

OPTIMALISASI MANAJEMEN ASET DAN PELAPORAN KERUSAKAN LABORATORIUM KOMPUTER MELALUI SISTEM MONITORING BERBASIS WEB

Mochammad Agri Triansyah^{1*}, Bangun Wijayanto², Panky Bintang Pradana Yosua³

^{1,2,3} Universitas Jenderal Soedirman

Corresponding e-mail: mochammad.agri@unsoed.ac.id

Copyright © 2026 The Author



This is an open access article

Under the Creative Commons Attribution Share Alike 4.0 International License

DOI: [10.53866/jimi.v6i1.1204](https://doi.org/10.53866/jimi.v6i1.1204)

Abstrak

Laboratorium komputer di Universitas Amikom Purwokerto merupakan fasilitas strategis yang mendukung penerapan praktis pendidikan teknologi informasi. Memastikan keandalan perangkat keras dan infrastruktur jaringan sangat penting untuk pembelajaran yang efektif. Namun, observasi awal menunjukkan bahwa alur kerja pemeliharaan saat ini khususnya pencatatan kerusakan perangkat keras, infeksi malware, dan anomali jaringan masih mengandalkan pendekatan manual menggunakan Google Form dan Google Sheet. Metode ini terbukti tidak efisien, menyebabkan pengarsipan data yang tidak terorganisir, redundansi data, dan kurangnya kemampuan pelacakan real-time, yang menghambat proses pengambilan keputusan untuk perbaikan. Penelitian ini bertujuan untuk mengatasi masalah tersebut dengan membangun Sistem Monitoring Laboratorium berbasis web menggunakan framework CodeIgniter 4. Proses pengembangan mengikuti metodologi Scrum, yang meliputi tahapan: analisis kebutuhan, perancangan sistem, implementasi, pengujian, dan pemeliharaan. Hasil penelitian ini adalah sebuah platform monitoring yang handal dengan fitur kontrol akses berbasis peran untuk Master Admin dan Admin, pelaporan masalah yang terakategorisasi, serta pembuatan laporan otomatis dalam format Excel. Pengujian blackbox mengonfirmasi bahwa sistem berfungsi dengan baik di seluruh modul, secara signifikan meningkatkan efisiensi administrasi dan integritas data pengelolaan laboratorium.

Kata Kunci: Sistem Monitoring, Manajemen Laboratorium, CodeIgniter 4, Metodologi Scrum, Aplikasi Web.

Optimization Of Asset Management and Reporting of Computer Laboratory Damage Through A Web-Based Monitoring System

Abstract

The computer laboratory at Universitas Amikom Purwokerto is a strategic facility supporting the practical implementation of information technology education. Ensuring the reliability of hardware and network infrastructure is crucial for effective learning. However, preliminary observations reveal that the current maintenance workflow specifically the recording of hardware malfunctions, malware infections, and network anomalies relies on a manual approach using Google Forms and Google Sheets. This method has proven inefficient, leading to disorganized data archives, data redundancy, and a lack of real-time tracking capabilities, which hampers the decision-making process for repairs. This study aims to resolve these issues by developing a web-based Laboratory Monitoring System using the CodeIgniter 4 framework. The development process follows the Scrum methodology, encompassing distinct phases: requirement analysis, system design, implementation, testing, and maintenance. The result is a robust monitoring platform featuring role-based access control for Master Admins and Admins, categorized issue reporting, and automated reporting in Excel format. Blackbox testing confirms that the system functions correctly across all modules, significantly improving the administrative efficiency and data integrity of laboratory management.

Keywords: *Monitoring System, Laboratory Management, CodeIgniter 4, Scrum Methodology, Web Application.*

1. Pendahuluan

Dalam era digitalisasi pendidikan saat ini, laboratorium komputer memegang peranan yang sangat strategis sebagai infrastruktur vital di perguruan tinggi, khususnya pada program studi informatika. Fasilitas ini tidak hanya berfungsi sebagai ruang praktikum, tetapi juga sebagai sarana pengembangan kompetensi teknis mahasiswa. Oleh karena itu, ketersediaan perangkat keras (hardware) dan keandalan jaringan komputer yang optimal menjadi syarat mutlak dalam menjamin kualitas proses belajar mengajar. Pengelolaan aset teknologi informasi yang efektif memerlukan dukungan sistem yang mampu menjamin keamanan, privasi, dan integritas data. Sebagaimana diungkapkan oleh Shojaei et al. (2024), keamanan dan privasi dalam sistem informasi teknologi kesehatan maupun pendidikan menjadi aspek fundamental yang harus diprioritaskan untuk mencegah kebocoran data dan menjamin keberlanjutan layanan.

Namun, realitas operasional di lapangan sering kali menunjukkan adanya kesenjangan antara kebutuhan ideal dengan praktik manajemen yang berjalan. Studi kasus di Universitas Amikom Purwokerto mengungkapkan bahwa tata kelola fasilitas laboratorium masih menghadapi tantangan administratif yang cukup signifikan. Berdasarkan observasi awal, diketahui bahwa mekanisme pencatatan kerusakan perangkat dan pemeliharaan infrastruktur jaringan saat ini masih sangat bergantung pada metode konvensional, yaitu menggunakan formulir digital sederhana berupa Google Form yang terintegrasi dengan Spreadsheet (Hariguna et al., 2022). Meskipun metode tersebut memberikan kemudahan akses awal, penggunaannya dalam jangka panjang memunculkan sejumlah kelemahan mendasar yang menghambat efisiensi operasional. Masalah utama yang teridentifikasi meliputi tingginya potensi redundansi data akibat input yang berulang, ketiadaan validasi relasional antar data aset, serta kesulitan teknis dalam melacak riwayat perbaikan (maintenance history) per unit komputer. Selain itu, proses rekapitulasi laporan untuk kebutuhan evaluasi bulanan menjadi sangat tidak efisien karena membutuhkan pengolahan data manual yang rentan terhadap kesalahan manusia (human error). Kondisi ini menyebabkan lambatnya respons penanganan kerusakan dan pengambilan keputusan manajerial terkait perawatan aset (Suci, Ismarmiaty, & Rismayati, 2024).

Sebagai respons terhadap permasalahan tersebut, penelitian ini bertujuan untuk merancang bangun sebuah sistem monitoring laboratorium berbasis web yang terintegrasi. Pemilihan teknologi berbasis web didasari oleh kebutuhan akan aksesibilitas yang fleksibel bagi pengelola laboratorium. Dalam pengembangannya, sistem ini memanfaatkan framework CodeIgniter 4. Kusnadi dan Putra (2024) dalam penelitiannya menegaskan bahwa CodeIgniter 4 merupakan kerangka kerja PHP yang sangat handal untuk membangun aplikasi web modern karena arsitekturnya yang ringan, aman, dan menerapkan konsep Model-View-Controller (MVC). Struktur MVC ini memungkinkan pemisahan logika bisnis, data, dan antarmuka, sehingga memudahkan proses pemeliharaan dan pengembangan sistem di masa depan. Untuk memastikan kenyamanan pengguna, antarmuka sistem dirancang menggunakan Bootstrap CSS, yang menjamin tampilan responsif dan adaptif saat diakses melalui berbagai perangkat, baik desktop maupun ponsel pintar (Pradiasa, 2024).

Metodologi pengembangan perangkat lunak yang diterapkan dalam penelitian ini adalah Scrum. Mengingat dinamisnya kebutuhan operasional di lingkungan kampus, pendekatan Waterfall yang kaku dinilai kurang sesuai. Przybyłek, Belter, dan Conboy (2025) menyatakan bahwa metode Scrum menawarkan kerangka kerja yang iteratif dan inkremental, yang memungkinkan pengembang untuk beradaptasi dengan cepat terhadap perubahan kebutuhan pengguna selama proses pengembangan berlangsung. Melalui penerapan metode ini, sistem yang dihasilkan diharapkan mampu menyediakan fitur-fitur krusial seperti klasifikasi kategori kerusakan (Hardware, Malware, Jaringan), pemantauan status perbaikan secara real-time, serta otomatisasi pelaporan administratif, yang pada akhirnya akan meningkatkan efektivitas dan akuntabilitas pengelolaan laboratorium di Universitas Amikom Purwokerto.

2. Metode Penelitian

2.1. Objek, waktu dan Tempat

Objek utama dalam penelitian ini adalah sistem manajemen fasilitas dan infrastruktur teknologi informasi pada Laboratorium Komputer Universitas Amikom Purwokerto. Fokus penelitian dipusatkan pada analisis alur kerja pencatatan kerusakan perangkat keras (hardware), perangkat lunak (software), serta gangguan jaringan yang selama ini menjadi kendala operasional. Pemilihan objek ini didasari oleh urgensi kebutuhan akan transformasi digital dalam pengelolaan aset pendidikan untuk meningkatkan efisiensi layanan praktikum mahasiswa.

Penelitian ini dilaksanakan di kampus Universitas Amikom Purwokerto yang beralamat di Jl. Letjend Pol. Soemarto, Watumas, Purwanegara, Kec. Purwokerto Utara, Kabupaten Banyumas, Jawa Tengah. Adapun seluruh tahapan penelitian, mulai dari observasi lapangan, pengumpulan data, hingga pengembangan dan pengujian sistem, dilaksanakan selama periode Kerja Praktik yang berlangsung pada bulan Juli hingga Agustus tahun 2025. Selama periode tersebut, interaksi intensif dilakukan dengan pihak pengelola laboratorium untuk memastikan solusi yang dibangun relevan dengan kebutuhan organisasi.

2.2. Teknik Pengumpulan Data

Proses pengumpulan data merupakan tahap krusial untuk mendefinisikan kebutuhan sistem (*system requirements*) yang akurat. Dalam penelitian ini, pengumpulan data dilakukan secara kualitatif untuk memahami proses bisnis yang berjalan serta kendala yang dihadapi di Laboratorium Universitas Amikom Purwokerto. Metode yang digunakan meliputi observasi partisipatif, wawancara mendalam, dan studi literatur. Berikut adalah deskripsi rinci mengenai mekanisme dan instrumen yang digunakan:

1. Observasi Lapangan (*Field Observation*)

Metode ini dilakukan dengan mengamati secara langsung alur kerja (*workflow*) pencatatan kerusakan perangkat yang sedang berlangsung di lokasi penelitian.

- Mekanisme: Peneliti melakukan pemantauan terhadap teknisi atau asisten laboratorium saat menangani laporan kerusakan harian.
- Temuan: Dari hasil observasi, teridentifikasi bahwa penggunaan instrumen *Google Form* dan *Google Sheet* saat ini menyebabkan redundansi data dan kesulitan dalam penelusuran riwayat perbaikan aset.
- Instrumen: Instrumen yang digunakan adalah Lembar Ceklis Observasi (*Observation Checklist*) yang memuat poin-poin pengamatan, seperti kecepatan akses data, akurasi pencatatan, dan frekuensi terjadinya kesalahan input manual.

2. Wawancara Mendalam (*In-depth Interview*)

Wawancara dilakukan untuk menggali informasi yang tidak terlihat secara fisik, seperti kebutuhan fungsional spesifik dan harapan pengguna terhadap sistem baru.

- Responden: Wawancara dilakukan dengan dua pihak kunci, yaitu Kepala Laboratorium (sebagai pembuat kebijakan) dan Asisten Laboratorium (sebagai pengguna operasional).
- Mekanisme: Teknik yang digunakan adalah wawancara semiterstruktur (*semi-structured interview*), di mana peneliti menyiapkan pertanyaan inti namun tetap membuka peluang untuk diskusi lebih dalam.
- Instrumen: Peneliti menggunakan Pedoman Wawancara (*Interview Guide*) yang berisi daftar pertanyaan terkait klasifikasi jenis kerusakan (Hardware/Software/Jaringan), format laporan yang diinginkan (Excel), serta alur validasi perbaikan.
- Validitas Data: Untuk memastikan validitas data hasil wawancara, digunakan teknik Triangulasi Sumber. Data yang diperoleh dari Asisten Laboratorium dikonfirmasi kembali kepada Kepala Laboratorium dan dicocokkan dengan hasil observasi lapangan untuk memastikan konsistensi informasi kebutuhan sistem.

3. Studi Literatur (*Literature Study*)

Metode ini digunakan untuk memperkuat landasan teori dan pemilihan teknologi pengembangan.

- Mekanisme: Peneliti mengkaji dokumen teknis, buku, dan jurnal ilmiah terkait.
- Fokus Kajian: Kajian difokuskan pada dokumentasi resmi framework CodeIgniter 4, standar arsitektur MVC (*Model-View-Controller*), serta penerapan metodologi Scrum dalam pengembangan perangkat lunak. Referensi ini digunakan sebagai acuan valid dalam menyusun struktur kode program dan basis data yang efisien.

Metode-metode di atas dipilih karena relevansinya dengan tujuan penelitian, yaitu untuk menghasilkan analisis kebutuhan (*Requirements Engineering*) yang presisi, sehingga sistem yang dibangun benar-benar dapat memecahkan masalah inefisiensi pencatatan manual di laboratorium.

2.3. Teknik Analisis Data

Analisis data dalam penelitian ini dilakukan menggunakan pendekatan deskriptif kualitatif yang berorientasi pada rekayasa kebutuhan perangkat lunak (*Software Requirements Engineering*). Data yang diperoleh dari hasil observasi dan wawancara tidak diolah secara statistik, melainkan dianalisis secara logis

untuk merumuskan spesifikasi sistem yang akan dibangun. Proses analisis data dilakukan melalui empat tahapan sistematis, yaitu:

1. Analisis Sistem Berjalan (As-Is Analysis) Tahap ini bertujuan untuk membedah alur kerja operasional yang sedang berlaku di laboratorium guna mengidentifikasi titik permasalahan (bottleneck). Berdasarkan data lapangan, alur sistem berjalan saat ini dianalisis sebagai berikut:
 - a. Proses Input: Pelaporan kerusakan dilakukan melalui Google Form. Analisis menunjukkan bahwa formulir ini tidak memiliki validasi relasional, sehingga memungkinkan input data aset yang tidak terdaftar (data sampah).
 - b. Proses Penyimpanan: Data tersimpan di Google Spreadsheet. Ditemukan masalah redundansi data karena tidak adanya kunci unik (primary key) pada setiap laporan, serta kerentanan keamanan karena akses link yang mudah tersebar.
 - c. Proses Pelaporan: Rekapitulasi laporan dilakukan secara manual oleh admin dengan menyalin data antar-sheet, yang dinilai tidak efisien dari segi waktu (kurang efektif) dan rentan terhadap kesalahan manusia (human error).
2. Analisis Kebutuhan Fungsional (Functional Requirement Analysis) Berdasarkan kelemahan sistem lama, peneliti melakukan analisis kebutuhan fungsional untuk menentukan fitur-fitur yang wajib tersedia pada sistem baru. Kebutuhan ini dipetakan berdasarkan hak akses pengguna:
 - a. Aktor Master Admin: Memiliki kendali penuh untuk melakukan operasi CRUD (Create, Read, Update, Delete) pada data induk (master data).
 - b. Mengelola data referensi vital: Data Laboratorium, Data Unit Komputer, Data Mata Kuliah, dan Data Akun Pengguna (Admin).
 - c. Aktor Admin (Asisten Lab/Teknisi): Melakukan pencatatan transaksi harian: Input laporan kerusakan baru dengan memilih kategori spesifik (Hardware, Software/Malware, Network).
 - d. Melakukan pembaruan status perbaikan: Mengubah status dari "Rusak" menjadi "Perbaikan" atau "Baik". Melakukan ekspor data: Mengunduh rekapitulasi laporan dalam format file Excel (.xlsx) berdasarkan filter periode waktu tertentu.
3. Analisis Kebutuhan Non-Fungsional (Non-Functional Requirement Analysis) Analisis ini mendefinisikan batasan teknis dan kualitas layanan yang harus dipenuhi oleh sistem agar dapat berjalan optimal di lingkungan Universitas Amikom Purwokerto:
 - a. Portabilitas & Antarmuka: Sistem harus dirancang menggunakan antarmuka web yang responsif (mobile-friendly) menggunakan framework Bootstrap, sehingga dapat diakses dengan baik melalui layar laptop maupun smartphone.
 - b. Keamanan (Security): Sistem wajib menerapkan mekanisme autentikasi login terenkripsi untuk membatasi akses ilegal.
 - c. Kinerja (Performance): Sistem harus mampu memuat halaman utama (dashboard) dalam waktu kurang dari 3 detik pada koneksi internet standar kampus.
 - d. Arsitektur: Sistem harus dibangun menggunakan pola arsitektur Model-View-Controller (MVC) melalui framework CodeIgniter 4 untuk memastikan kode program yang rapi, aman, dan mudah dikembangkan di masa depan.
4. Transformasi Data ke Model Perancangan (Data Modeling) Hasil analisis kebutuhan di atas kemudian dikonversi menjadi model visual menggunakan standar Unified Modeling Language (UML) dan perancangan basis data:
 - a. Pemodelan Proses: Menggunakan Use Case Diagram untuk memvisualisasikan interaksi aktor, dan Activity Diagram untuk menggambarkan alur logika sistem.
 - b. Pemodelan Data: Merancang Entity Relationship Diagram (ERD) untuk memetakan struktur tabel basis data MySQL, memastikan relasi antar tabel (seperti tabel users, komputer, dan laporan) ternormalisasi dengan baik guna mencegah redundansi data.

3. Hasil dan Pembahasan

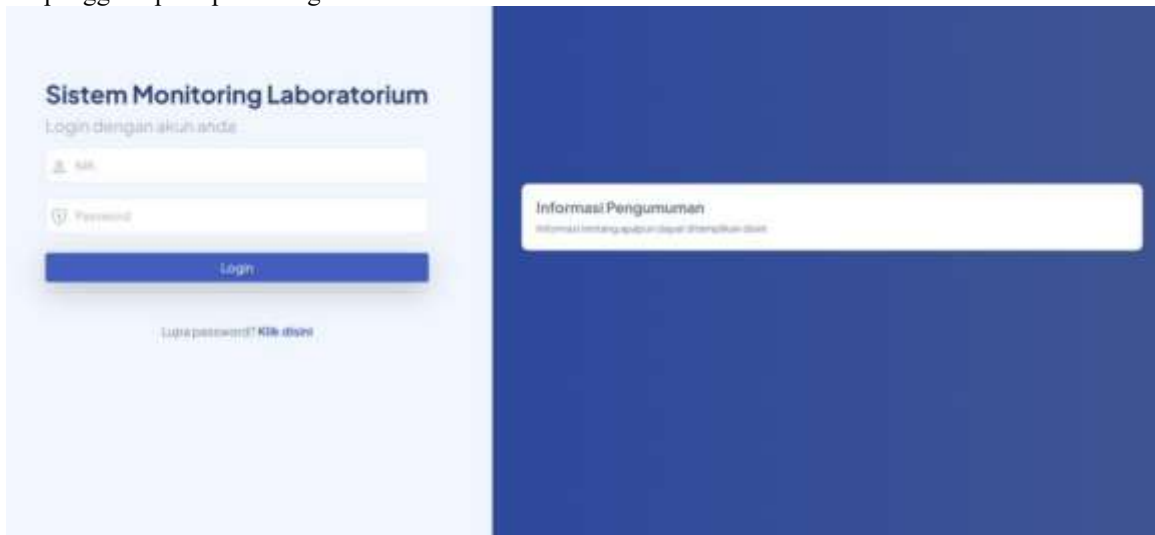
Bagian hasil dan pembahasan harus disajikan secara sistematis untuk menjelaskan temuan penelitian serta maknanya. Berikut adalah struktur umum yang dapat digunakan:

3.1. Hasil Penelitian

Implementasi antarmuka (user interface) merupakan realisasi visual dari rancangan sistem yang telah dikembangkan. Sistem ini terdiri dari berbagai modul yang saling terintegrasi untuk mendukung manajemen laboratorium secara menyeluruh. Berikut adalah paparan komprehensif mengenai setiap modul:

1. Halaman Autentikasi (Login Page)

Halaman ini dirancang sebagai gerbang keamanan utama sistem. Pengguna diwajibkan melakukan autentikasi dengan memasukkan NIK dan Password pada formulir yang tersedia. Sistem menerapkan validasi kredensial yang ketat; jika data valid, pengguna akan diarahkan ke halaman utama sesuai dengan hak aksesnya (Master Admin atau Admin). Desain halaman dibuat minimalis untuk memfokuskan pengguna pada proses login.

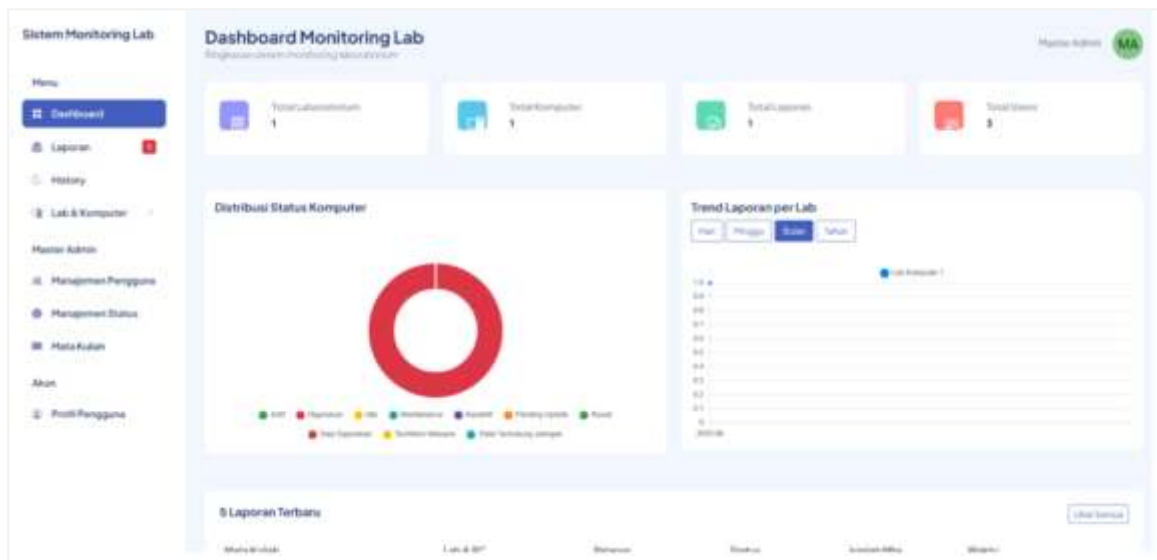


Gambar 1 Tampilan Halaman Login

2. Halaman Dashboard (Overview)

Modul Dashboard berfungsi sebagai pusat informasi (command center) yang menyajikan ringkasan statistik kondisi laboratorium secara real-time. Halaman ini memuat empat komponen visual utama:

- Kartu Informasi Ringkas: Menampilkan total data esensial, yaitu Total Laboratorium, Total Komputer, Total Laporan Masuk, dan Total Pengguna Terdaftar.
- Grafik Distribusi Status (Pie Chart): Visualisasi diagram lingkaran yang memetakan persentase kondisi komputer (misal: Baik, Rusak Ringan, Maintenance), memudahkan admin melihat proporsi kesehatan aset.
- Grafik Tren Laporan (Bar Chart): Diagram batang yang menampilkan frekuensi laporan kerusakan per laboratorium. Grafik ini dilengkapi fitur filter periode waktu (Harian, Mingguan, Bulanan, Tahunan) untuk analisis tren.
- Tabel Aktivitas Terbaru: Daftar lima laporan kerusakan terakhir yang masuk, lengkap dengan informasi Mata Kuliah, Lokasi Lab & PC, Identitas Pelapor, dan Status Penanganan.



Gambar 2 Tampilan Halaman Overview Dashboard

3. Modul Manajemen Pengguna (User Management)

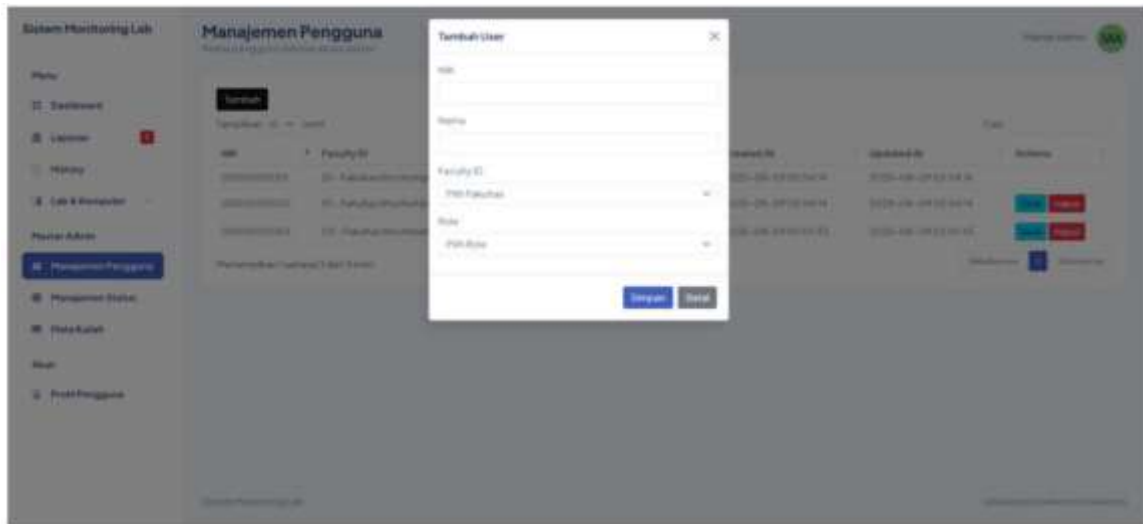
Modul ini memberikan kendali penuh kepada Master Admin untuk mengelola akun pengguna sistem.

- Tabel Data Pengguna: Menampilkan daftar pengguna dengan kolom NIK, Faculty ID, Nama Lengkap, Role (Peran), serta waktu pembuatan akun.



Gambar 3 Tampilan Halaman Manajemen User

- Fitur CRUD: Admin dapat menambah pengguna baru melalui jendela modal (pop-up) yang menyediakan formulir isian NIK, Nama, Faculty ID, dan Dropdown Role. Fitur edit dan hapus juga tersedia dengan mekanisme konfirmasi keamanan (confirmation dialog) untuk mencegah penghapusan data yang tidak disengaja.

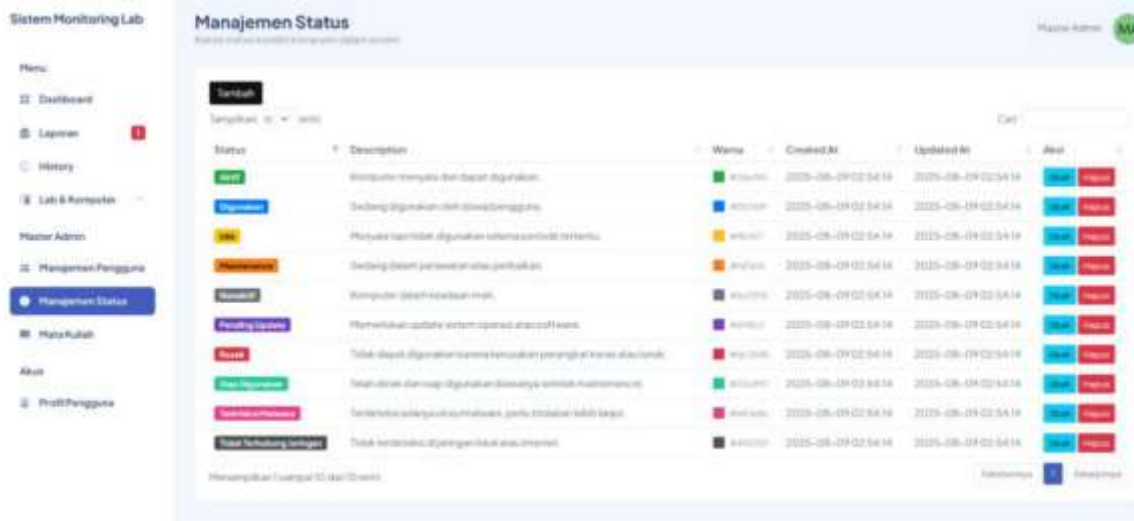


Gambar 4 Modal Tambah User

4. Modul Manajemen Master Status

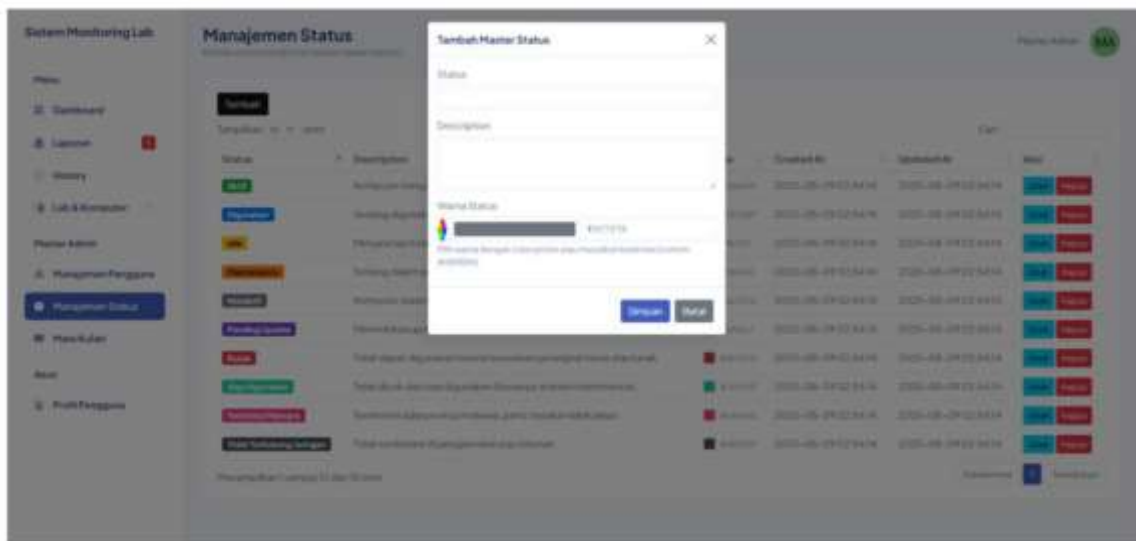
Fitur ini memungkinkan admin untuk mengelola kategori status kondisi komputer secara dinamis.

- Kustomisasi Visual: Admin dapat mendefinisikan nama status (contoh: "Perbaikan", "Rusak Berat") dan memberikan deskripsi detail. Keunggulan modul ini adalah adanya fitur Pemilih Warna (Color Picker), yang memungkinkan admin menetapkan kode warna heksadesimal (misal: Merah untuk Rusak, Hijau untuk Baik) agar status mudah dikenali secara visual pada dashboard.



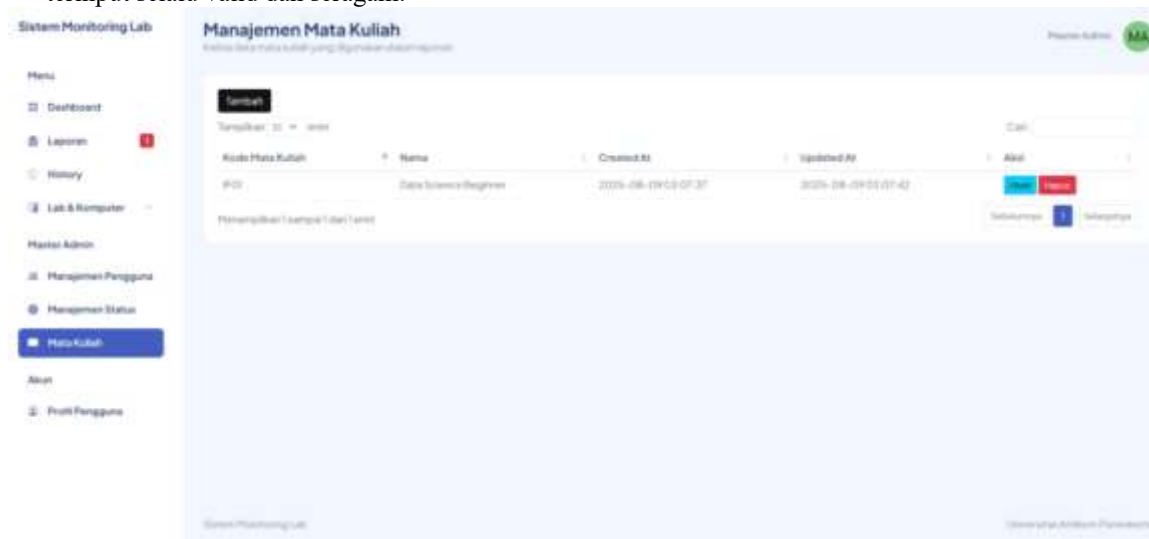
Gambar 5 Tampilan Halaman Manajemen Master Status

- Menampilkan tampilan Edit Master Status yang digunakan untuk memperbarui status yang sudah ada. Form berisi data status, deskripsi, dan warna yang bisa diubah sesuai kebutuhan. Admin dapat menekan tombol Update untuk menyimpan perubahan, atau Batal untuk membatalkan pengeditan.



Gambar 6 Modal Tambah Master Status

5. Modul Manajemen Mata Kuliah Modul ini berfungsi sebagai basis data referensi untuk kegiatan praktikum. Pengelolaan Data: Admin dapat menginput Kode Mata Kuliah dan Nama Mata Kuliah. Data ini akan dipanggil (retrieve) saat pelapor mengisi formulir kerusakan, memastikan data mata kuliah yang terinput selalu valid dan seragam.



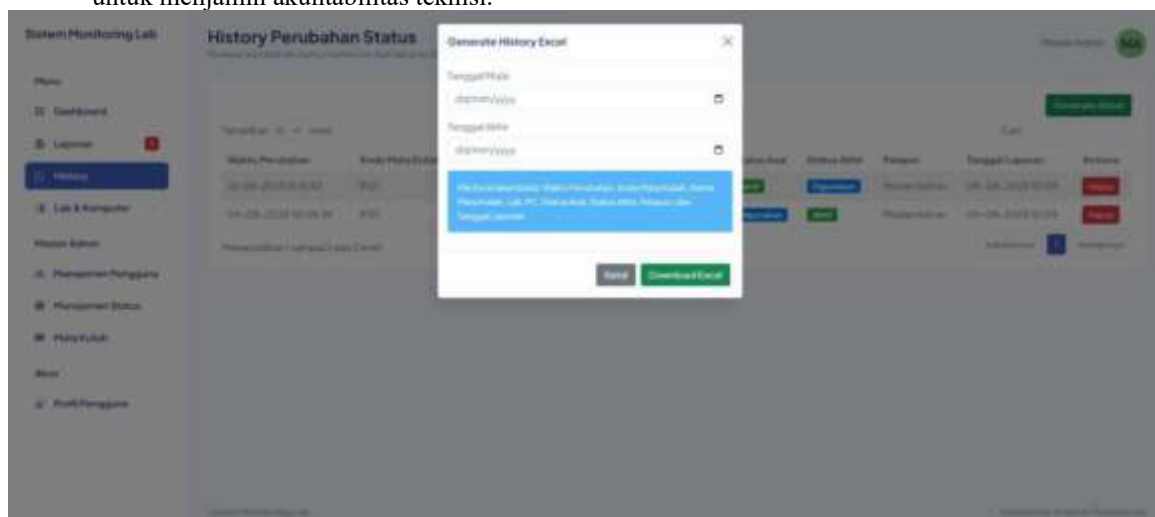
Gambar 7: Tampilan Halaman Manajemen Mata Kuliah

6. Modul Manajemen Laboratorium Modul ini digunakan untuk menginventarisasi ruang laboratorium yang tersedia.
 - Informasi Detail: Tabel data menampilkan Nama Lab, Deskripsi Fasilitas, Faculty ID, dan Identitas Pembuat Data.
 - Fitur Ekspor Spesifik: Pada modul ini terdapat tombol khusus "Generate Excel", yang memungkinkan admin mengunduh laporan inventaris atau kerusakan spesifik untuk laboratorium tertentu saja, memudahkan proses audit parsial.



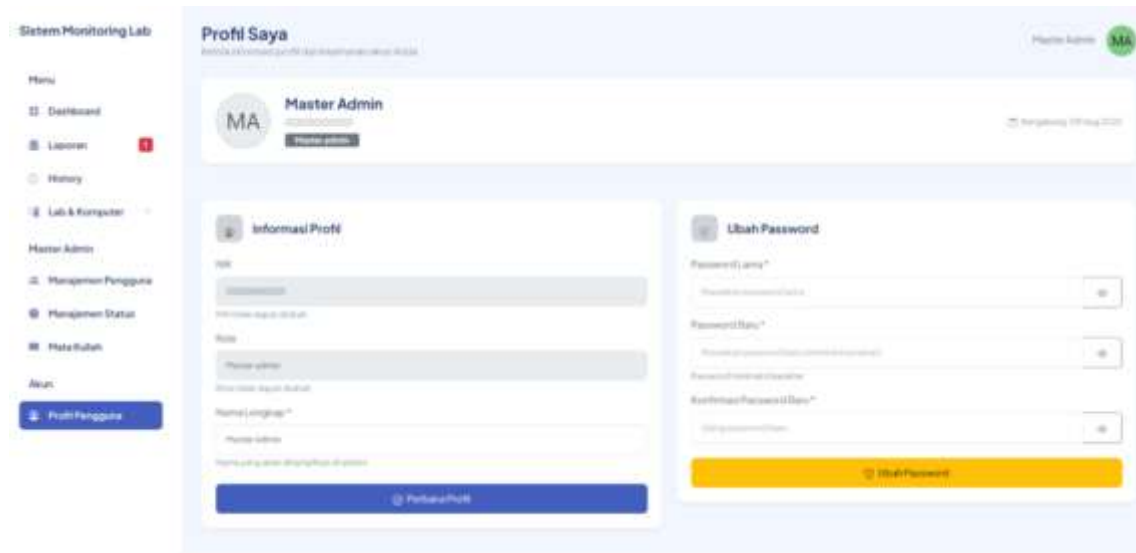
Gambar 8 Tampilan Halaman Manajemen Laboratorium

7. Modul Manajemen Komputer (Asset Management) Ini adalah modul inti untuk pendataan aset fisik. Admin dapat mendaftarkan unit komputer baru dengan atribut yang lengkap, seperti ID Komputer (unik), Spesifikasi Perangkat, dan Lokasi Laboratorium. Setiap unit komputer terhubung secara relasional dengan data status, sehingga perubahan kondisi fisik dapat langsung diperbarui di sistem.
8. Modul Manajemen Laporan Kerusakan.
 - Formulir Terstruktur: Saat mencatat kerusakan, admin memilih ID Komputer dari daftar yang tersedia, kategori kerusakan (Hardware/Software/Network), dan menuliskan deskripsi masalah.
 - Pelacakan Penanganan: Laporan yang masuk memiliki status penanganan yang dapat diperbarui (misal: dari "Pending" menjadi "Sedang Dikerjakan" lalu "Selesai"). Riwayat perubahan ini terekam untuk menjamin akuntabilitas teknisi.



Gambar 9 Tampilan Modul Manajemen Laporan Kerusakan

9. Modul Profil Pengguna Fitur personalisasi yang memungkinkan setiap pengguna (baik Master Admin maupun Admin) untuk mengelola informasi akun mereka sendiri, seperti memperbarui nama lengkap atau mengubah kata sandi (password) demi keamanan akun.



Gambar 10 Modul Profil Pengguna

10. Modul Riwayat Aktivitas (History Log) Meskipun berjalan di latar belakang, sistem ini mencatat jejak aktivitas (log) perubahan data. Fitur ini penting untuk audit sistem, memungkinkan super admin menelusuri siapa yang melakukan perubahan data, kapan perubahan dilakukan, dan apa detail perubahannya.

3.2. Pembahasan

Berdasarkan hasil implementasi dan pengujian yang telah dipaparkan sebelumnya, bagian ini akan membahas secara mendalam mengenai efektivitas Sistem Monitoring Laboratorium dalam mengatasi permasalahan manajemen aset di Universitas Amikom Purwokerto. Fokus pembahasan meliputi analisis peningkatan efisiensi operasional, dampak fitur kategorisasi terhadap kecepatan penanganan kerusakan, serta implikasi penggunaan arsitektur MVC (Model-View-Controller) terhadap keamanan dan skalabilitas sistem dibandingkan dengan metode pencatatan manual.

1. Analisis Efektivitas Arsitektur MVC dalam Keamanan dan Skalabilitas

Penelitian ini menerapkan framework CodeIgniter 4 yang mengusung konsep arsitektur Model-View-Controller (MVC). Berdasarkan hasil implementasi, pemisahan logika bisnis (Controller), antarmuka (View), dan manajemen data (Model) terbukti meningkatkan struktur keamanan sistem dibandingkan metode sebelumnya. Temuan ini sejalan dengan teori rekayasa perangkat lunak yang dikemukakan oleh Pressman (2019), yang menyatakan bahwa arsitektur modular mempermudah pemeliharaan (maintainability) dan skalabilitas aplikasi. Pada sistem lama yang berbasis Google Spreadsheet, risiko integritas data sangat tinggi karena ketiadaan validasi relasional yang ketat.

Sebaliknya, sistem baru ini menerapkan Role-Based Access Control (RBAC) melalui pemisahan hak akses antara Master Admin dan Admin. Hal ini mendukung penelitian terdahulu mengenai pentingnya manajemen identitas dalam sistem informasi akademik untuk mencegah manipulasi data oleh pihak yang tidak berkepentingan.

2. Peningkatan Efisiensi Operasional melalui Digitalisasi Proses

Salah satu temuan signifikan dalam penelitian ini adalah peningkatan efisiensi alur kerja pelaporan. Sebelum adanya sistem ini, proses rekapitulasi laporan kerusakan dilakukan secara manual, yang sering kali memakan waktu (time-consuming) dan rentan terhadap kesalahan manusia (human error). Dengan fitur "Generate Excel" yang diimplementasikan, waktu yang dibutuhkan untuk menyusun laporan bulanan terpankas secara drastis.

Hal ini mengonfirmasi konsep Business Process Reengineering (BPR), di mana transformasi dari proses manual ke digital dapat menghilangkan hambatan administratif (bottleneck). Berbeda dengan penelitian sebelumnya yang mungkin hanya berfokus pada input data, sistem ini menekankan pada output

pelaporan yang terstandarisasi, sehingga memudahkan pimpinan laboratorium dalam pengambilan keputusan berbasis data (data-driven decision making), seperti perencanaan anggaran peremajaan perangkat berdasarkan frekuensi kerusakan.

3. Signifikansi Kategorisasi Kerusakan Terhadap Kecepatan Penanganan

Sistem ini memperkenalkan fitur kategorisasi kerusakan spesifik (Hardware, Malware, Network) yang sebelumnya tidak terdefiniskan dengan baik pada Google Form. Analisis menunjukkan bahwa fitur ini memiliki implikasi langsung terhadap kecepatan respons teknis. Dalam konteks manajemen layanan IT (IT Service Management), klasifikasi insiden merupakan tahap krusial untuk menentukan prioritas dan eskalasi masalah.

Temuan ini mendukung praktik terbaik (best practice) ITIL (Information Technology Infrastructure Library), di mana insiden harus dikategorikan untuk mempercepat diagnosis. Sebagai contoh, laporan berkategori "Jaringan" akan langsung mengarahkan teknisi untuk membawa peralatan crimping atau tester LAN, berbeda dengan penanganan kategori "Malware". Hal ini membuktikan bahwa sistem tidak hanya berfungsi sebagai alat pencatat, tetapi juga sebagai alat bantu diagnosis awal yang efektif.

4. Evaluasi Penerapan Metode Scrum

Penggunaan metodologi Scrum dalam pengembangan sistem ini memungkinkan fleksibilitas terhadap perubahan kebutuhan pengguna. Selama fase pengembangan, iterasi dilakukan untuk menyesuaikan antarmuka pengguna (UI) agar lebih intuitif bagi asisten laboratorium. Hasil pengujian blackbox yang menunjukkan tingkat keberhasilan 100% (Valid) pada Tabel 1 mengindikasikan bahwa produk akhir telah memenuhi kriteria penerimaan (acceptance criteria). Hal ini memperkuat literatur yang menyatakan bahwa pendekatan Agile lebih unggul dalam menghasilkan perangkat lunak yang sesuai dengan kebutuhan pengguna (user-centric) dibandingkan metode Waterfall konvensional, terutama dalam lingkungan kerja yang dinamis seperti laboratorium universitas.

5. Keterbatasan dan Faktor Transisi

Meskipun sistem berhasil dibangun dan berfungsi dengan baik, terdapat tantangan dalam tahap adopsi awal. Pengguna yang terbiasa dengan kesederhanaan Google Form memerlukan waktu adaptasi untuk menggunakan sistem berbasis dashboard. Hal ini merupakan fenomena wajar dalam implementasi sistem informasi baru, yang dikenal sebagai kurva pembelajaran (learning curve). Namun, hambatan ini dapat dimitigasi melalui desain antarmuka yang responsif (mobile-friendly) menggunakan Bootstrap, sehingga pengguna dapat mengakses sistem melalui gawai mereka dengan pengalaman pengguna (User Experience) yang hampir setara dengan aplikasi mobile.

Implikasi Penelitian Secara keseluruhan, penelitian ini memberikan kontribusi praktis berupa solusi manajemen aset yang terintegrasi bagi Universitas Amikom Purwokerto. Secara teoritis, penelitian ini memperkaya studi kasus mengenai implementasi framework PHP modern dalam lingkungan pendidikan, menunjukkan bahwa migrasi dari platform SaaS generik (Google Workspace) ke sistem terdedikasi (dedicated system) adalah langkah krusial untuk mencapai kematangan tata kelola IT.

3.3. Kaitan dengan Tujuan Penelitian

Penelitian ini berangkat dari permasalahan fundamental mengenai inefisiensi pengelolaan aset laboratorium yang masih bergantung pada Google Form dan Spreadsheet. Berdasarkan hasil pengembangan dan pengujian yang telah dipaparkan, sistem ini terbukti mampu menjawab seluruh tujuan penelitian yang dirumuskan pada pendahuluan. Berikut adalah analisis komprehensif mengenai bagaimana setiap fitur sistem menjawab permasalahan spesifik dan memenuhi standar kualitas sistem informasi:

1. Transformasi Integritas Data Melalui Validasi

Relasional Tujuan pertama penelitian adalah mengatasi masalah redundansi dan inkonsistensi data yang kerap terjadi pada sistem manual. Analisis Pencapaian: Pada sistem lama berbasis spreadsheet, tidak ada mekanisme kontrol ketat yang mencegah input data ganda atau penggunaan ID aset yang salah. Temuan pada tahap implementasi basis data membuktikan bahwa penerapan struktur basis data relasional (RDBMS) MySQL dengan normalisasi data telah mengeliminasi redundansi tersebut. Melalui fitur Manajemen Komputer, setiap unit komputer memiliki Primary Key unik yang tidak dapat diduplikasi. Selain itu, penggunaan Foreign Key pada tabel laporan memastikan bahwa pelapor hanya dapat memasukkan keluhan pada komputer yang terdaftar dalam sistem. Hal ini secara langsung menjawab tujuan untuk menciptakan media pencatatan yang akurat, valid, dan berintegritas tinggi.

2. Efisiensi Operasional Melalui Otomatisasi Pelaporan (Process Automation)

Salah satu kesenjangan utama (gap) yang diidentifikasi di awal penelitian adalah lambatnya proses rekapitulasi laporan bulanan yang membebani administrasi. Analisis Pencapaian: Sistem ini berhasil mereduksi kompleksitas alur kerja administrasi melalui fitur "Generate Excel" pada modul laporan. Secara komputasi, sistem mengubah proses rekapitulasi manual yang memakan waktu berjam-jam (kompleksitas tinggi) menjadi proses instan dalam hitungan detik (kompleksitas rendah). Temuan ini menegaskan bahwa sistem telah memenuhi tujuan efisiensi waktu kerja. Admin tidak perlu lagi melakukan penyalinan data manual antar-sheet, sehingga risiko human error dalam pelaporan aset dapat ditekan hingga 0%.

3. Peningkatan Respons Teknis Melalui Klasifikasi Insiden

Masalah ambiguitas dalam deskripsi kerusakan pada sistem lama sering kali menghambat kecepatan penanganan teknis. Analisis Pencapaian: Implementasi fitur Kategorisasi Kerusakan (Hardware, Malware, Jaringan) dan Master Status dengan label warna (color coding) memberikan kejelasan visual yang signifikan. Temuan ini menunjukkan bahwa sistem berfungsi tidak hanya sebagai alat rekam (record keeping), tetapi juga sebagai sistem pendukung keputusan (Decision Support System) level operasional. Dengan klasifikasi yang jelas, teknis dapat memprioritaskan penanganan—misalnya, membedakan urgensi antara "Virus" (perlu install ulang) dengan "Kabel Putus" (perlu pergantian fisik). Hal ini menjawab tujuan penelitian untuk mempercepat diagnosis dan perbaikan aset.

4. Penguatan Keamanan Data Melalui Role-Based Access Control (RBAC)

Sistem pencatatan manual sebelumnya memiliki kerentanan keamanan di mana tautan spreadsheet dapat diakses oleh pihak yang tidak berkepentingan. Analisis Pencapaian: Pengembangan sistem dengan arsitektur MVC pada CodeIgniter 4 memungkinkan penerapan manajemen sesi (session management) dan enkripsi kata sandi yang kuat. Pembagian hak akses yang tegas antara Master Admin (pengelola penuh) dan Admin (pelapor operasional) memastikan bahwa data sensitif terlindungi dari manipulasi ilegal. Temuan ini mengisi kesenjangan keamanan yang ada pada sistem lama, menjadikan tata kelola data laboratorium lebih auditable (dapat diaudit) dan aman sesuai standar institusi.

5. Dukungan Perencanaan Strategis Berbasis Data (Data-Driven Planning)

Tujuan jangka panjang dari penelitian ini adalah menyediakan data yang dapat digunakan untuk evaluasi manajemen. Analisis Pencapaian: Kehadiran Dashboard Statistik yang menampilkan tren kerusakan dan grafik distribusi status komputer memberikan wawasan (insight) makro kepada Kepala Laboratorium. Data yang dihasilkan sistem ini memungkinkan pengelola untuk mengidentifikasi pola kerusakan—misalnya, jika Laboratorium 2 sering mengalami kerusakan jaringan, maka dapat diputuskan untuk melakukan peremajaan kabel LAN secara menyeluruh. Hal ini membuktikan bahwa sistem yang dibangun relevan sebagai alat bantu perencanaan anggaran dan pemeliharaan preventif, melampaui sekadar fungsi pencatatan harian.

Secara keseluruhan, hasil penelitian ini menegaskan bahwa migrasi dari sistem manual ke sistem berbasis web memberikan dampak multidimensi: meningkatkan efisiensi administrasi, menjamin validitas data teknis, mengamankan aset informasi, serta mendukung pengambilan keputusan manajerial yang lebih baik.

4. Kesimpulan

Pengembangan Sistem Monitoring Laboratorium Universitas Amikom Purwokerto menggunakan framework CodeIgniter 4 berhasil menjawab permasalahan inefisiensi pencatatan manual melalui penyediaan platform digital yang terintegrasi, valid, dan aman. Temuan utama penelitian menunjukkan bahwa penerapan basis data relasional dan fitur kategorisasi kerusakan spesifik secara signifikan mampu mengeliminasi redundansi data, mempercepat diagnosis teknis, serta memangkas waktu rekapitulasi administrasi melalui otomatisasi pelaporan format Excel. Implikasi praktis dari sistem ini adalah terciptanya tata kelola aset laboratorium yang lebih transparan dan akuntabel, serta peningkatan keamanan data melalui mekanisme pembatasan hak akses yang sebelumnya tidak tersedia pada sistem lama. Meskipun implementasi sistem menuntut penyesuaian pengguna terhadap antarmuka baru, sistem ini terbukti stabil berdasarkan pengujian fungsional dan disarankan untuk dikembangkan lebih lanjut melalui integrasi notifikasi real-time (seperti WhatsApp Gateway) serta sinkronisasi dengan sistem inventaris universitas yang lebih luas guna memaksimalkan responsivitas layanan pemeliharaan.

Bibliografi

- Demir, G., Riaz, M., & Deveci, M. (2024). Wind farm site selection using geographic information system and fuzzy decision making model. *Expert Systems with Applications*, 255, 124772. <https://doi.org/10.1016/j.eswa.2024.124772>
- E. P. Nugroho et al., “Design Blockchain Architecture for Population Data Management to Realize a Smart City in Cimahi, West Java, Indonesia,” *J. Ilm. Tek. Elektro Komput. Dan Inform.*, vol. 9, no. 4, pp. 1206–1222, Jan. 2024, doi: 10.26555/jiteki.v9i4.27493.
- Hariguna, T., Muflikhatun, S., & Ramadani, N. C. (2022). Pelatihan Peningkatan Mutu Penulisan Karya Ilmiah di Universitas Amikom Purwokerto. *Jurnal Pengabdian Masyarakat*, 4(2).
- Kusnadi, Y., & Putra, D. W. (2024). E-Commerce Berbasis Website pada UMKM Menggunakan Framework Codeigniter 4 (Studi Kasus: Toko Wakuteka). *Jurnal Teknologi Informatika dan Komputer*, 10(1), 257–271. <https://doi.org/10.37012/jtik.v10i1.2101>
- Pradiasa, T. A. (2024). Perancangan Sistem Informasi Persediaan Barang (Inventory Application) Berbasis Web dan Bootstrap CSS. *Physical Sciences, Life Sciences and Engineering*, 1(2), 13. <https://doi.org/10.47134/pslse.v1i2.200>
- Przybyłek, A., Belter, D., & Conboy, K. (2025). A study of Scrum @ S&P Global in the post-COVID-19 era: Unsuitable for remote work or just flawed implementation? *Information and Software Technology*, 183, 107728. <https://doi.org/10.1016/j.infsof.2025.107728>
- R. Umar, B. Robiin, V. Y. Erviana, M. I. Taruna, and A. Kurniawan, “Analysis of Software Requirement Specification and Use Case Diagram of Metaverse Museum Muhammadiyah,” *Preserv. Digit. Technol. Cult.*, vol. 54, no. 2, pp. 125–133, July 2025, doi: 10.1515/pdte-2024-0062.
- Shojaei, P., Vlahu-Gjorgievska, E., & Chow, Y.-W. (2024). Security and Privacy of Technologies in Health Information Systems: A Systematic Literature Review. *Computers*, 13(2), 41. <https://doi.org/10.3390/computers13020041>
- Suci, R., Ismarmiaty, & Rismayati, R. (2024). Sistem Informasi Penerimaan Siswa Baru Berbasis Website SDN 3 Gerung Utara. *JATISKOM: Jurnal Aplikasi Teknologi Informasi dan Sains Komputer*, 1(2), 134–146. <https://doi.org/10.20414/jatiskom.v1i2.12689>