwayat penggunaan aset. Alur yang terstruktur dalam gambar ini memastikan seluruh proses peminjaman berlangsung konsisten, terdokumentasi, dan bebas dari duplikasi pencatatan.

Untuk mendukung proses pencatatan aset, peminjaman, pengaduan, dan monitoring, diperlukan struktur basis data yang dirancang secara sistematis dan konsisten. Oleh karena itu, dibuat *Physical Data Model* (PDM) sebagai representasi rinci hubungan antar entitas di dalam sistem. PDM berfungsi untuk

memastikan setiap data yang disimpan memiliki integritas tinggi melalui penerapan *primary key*, *foreign key*, serta aturan relasional yang sesuai dengan proses bisnis. Dengan pemodelan ini, desain database dapat diimplementasikan secara optimal sehingga mampu mendukung kebutuhan operasional maupun kebutuhan analisis data di tahap selanjutnya.



Gambar 4. Activity Diagram Peminjaman Aset

Berdasarkan struktur pada PDM tersebut, terlihat bahwa tabel-tabel inti seperti aset, peminjaman, dan pengaduan memiliki hubungan relasional yang merepresentasikan alur kerja sistem secara nyata. Tabel aset menjadi entitas utama yang terhubung dengan berbagai transaksi peminjaman dan laporan pengaduan, memungkinkan pelacakan riwayat secara menyeluruh terhadap setiap aset. Relasi *one-to-many* yang diterapkan memastikan data tetap konsisten sekaligus mencegah terjadinya duplikasi. Desain yang terstruktur ini mempermudah proses *query* untuk kebutuhan monitoring, audit, dan evaluasi, sehingga mendukung peningkatan akurasi dan efisiensi dalam pengelolaan fasilitas sekolah.

Secara keseluruhan, rangkaian diagram pada Gambar 2 hingga Gambar 4 digunakan sebagai dasar konseptual dan teknis dalam perancangan sistem. Ketiga diagram tersebut saling melengkapi untuk menggambarkan struktur, alur proses, serta hubungan antar komponen, sehingga pada tahap implementasi sistem dapat dikembangkan secara lebih terarah, konsisten, dan sesuai kebutuhan.

### 3.2. Implementasi Sistem

Implementasi sistem dilakukan dengan memfokuskan pengembangan pada fungsi utama pengelolaan aset, yaitu *CreateRincianAset*. Fungsi ini bertanggung jawab untuk menambahkan data aset baru ke dalam sistem secara terstruktur dan akurat. Pada tahap ini, dilakukan sejumlah proses penting mencakup validasi kelengkapan dan format input, pengecekan duplikasi data berdasarkan kode aset, serta penyimpanan data ke dalam database sesuai skema yang telah dirancang. Proses implementasi tersebut disusun dalam bentuk *pseudocode* untuk memastikan konsistensi logika dan memudahkan pengujian awal. Alur logis dan langkah-langkah proseduralnya ditampilkan pada Tabel 1.

Tabel 1. *Pseudocode Method CreateRincianAset*

No	Langkah	Keterangan
1	Mulai	Inisialisasi method CreateRincianAset
2	Validasi Input	Memastikan seluruh data aset terisi lengkap dan sesuai
3	Cek Duplikasi	Memeriksa apakah kode atau nama aset sudah ada di database
4	Simpan Data	Menambahkan data aset ke tabel rincian_aset
5	Selesai	Mengembalikan status berhasil atau gagal

*Pseudocode* pada Tabel 1 menggambarkan alur dasar eksekusi *method CreateRincianAset* yang digunakan untuk menambahkan data aset baru ke dalam sistem. Proses dimulai dengan inisialisasi *method*, kemudian dilakukan validasi terhadap seluruh input untuk memastikan data memenuhi kriteria format dan kelengkapan. Setelah itu, sistem melakukan pengecekan duplikasi guna mencegah terjadinya konflik data dengan aset yang telah tersimpan sebelumnya. Jika tidak ditemukan duplikasi, sistem melanjutkan proses penyimpanan data ke dalam tabel *rincian\_aset*. Tahap akhir dari proses ini adalah pengembalian status eksekusi berupa informasi apakah operasi penambahan aset berhasil dilakukan atau mengalami kegagalan. Struktur langkah yang terurut ini memastikan bahwa proses perekaman aset berlangsung secara konsisten, aman, dan bebas dari redundansi data.

Implementasi teknis dari *pseudocode* tersebut direalisasikan menggunakan *framework CodeIgniter 4*. Pada tahap ini, logika proses *CreateRincianAset* diubah ke dalam bentuk kode program yang dijalankan oleh *controller*. Implementasi meliputi proses validasi input, pemeriksaan duplikasi data, serta eksekusi penyimpanan aset baru ke dalam *database*. Potongan kode inti yang menggambarkan langkah-langkah tersebut diringkas pada Tabel 2.

Tabel 2. Implementasi *Method CreateRincianAset* (PHP / CodeIgniter 4)

No	Kode / Fungsi	Penjelasan
1	<code>\$this-&gt;validate([...])</code>	Validasi input data aset
2	<code>\$this-&gt;RincianAsetModel-&gt;cekDuplikasi(...)</code>	Mengecek apakah aset sudah tersimpan
3	<code>\$this-&gt;RincianAsetModel-&gt;insert(...)</code>	Menyimpan data aset ke database
4	<code>return redirect()-&gt;to(...)-&gt;with('success', ...)</code>	Menampilkan notifikasi keberhasilan

Berdasarkan Tabel 2, implementasi *method CreateRincianAset* dilakukan dengan mengikuti alur logis yang telah ditetapkan pada *pseudocode* sebelumnya. Proses dimulai dengan validasi data menggunakan fungsi *validate()* untuk memastikan bahwa seluruh input yang diterima memenuhi aturan format dan kelengkapan. Setelah itu, dilakukan pemanggilan fungsi *cekDuplikasi()* pada model guna memastikan tidak terdapat aset dengan kode atau nama yang sama di dalam database. Apabila data dinyatakan unik, sistem akan mengeksekusi fungsi *insert()* untuk menambahkan data aset ke tabel *rincian\_aset*. Tahap akhir berupa pengiriman notifikasi sukses melalui mekanisme *redirect*, yang berfungsi memberikan umpan balik langsung kepada pengguna. Implementasi ini menunjukkan bahwa sistem telah dirancang dengan memperhatikan aspek keamanan data, konsistensi, serta kemudahan penggunaan.

Implementasi teknis dari proses penambahan data aset dilakukan menggunakan *framework CodeIgniter 4*. Pada tahap ini, fungsi utama yang diimplementasikan adalah proses *CreateRincianAset*, yang bertanggung jawab untuk memvalidasi input, memastikan tidak terjadi duplikasi data, serta menyimpan data aset baru ke dalam basis data. Potongan kode berikut menunjukkan bagian inti dari fungsi tersebut, yang disederhanakan agar sesuai dengan kebutuhan penulisan ilmiah dan tidak menampilkan keseluruhan struktur sistem.

Pemisahan antara penyajian *pseudocode* dan implementasi teknis dalam bentuk kode program memberikan kejelasan dalam memahami proses pengembangan sistem secara menyeluruh. *Pseudocode* berfungsi sebagai gambaran konseptual alur logika sistem, sedangkan implementasi kode merepresentasikan penerapan logika tersebut ke dalam lingkungan pemrograman yang sesungguhnya. Pendekatan ini membantu memastikan bahwa setiap tahapan proses yang dirancang pada level konseptual

dapat direalisasikan secara konsisten pada tahap implementasi.

Selain itu, implementasi fungsi *CreateRincianAset* dirancang dengan memperhatikan ketertelusuran proses dan kejelasan alur eksekusi. Setiap tahapan, mulai dari validasi input hingga penyimpanan data ke dalam basis data, dilakukan secara berurutan untuk meminimalkan potensi kesalahan serta menjaga kualitas dan konsistensi data aset. Dengan struktur implementasi yang jelas, sistem diharapkan mampu mendukung pengembangan modul lain secara berkelanjutan serta mempermudah proses pemeliharaan dan pengembangan sistem di masa mendatang.

Dengan demikian, implementasi fungsi *CreateRincianAset* menjadi landasan penting dalam mendukung proses pengelolaan aset yang terintegrasi dan andal.

Listing 1. Implementasi *Method CreateRincianAset*

```
public function create()
{
    $validate = $this->validate([
        'nama_aset' => 'required',
        'kode_aset' => 'required',
        'kategori' => 'required'
    ]);

    if (!$validate) {
        return redirect()->back()->with('error', 'Input tidak
valid');
    }

    if ($this->RincianAsetModel->cekDuplikasi(
        $this->request->getPost('kode_aset')
    )) {
        return redirect()->back()->with('error', 'Aset sudah
terdaftar');
    }

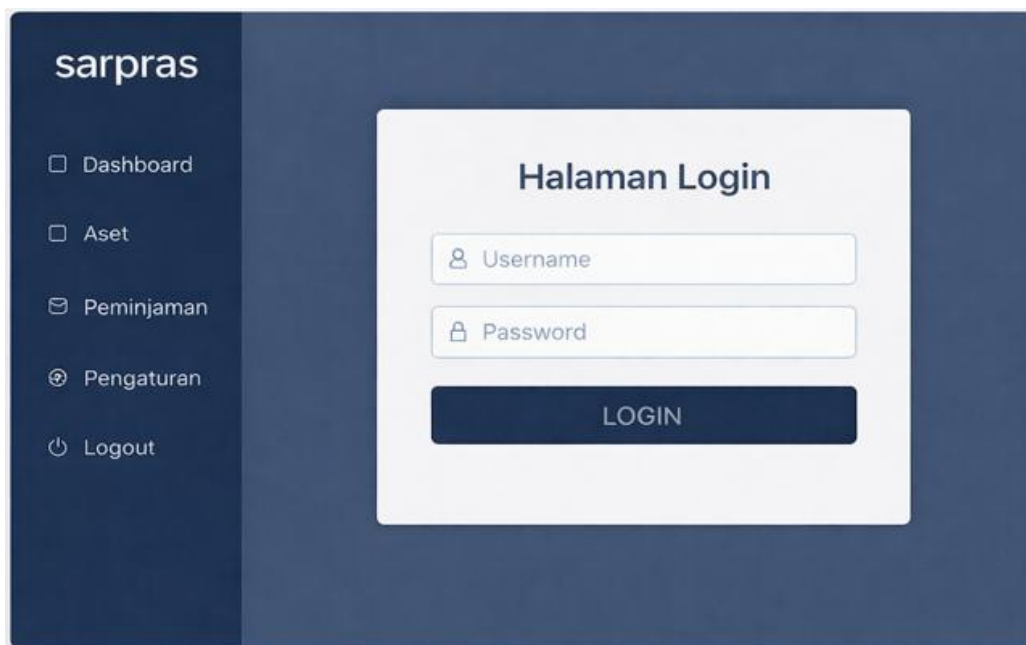
    $this->RincianAsetModel->insert([
        'nama_aset' => $this->request->getPost('nama_aset'),
        'kode_aset' => $this->request->getPost('kode_aset'),
        'kategori' => $this->request->getPost('kategori')
    ]);

    return redirect()->to('/aset')
        ->with('success', 'Data aset berhasil
ditambahkan');
}
```

Potongan kode tersebut menunjukkan alur utama proses *create* dalam modul pencatatan aset. Proses dimulai dengan validasi input menggunakan *\$this->validate()*, yang memastikan bahwa atribut penting seperti nama aset, kode aset, dan kategori telah diisi. Setelah itu, sistem melakukan pengecekan duplikasi melalui method *cekDuplikasi()* untuk mencegah entri aset yang sama dimasukkan lebih dari satu kali. Apabila data dinyatakan valid dan tidak ditemukan duplikasi, maka data akan disimpan ke dalam basis data menggunakan fungsi *insert()*. Sistem kemudian mengarahkan pengguna kembali ke halaman daftar aset beserta pesan notifikasi keberhasilan. Implementasi ini menggambarkan prinsip dasar sistem: validasi ketat, pencegahan redundansi, dan akurasi data.

Setelah proses *backend* selesai diimplementasikan, tahap berikutnya adalah merancang dan membangun tampilan antarmuka (*User Interface/UI*) agar pengguna dapat melakukan pencatatan aset, monitoring, serta pengaduan dengan lebih mudah dan intuitif. Perancangan UI dilakukan dengan mengacu pada prinsip kesederhanaan, konsistensi, dan kemudahan navigasi, sehingga setiap pengguna, baik admin maupun pegawai, dapat mengakses fitur secara efisien tanpa memerlukan pelatihan tambahan.

Sebagai tahap awal dalam proses desain antarmuka, dibuatlah *wireframe* yang berfungsi sebagai representasi visual awal dari tata letak sistem. *Wireframe* digunakan untuk menentukan hierarki elemen, alur interaksi pengguna, serta penempatan komponen utama sebelum dikembangkan menjadi tampilan final. Pendekatan ini umum digunakan dalam pengembangan sistem modern karena dapat meminimalkan kesalahan desain di tahap implementasi.

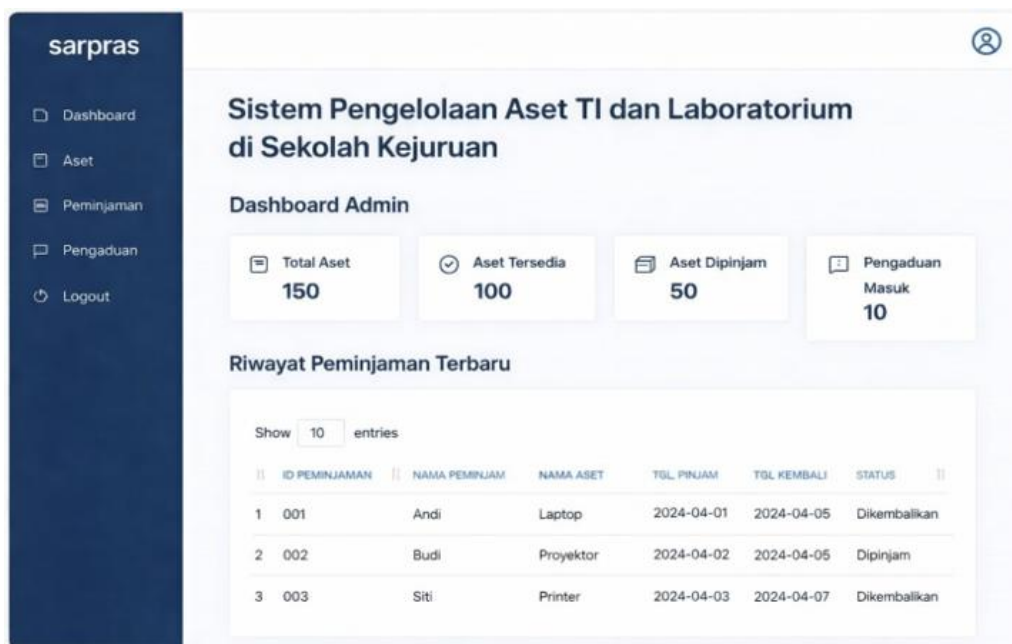


Gambar 5. *Wireframe* Halaman Login

*Wireframe* pada Gambar 5 menampilkan rancangan Halaman Login yang digunakan sebagai antarmuka awal dalam mengakses sistem informasi aset. Halaman ini dirancang untuk mendukung proses autentikasi pengguna dengan menampilkan elemen-elemen utama secara sederhana dan jelas, seperti input *username*, *password*, serta tombol masuk. Perancangan halaman *login* difokuskan pada kemudahan penggunaan agar pengguna dapat melakukan proses masuk ke sistem tanpa mengalami kebingungan, sekaligus memastikan bahwa hanya pengguna yang memiliki hak akses yang dapat melanjutkan ke tahap berikutnya. *Wireframe* ini memberikan gambaran awal struktur antarmuka login sebelum sistem diimplementasikan menggunakan teknologi web.

Selanjutnya, sistem menampilkan Halaman Dashboard Admin yang berfungsi sebagai tampilan utama setelah pengguna berhasil melakukan login. Halaman ini digunakan untuk menyajikan informasi ringkas terkait data aset dan aktivitas peminjaman dalam sistem. Dashboard dirancang agar informasi utama dapat ditampilkan secara jelas dan mudah dipahami oleh administrator.

Pada tahap berikutnya, setelah proses autentikasi pengguna berhasil dilakukan melalui Halaman Login, sistem menampilkan Halaman Dashboard Admin sebagai antarmuka utama dalam pengelolaan sistem informasi aset. Halaman ini berfungsi untuk menyajikan ringkasan informasi terkait data aset dan aktivitas peminjaman yang tersimpan di dalam sistem. Perancangan Dashboard Admin difokuskan pada penyajian informasi secara ringkas dan terstruktur agar administrator dapat memperoleh gambaran awal kondisi data aset sebelum melakukan pengelolaan lebih lanjut. Dengan tampilan yang jelas dan mudah dipahami, Dashboard Admin membantu administrator dalam mengakses informasi utama melalui satu halaman utama.



Gambar 6. Implementasi *Interface* Halaman Dashboard Admin

Pada Gambar 6, tampilan Dashboard Admin menyajikan rekapitulasi data aset dalam bentuk ringkasan statistik yang meliputi Total Aset (150), Aset Tersedia (100), dan Aset Dipinjam (50). Penyajian informasi ini bertujuan memberikan gambaran awal mengenai kondisi data aset yang tersimpan di dalam sistem tanpa perlu membuka menu lain. Dengan adanya ringkasan statistik tersebut, administrator dapat mengetahui kondisi umum data aset secara cepat melalui satu tampilan utama.

Selain ringkasan statistik, dashboard juga menampilkan komponen Riwayat Peminjaman Terbaru yang berisi daftar aktivitas peminjaman aset yang dilakukan oleh pengguna. Informasi yang ditampilkan pada bagian ini mencakup data aset yang dipinjam serta status peminjaman yang sedang berlangsung. Keberadaan komponen ini membantu administrator dalam memantau aktivitas peminjaman berdasarkan data yang tersaji pada sistem, sehingga status penggunaan aset dapat diketahui melalui halaman utama.

Penyajian informasi pada halaman Dashboard Admin dirancang secara ringkas dan terstruktur agar mudah dipahami oleh pengguna. Tata letak elemen antarmuka disusun untuk menampilkan informasi penting secara jelas, sehingga administrator dapat memperoleh gambaran awal kondisi data aset dan aktivitas peminjaman sebelum melakukan pengelolaan lebih lanjut melalui menu yang tersedia pada sistem.

Untuk melengkapi proses pencatatan aset, sistem menyediakan halaman input rincian aset yang memungkinkan administrator memasukkan data aset baru secara lengkap dan terstruktur. Halaman ini dirancang agar alur pengisian data mudah diikuti serta sesuai dengan proses pencatatan yang telah ditentukan dalam sistem, sehingga data aset yang dimasukkan dapat tersimpan dengan baik dan siap digunakan pada proses pengelolaan selanjutnya.

Selain berfungsi sebagai tampilan ringkasan data, halaman Dashboard Admin juga berperan sebagai titik awal navigasi bagi administrator untuk mengakses fitur-fitur pengelolaan aset yang tersedia pada sistem. Melalui tampilan ini, administrator dapat berpindah ke menu pencatatan aset, peminjaman, maupun pengelolaan data lainnya sesuai kebutuhan operasional. Dengan adanya dashboard sebagai halaman utama, alur penggunaan sistem menjadi lebih terstruktur karena pengguna diarahkan terlebih dahulu pada informasi umum sebelum melakukan proses pengelolaan data secara lebih detail.

Keberadaan Dashboard Admin sebagai halaman utama juga membantu dalam menjaga konsistensi alur penggunaan sistem. Administrator diarahkan untuk selalu memulai interaksi melalui halaman ini sehingga informasi yang ditampilkan dapat menjadi acuan awal sebelum melakukan aktivitas pengelolaan data. Dengan alur tersebut, sistem tidak hanya menampilkan data, tetapi juga memberikan konteks awal terkait kondisi aset dan aktivitas peminjaman yang sedang berlangsung berdasarkan data yang tersimpan di dalam sistem. Pendekatan ini mendukung penggunaan sistem yang lebih terstruktur karena setiap proses pengelolaan data diawali dari satu titik tampilan utama yang sama.

Gambar 7. Rancangan Halaman *Input Rincian Aset*

Form input terdiri dari nama aset, kode aset, kategori, serta elemen tambahan yang relevan. Tampilan yang sederhana dan intuitif ini memudahkan admin dalam melakukan pencatatan, sekaligus meminimalkan kesalahan input. Desain halaman mengikuti prinsip *user-centered design*, di mana kenyamanan dan kemudahan interaksi pengguna menjadi prioritas utama.

### 3.3. Pengujian Sistem

Pengujian sistem dilakukan untuk memastikan bahwa seluruh fitur yang dikembangkan berfungsi sesuai spesifikasi dan kebutuhan pengguna. Pada tahap ini, digunakan *Black Box Testing*, yaitu metode pengujian yang memfokuskan pada output yang dihasilkan dari input tertentu, tanpa memperhatikan struktur internal kode program. Pendekatan ini berguna untuk mengevaluasi fungsionalitas sistem secara menyeluruh, termasuk proses pencatatan aset, peminjaman, monitoring, dan pengaduan.

Hasil pengujian menunjukkan bahwa seluruh fitur utama sistem berjalan sesuai harapan, sehingga dapat mendukung aktivitas pengelolaan aset secara efektif dan efisien.

Tabel 3. Hasil Pengujian *Black Box*

No	Fitur Sistem	Input Uji	Output yang Diharapkan	Hasil
1	Pencatatan Aset	Data aset lengkap	Data tersimpan di database	Berhasil
2	Peminjaman Aset	Request peminjaman valid	Status peminjaman tercatat	Berhasil
3	Pengaduan	Form keluhan lengkap	Keluhan tersimpan dan diteruskan	Berhasil
4	Monitoring	Pilih aset	Informasi aset ditampilkan	Berhasil

Berdasarkan hasil pengujian ini, sistem menunjukkan stabilitas dan konsistensi dalam menjalankan fungsi-fungsi utamanya. *Black Box Testing* memastikan bahwa semua fitur dapat beroperasi tanpa kesalahan, sehingga proses pencatatan, peminjaman, monitoring, dan pengaduan aset dapat dilakukan secara akurat dan handal.

### 3.4. Hasil UAT – Skala Likert

Sebanyak 20 responden dilibatkan dalam *User Acceptance Test* (UAT) untuk mengevaluasi tingkat kepuasan terhadap sistem yang dikembangkan. Penilaian difokuskan pada empat aspek utama, yaitu kemudahan navigasi, tampilan antarmuka, kecepatan sistem, dan fungsionalitas. Setiap responden memberikan skor menggunakan skala Likert 1–5, di mana 1 menunjukkan Sangat Tidak Setuju dan 5 menunjukkan Sangat Setuju.

Tabel 4. Hasil UAT – Skala Likert

No	Aspek Sistem	1	2	3	4	5	Skor Rata-rata
1	Kemudahan Navigasi	0	0	1	5	14	4,6
2	Tampilan Antarmuka	0	0	2	6	12	4,4
3	Kecepatan Sistem	0	1	2	5	12	4,3
4	Fungsionalitas	0	0	1	6	13	4,5

Perhitungan skala Likert dilakukan dengan rumus berikut:

$$X = \frac{\sum(fi \times si)}{N}$$

Dengan Contoh Perhitungan:

$$X = \frac{(0 \times 1) + (0 \times 2) + (1 \times 3) + (5 \times 4) + (14 \times 5)}{20} = 4,6$$

Berdasarkan hasil pengujian, seluruh aspek memperoleh skor rata-rata tinggi, yang mengindikasikan bahwa pengguna menilai sistem ini sangat memuaskan. Secara khusus, skor tertinggi diperoleh pada kemudahan navigasi (4,6), menunjukkan bahwa pengguna dapat mengoperasikan sistem dengan lancar dan tanpa hambatan. Hasil ini memperkuat kesimpulan bahwa sistem telah dikembangkan sesuai dengan prinsip *user-centered design* dan mampu memenuhi ekspektasi pengguna dalam konteks pengelolaan aset.

### 3.5. Rekapitulasi Penilaian UAT

Berdasarkan hasil *User Acceptance Test* (UAT), dilakukan rekapitulasi skor rata-rata untuk masing-masing aspek penilaian sistem. Hasil evaluasi menunjukkan bahwa seluruh aspek memperoleh skor tinggi, yang menandakan tingkat kepuasan pengguna yang sangat baik.

Tabel 5. Rekapitulasi Penilaian UAT

No	Aspek Sistem	Skor Rata-rata	Kategori
1	Kemudahan Navigasi	4,6	Sangat Baik
2	Tampilan Antarmuka	4,4	Sangat Baik
3	Kecepatan Sistem	4,3	Sangat Baik
4	Fungsionalitas	4,5	Sangat Baik
Rata-rata Keseluruhan		4,45	Sangat Baik

Dari rekapitulasi ini dapat disimpulkan bahwa sistem memiliki kualitas antarmuka, navigasi, kecepatan, dan fungsionalitas yang sangat baik, sehingga dapat memenuhi kebutuhan dan ekspektasi pengguna. Hasil ini menunjukkan bahwa pengembangan sistem telah sesuai dengan prinsip *user-centered design* dan standar kualitas perangkat lunak yang diharapkan.

## 4. Kesimpulan dan Saran

### 4.1. Kesimpulan

Berdasarkan hasil penelitian dan pengembangan sistem informasi manajemen aset berbasis web, dapat disimpulkan bahwa sistem yang dikembangkan mendukung proses pencatatan, peminjaman, dan penyajian informasi aset secara terstruktur dan terintegrasi. Implementasi arsitektur *Model-View-Controller* (MVC) pada *framework CodeIgniter 4* menghasilkan struktur sistem yang terorganisir dan mudah dipelihara. Pengujian menggunakan *Black Box Testing* menunjukkan bahwa seluruh fitur utama

---

sistem berfungsi sesuai dengan kebutuhan fungsional. Selain itu, *User Acceptance Testing* (UAT) dengan skala Likert menunjukkan tingkat kepuasan pengguna yang sangat baik pada aspek navigasi, tampilan antarmuka, kecepatan, dan fungsionalitas sistem. Dengan demikian, sistem yang dikembangkan dapat diterima dengan baik oleh pengguna dan berpotensi digunakan sebagai pendukung pengelolaan aset di lingkungan sekolah.

#### 4.2. Saran

Berdasarkan pengalaman pengembangan dan pengujian sistem, beberapa saran untuk penelitian atau pengembangan selanjutnya adalah:

1. Pengembangan fitur tambahan, seperti modul notifikasi otomatis untuk peminjaman dan pengembalian aset, diharapkan dapat membantu proses pengawasan dan pengingat bagi pengguna.
2. Integrasi sistem dengan basis data sekolah atau sistem keuangan dapat memperluas fungsi pengelolaan dan pemantauan data aset secara terintegrasi.
3. Peningkatan antarmuka pengguna, termasuk dashboard interaktif dan fitur personalisasi, dapat meningkatkan pengalaman pengguna agar lebih intuitif dan informatif.
4. Pengujian lebih lanjut dengan jumlah responden yang lebih besar dan beragam, serta pengujian performa sistem, dapat dilakukan untuk mengetahui kemampuan sistem dalam menangani volume data aset yang lebih besar.

#### Daftar Pustaka

- [1] M. Alfiareza, and N. Wilantika, "SISTEM PELACAKAN Dokumen Berbasis Seluler dan Kode QR," vol. 9, no. 3, pp. 453–462, 2022, doi: 10.25126/jtiik.202294519.
- [2] E. P. Anggraeni and J. Pamungkas, "Sarana dan Prasarana Lembaga dalam Menciptakan Potensi Pengembangan Seni Anak Usia Dini," *Jurnal Obsesi : Jurnal Pendidikan Anak Usia Dini*, vol. 7, no. 1, pp. 85–93, Jan. 2023, doi: 10.31004/obsesi.v7i1.2864.
- [3] I. Hamidah, I. Nugroho, and S. Surejo, "Penerapan Interaksi Manusia dan Komputer Pada Antarmuka Sistem Informasi Akademik," 2023.
- [4] S. N. Safira, Y. T. Mursityo, and M. C. Saputra, "Pengembangan Sistem Monitoring Pendataan Aplikasi Berbasis Web pada Kementerian Pekerjaan Umum dan Perumahan Rakyat," *Jurnal Teknologi Informasi dan Ilmu Komputer*, vol. 10, no. 5, pp. 983–992, Oct. 2023, doi: 10.25126/jtiik.2023106891.
- [5] J. Tabita Tanod, H. Jati Setyadi, and V. Zahrotun Kamila, "Perancangan User Interface (UI) & User Experience (UX) Aplikasi SI Presensi Berbasis Mobile Menggunakan Metode Design Thinking," *SIMKOM*, vol. 10, no. 2, pp. 182–195, Jul. 2025, doi: 10.51717/simkom.v10i2.605.
- [6] Deden Herdiana, *Interaksi Manusia dan Komputer*. Informatika Bandung, 2021.
- [7] Ian Sommerville, *Software Engineering (10th Edition)*. Pearson Education, 2020.
- [8] Kenneth E. Kendall & Julie E. Kendall, *Systems Analysis and Disign (10th Edition)*. Pearson, 2019.
- [9] Roger S. Pressman & Bruce R. Maxim, *Shoftware Engineering A Practitioner's Approach (9th Edition)*. McGraw-Hill, 2019.
- [10] Y. Prastyaningsih, M. N. Ridha, N. M. Aristi, and R. Husnah, "Rancang Bangun Sistem Informasi Si Ujikom Lembaga Sertifikasi Profesi SMKN 1 Pelaihari," *SIMKOM*, vol. 10, no. 2, pp. 224–230, Jul. 2025, doi: 10.51717/simkom.v10i2.869.
- [11] Putri Nabella, Rudy Herteno, Setyo Wahyu Saputro, Friska Abadi, and Muhammad Itqan Mazdadi, "Pengembangan Sistem Manajemen Sarana Dan Prasarana, IT, Serta Laboratorium di SMK Telekomunikasi," *Jurnal Teknologi Informasi dan Ilmu Komputer*, vol. 12, no. 1, pp. 1–10, Feb. 2025, doi: 10.25126/jtiik.2025128649.