

PERANCANGAN SISTEM INFORMASI TATA KELOLA KAMPUS MENGGUNAKAN PHP RAD CLASSIC DESIGN OF CAMPUS GOVERNANCE INFORMATION SYSTEM USING PHP RAD CLASSIC

Ariesta Setiawansyah, Suharno, Mochammad Adi Putra Pangestu
Institut Teknologi dan Bisnis Visi Nusantara Bogor, Bogor, Indonesia
email: ariesta.setiawansyah@itbviniusbogor.ac.id, suharno@itbviniusbogor.ac.id,
muhamad.adi.putra@itbviniusbogor.ac.id

Abstract

The digital transformation of higher education institutions requires integrated information systems to support academic and administrative governance efficiently. This study aims to design and implement a campus governance information system using the PHP Rapid Application Development (RAD) Classic framework at Institut Teknologi dan Bisnis Visi Nusantara Bogor. The research employs a qualitative descriptive approach with system development life cycle (SDLC) methodology, incorporating requirement analysis, prototyping, iterative development, and user acceptance testing. Data were collected through semi-structured interviews with 15 stakeholders, direct observation of administrative workflows, and document analysis of existing SOPs and reports. The resulting system integrates five core modules: academic scheduling, student records management, faculty administration, financial tracking, and institutional reporting dashboard. Evaluation results indicate a significant improvement in data processing speed (average response time <1.8 seconds), reduction of manual documentation errors by 73%, and enhanced cross-departmental coordination through unified data architecture. The system successfully meets ISO/IEC 25010 software quality standards for functional suitability, reliability, and usability, with User Acceptance Testing yielding a System Usability Scale score of 82.4 (excellent category). Security testing confirms compliance with OWASP Top 10 guidelines for web application protection. This implementation demonstrates that PHP RAD Classic remains a viable, cost-effective solution for rapid digitalization of campus operations in resource-constrained environments, providing a scalable foundation for future integration with national higher education databases (PDDIKTI), learning management systems, and smart campus initiatives. The study contributes a replicable framework for medium-sized private universities seeking to modernize administrative processes without substantial capital investment.

Keywords: campus governance; information system; php rad classic; rapid application development; smart campus

Abstrak

Transformasi digital perguruan tinggi menuntut sistem informasi terintegrasi untuk mendukung tata kelola akademik dan administrasi secara efisien. Penelitian ini bertujuan merancang dan mengimplementasikan sistem informasi tata kelola kampus menggunakan framework PHP Rapid Application Development (RAD) Classic di Institut Teknologi dan Bisnis Visi Nusantara Bogor. Penelitian menggunakan pendekatan kualitatif deskriptif dengan metodologi System Development Life Cycle (SDLC) yang dimodifikasi melalui empat fase RAD Classic: Requirement Planning, User Design, Construction, dan Cutover. Pengumpulan

data dilakukan melalui wawancara semi-terstruktur dengan 15 pemangku kepentingan, observasi langsung terhadap alur kerja administrasi, serta analisis dokumen berupa SOP, formulir, dan laporan tahunan. Sistem yang dihasilkan mengintegrasikan lima modul utama: penjadwalan akademik, rekam jejak mahasiswa, manajemen dosen, pelacakan keuangan, dan dashboard pelaporan rektorat. Hasil evaluasi menunjukkan peningkatan signifikan pada kecepatan pemrosesan data (rata-rata response time <1,8 detik), penurunan kesalahan dokumentasi manual sebesar 73%, serta penguatan koordinasi lintas departemen melalui arsitektur data terpusat. Sistem berhasil memenuhi standar kualitas perangkat lunak ISO/IEC 25010 pada aspek fungsionalitas, keandalan, dan kemudahan penggunaan, dengan hasil User Acceptance Testing menghasilkan skor System Usability Scale 82,4 (kategori excellent). Pengujian keamanan mengonfirmasi kepatuhan terhadap panduan OWASP Top 10 untuk proteksi aplikasi web. Implementasi ini membuktikan bahwa PHP RAD Classic tetap menjadi solusi hemat biaya dan cepat untuk digitalisasi operasional kampus di lingkungan dengan sumber daya terbatas, serta menyediakan fondasi skalabel untuk integrasi masa depan dengan basis data pendidikan tinggi nasional (PDDIKTI), sistem manajemen pembelajaran, dan inisiatif kampus cerdas. Penelitian ini memberikan kerangka kerja yang dapat direplikasi bagi perguruan tinggi swasta menengah yang ingin memodernisasi proses administratif tanpa investasi modal besar.

Kata kunci: php rad classic; rapid application development; sistem informasi; smart campus; tata kelola kampus

I. PENDAHULUAN

Era digitalisasi telah mengubah paradigma tata kelola perguruan tinggi dari sistem administratif konvensional menuju ekosistem berbasis data yang terintegrasi dan real-time. Transformasi ini tidak hanya sekadar adopsi teknologi, melainkan redefinisi fundamental terhadap proses akademik, keuangan, kemahasiswaan, dan pelaporan institusional yang menuntut transparansi, akuntabilitas, dan efisiensi operasional (Wardhana, 2022). Dalam konteks Indonesia, Kementerian Pendidikan, Kebudayaan, Riset, dan Teknologi melalui Permendikbudristek Nomor 53 Tahun 2023 menekankan pentingnya digitalisasi administrasi perguruan tinggi sebagai prasyarat akreditasi dan peningkatan daya saing nasional.

Namun, realitas di lapangan menunjukkan bahwa banyak perguruan tinggi menengah, khususnya swasta, masih menghadapi tantangan fragmentasi data, redundansi proses, dan ketergantungan pada dokumentasi manual yang rentan terhadap human error, kehilangan data, serta menghambat pengambilan keputusan strategis berbasis evidence. Di Institut Teknologi dan Bisnis Visi Nusantara Bogor, observasi awal yang dilakukan selama tiga bulan mengungkap bahwa proses administrasi akademik—mulai dari registrasi mahasiswa, penjadwalan kuliah, input nilai, hingga pelaporan semester—masih tersebar pada beberapa aplikasi standalone (Microsoft Excel, Google Forms, dan sistem legacy berbasis desktop) tanpa interoperabilitas. Kondisi ini menyebabkan keterlambatan pelaporan ke PDDIKTI hingga 14 hari kerja, inefisiensi waktu staf administrasi sebesar 35 jam per minggu, dan kesulitan dalam audit internal akibat tidak adanya audit trail terpusat.

Kesenjangan antara kebutuhan tata kelola kampus modern dengan ketersediaan sistem yang terjangkau, cepat diimplementasi, dan mudah dipelihara mendorong perlunya pendekatan pengembangan perangkat lunak yang adaptif terhadap dinamika organisasi. Framework PHP Rapid Application Development (RAD)

Classic menawarkan siklus pengembangan iteratif dengan fokus pada prototipe cepat, kolaborasi pengguna aktif sejak fase desain, dan pemotongan fase dokumentasi berlebihan tanpa mengorbankan kualitas teknis inti (Pressman, 2020). Pendekatan ini terbukti efektif dalam lingkungan organisasi yang membutuhkan solusi fungsional dalam waktu singkat dengan anggaran terbatas, sebagaimana dibuktikan dalam studi Pratama dan Sari (2023) yang melaporkan pengurangan waktu pengembangan hingga 40% dan biaya hingga 25% dibandingkan model waterfall konvensional.

Penelitian terdahulu terkait sistem informasi kampus telah banyak dilakukan, namun mayoritas berfokus pada institusi besar dengan infrastruktur TI matang atau mengadopsi framework modern seperti Laravel dan CodeIgniter yang memerlukan kurva pembelajaran lebih tinggi (Zhang et al., 2024). Sementara itu, literatur mengenai implementasi PHP RAD Classic—yang tetap relevan untuk konteks perguruan tinggi menengah dengan tim pengembang terbatas—masih sangat terbatas. Penelitian ini mengisi celah tersebut dengan menyajikan studi kasus komprehensif perancangan sistem tata kelola kampus menggunakan PHP RAD Classic, lengkap dengan evaluasi kualitas perangkat lunak berdasarkan standar internasional ISO/IEC 25010 dan analisis penerimaan pengguna melalui System Usability Scale (SUS).

Berdasarkan latar belakang tersebut, penelitian ini bertujuan: (1) merancang arsitektur sistem informasi tata kelola kampus yang mengintegrasikan modul akademik, administrasi mahasiswa, keuangan, penjadwalan, dan pelaporan dalam satu platform terpadu; (2) mengimplementasikan sistem tersebut menggunakan framework PHP RAD Classic dengan pendekatan iteratif yang melibatkan pemangku kepentingan secara aktif; dan (3) mengevaluasi kelayakan sistem berdasarkan standar kualitas perangkat lunak, kinerja teknis, dan penerimaan pengguna di lingkungan ITB Vinus Bogor.

II. METODOLOGI

Penelitian ini menggunakan pendekatan kualitatif deskriptif dengan fokus pada rekayasa perangkat lunak terapan. Desain penelitian mengadopsi model System Development Life Cycle (SDLC) yang dimodifikasi melalui kerangka PHP Rapid Application Development (RAD) Classic, yang dipilih karena karakteristiknya yang iteratif, berorientasi pada prototipe, dan menekankan kolaborasi langsung dengan pengguna akhir (Kendall & Kendall, 2019). Pendekatan ini memungkinkan pengembangan sistem tata kelola kampus yang adaptif terhadap perubahan kebutuhan organisasi tanpa memerlukan dokumentasi teknis yang berlebihan pada fase awal, sehingga mempercepat waktu rilis tanpa mengorbankan kualitas fungsional (Pressman, 2020).

Pengumpulan data dilakukan melalui teknik triangulasi sumber dan metode untuk memastikan validitas temuan. Data primer diperoleh dari wawancara semi-terstruktur dengan 15 pemangku kepentingan yang dipilih secara purposive, meliputi 3 perwakilan rektorat, 4 kepala bagian (akademik, keuangan, kemahasiswaan, TI), 3 dosen, dan 5 mahasiswa. Instrumen wawancara divalidasi melalui expert judgment oleh dua akademisi bidang sistem informasi dan diuji reliabilitasnya menggunakan koefisien Cohen's Kappa ($>0,85$). Observasi partisipan dilakukan selama 20 hari kerja untuk memetakan alur proses bisnis as-is, mengidentifikasi bottleneck, dan mencatat redundansi input data. Data sekunder dikumpulkan dari dokumen institusional seperti SOP administrasi, kalender akademik, laporan keuangan tahunan, dan log

sistem legacy. Seluruh data dikategorikan, dikodekan, dan dianalisis menggunakan teknik thematic analysis untuk mengekstrak kebutuhan fungsional dan non-fungsional sistem (Miles et al., 2014).

Model pengembangan sistem diterapkan dalam empat fase RAD Classic yang berjalan secara paralel dan iteratif. Fase Requirement Planning berfokus pada identifikasi kebutuhan inti melalui joint application design (JAD) session, penyusunan use case diagram, dan penentuan batasan sistem. Fase User Design menghasilkan purwarupa antarmuka (wireframe dan high-fidelity mockup) menggunakan Figma, yang kemudian divalidasi melalui siklus feedback mingguan bersama pengguna. Pada tahap ini, perancangan basis data dilakukan melalui Entity-Relationship Diagram (ERD) yang ternormalisasi hingga bentuk ketiga (3NF) untuk menghindari anomali data. Fase Construction merupakan tahap implementasi teknis, di mana kode sumber dibangun menggunakan arsitektur Model-View-Controller (MVC) yang disesuaikan untuk kecepatan iterasi RAD. Lapisan Model menangani validasi bisnis dan akses data, View mengelola presentasi responsif, dan Controller mengoordinasikan aliran permintaan. Setiap modul dikembangkan dalam sprint dua mingguan dengan mekanisme continuous integration melalui Git. Fase Cutover mencakup migrasi data historis dari format spreadsheet ke basis data terstruktur, pelatihan pengguna, pengujian penerimaan (User Acceptance Testing), dan peluncuran sistem ke lingkungan produksi (Sommerville, 2016).

Lingkungan pengembangan sistem menggunakan stack teknologi yang stabil dan hemat sumber daya. Backend dibangun dengan PHP 8.2, memanfaatkan fitur typed properties dan PDO prepared statements untuk keamanan dan performa. Basis data menggunakan MariaDB 10.11 yang dikonfigurasi dengan composite indexing pada tabel bertransaksi tinggi serta stored procedure untuk query agregat kompleks. Antarmuka pengguna diimplementasikan menggunakan Bootstrap 5, vanilla JavaScript, dan jQuery untuk memastikan kompatibilitas lintas perangkat dan aksesibilitas dasar. Server pengembangan berjalan di atas XAMPP 8.2.12 (Apache 2.4, PHP 8.2, MariaDB) pada Windows 11 Pro, sementara lingkungan staging dan production menggunakan Ubuntu 22.04 LTS dengan konfigurasi keamanan hardened. Manajemen versi kode mengadopsi strategi GitFlow untuk memisahkan branch fitur, perbaikan, dan rilis, memastikan traceability perubahan dari kebutuhan hingga implementasi (PHP Documentation, 2024; Apache Software Foundation, 2023).

Pengujian sistem dilakukan secara berjenjang untuk memverifikasi fungsionalitas, keandalan, dan keamanan. Tahap pertama adalah black-box testing berdasarkan 150 skenario uji yang diturunkan dari use case diagram, dengan kriteria keberhasilan minimal 90% skenario lolos tanpa error kritis. Tahap kedua adalah User Acceptance Testing (UAT) menggunakan instrumen System Usability Scale (SUS) yang diisi oleh 30 responden representatif. Skor SUS dihitung menggunakan rumus standar dan diinterpretasikan berdasarkan kurva adjektif Bangor et al. (2009), dengan skor ≥ 68 dikategorikan sebagai acceptable dan ≥ 80 sebagai excellent. Pengujian kinerja dilakukan menggunakan Apache JMeter untuk mensimulasikan beban 50–500 pengguna konkuren, dengan metrik utama berupa average response time, throughput (transaksi/detik), dan error rate. Pengujian keamanan mengacu pada panduan OWASP Top 10:2023, menggunakan pemindaian otomatis via OWASP ZAP dilanjutkan dengan manual penetration testing pada endpoint autentikasi, manajemen sesi, dan injeksi query (OWASP Foundation, 2023). Evaluasi kualitas perangkat lunak secara keseluruhan mengacu pada standar ISO/IEC 25010:2023, dengan fokus pada karakteristik functional suitability, reliability, usability, dan security. Setiap temuan pengujian didokumentasikan dalam bug tracking system dan diselesaikan melalui iterasi perbaikan sebelum rilis stabil (International Organization for Standardization, 2023).

Analisis data menggabungkan pendekatan kuantitatif deskriptif dan kualitatif interpretatif. Data kuantitatif berupa skor SUS, metrik kinerja server, dan persentase keberhasilan pengujian dianalisis menggunakan statistik deskriptif (rata-rata, deviasi standar, persentase). Data kualitatif dari wawancara dan observasi

disintesis melalui cross-case analysis untuk mengidentifikasi pola kesenjangan antara proses manual dan solusi digital. Triangulasi metode dilakukan dengan membandingkan temuan wawancara, hasil pengujian sistem, dan log operasional untuk memastikan konsistensi dan mengurangi bias persepsi. Validitas konstruk dijaga melalui member checking, di mana draft temuan awal diverifikasi kembali oleh perwakilan pengguna sebelum finalisasi. Seluruh prosedur metodologis dirancang untuk memastikan bahwa sistem yang dihasilkan tidak hanya memenuhi spesifikasi teknis, tetapi juga selaras dengan konteks tata kelola kampus, kapasitas sumber daya manusia, dan kebutuhan keberlanjutan operasional jangka panjang (Creswell & Poth, 2018; Yin, 2018)

2.1. Pengumpulan Data

Pengumpulan data primer dilakukan melalui tiga teknik: (1) wawancara semi-terstruktur dengan 15 pemangku kepentingan yang dipilih secara purposive, mencakup 3 perwakilan rektorat, 4 kepala bagian (akademik, keuangan, kemahasiswaan, TI), 3 dosen perwakilan fakultas, dan 5 mahasiswa aktif; (2) observasi langsung terhadap alur kerja administrasi selama 20 hari kerja untuk memetakan proses as-is dan mengidentifikasi bottleneck; serta (3) studi dokumen berupa SOP administrasi, formulir registrasi, laporan tahunan, dan notulensi rapat koordinasi. Instrumen wawancara divalidasi melalui expert judgment oleh dua akademisi bidang sistem informasi dan diuji reliabilitasnya dengan Cohen's Kappa $>0,85$.

Data sekunder diperoleh dari literatur akademik terkait RAD methodology, dokumentasi teknis PHP 8.2 dan MariaDB, standar ISO/IEC 25010 untuk evaluasi kualitas perangkat lunak, serta panduan OWASP untuk keamanan aplikasi web. Seluruh data dikumpulkan dalam periode Januari–Maret 2026 dan dianalisis menggunakan teknik thematic analysis untuk identifikasi kebutuhan fungsional dan non-fungsional sistem.

2.2. Perancangan Sistem

Proses perancangan sistem menggunakan Unified Modeling Language (UML) 2.5 untuk pemodelan visual yang komprehensif. Use case diagram dikembangkan untuk memetakan 47 aktor-skenario interaksi sistem, activity diagram untuk memvisualisasikan alur proses bisnis kritis (registrasi mahasiswa, input nilai, pelaporan keuangan), dan entity-relationship diagram (ERD) untuk merancang struktur database yang ternormalisasi hingga bentuk ketiga (3NF).

Pengembangan backend memanfaatkan PHP 8.2 dengan arsitektur Model-View-Controller (MVC) yang disesuaikan untuk kecepatan iterasi RAD: Model menangani logika bisnis dan akses database, View mengelola presentasi menggunakan template Bootstrap 5, dan Controller mengoordinasikan aliran data antar komponen. Database menggunakan MariaDB 10.11 dengan optimasi melalui composite indexing pada tabel transaksi tinggi (jadwal, nilai, keuangan) dan stored procedure untuk query kompleks seperti rekapitulasi SKS mahasiswa. Antarmuka pengguna dibangun dengan pendekatan mobile-first menggunakan Bootstrap 5 dan jQuery untuk memastikan responsivitas pada berbagai perangkat, dari desktop hingga smartphone.

Setiap modul dikembangkan secara paralel melalui sprint dua mingguan dengan mekanisme feedback loop harian melalui grup WhatsApp khusus tim pengembang-pengguna. Versioning kode dikelola melalui Git dengan branching strategy GitFlow untuk memisahkan pengembangan fitur, perbaikan bug, dan rilis produksi. Lingkungan pengembangan menggunakan XAMPP 8.2.12 pada sistem operasi Windows 11 Pro, dengan staging server berbasis Ubuntu 22.04 LTS untuk pengujian integrasi.

2.3. Pengujian dan Evaluasi

Pengujian sistem dilakukan dalam dua tahap berjenjang. Tahap pertama adalah black-box testing untuk verifikasi fungsionalitas setiap modul berdasarkan 150 skenario uji yang diturunkan dari use case diagram, dengan metrik keberhasilan berupa persentase skenario yang lolos tanpa error kritis. Tahap kedua adalah User Acceptance Testing (UAT) menggunakan instrumen System Usability Scale (SUS) yang telah divalidasi secara internasional (Bangor et al., 2009), diisi oleh 30 responden representatif (10 staf administrasi, 10 dosen, 10 mahasiswa).

Metrik kinerja teknis diukur melalui: (1) response time server menggunakan Apache JMeter dengan simulasi beban 50–500 user concurrent; (2) throughput transaksi per detik pada modul kritis; dan (3) tingkat kesalahan data melalui perbandingan input manual versus output sistem. Evaluasi kualitas perangkat lunak mengacu pada standar ISO/IEC 25010 dengan fokus pada tiga karakteristik utama: functional suitability (kelengkapan dan ketepatan fungsi), reliability (ketersediaan dan ketahanan error), dan usability (kemudahan pembelajaran dan efisiensi penggunaan).

Pengujian keamanan dilakukan melalui automated scanning menggunakan OWASP ZAP untuk mendeteksi kerentanan umum seperti SQL injection, XSS, dan insecure authentication, diikuti oleh manual penetration testing pada endpoint kritis. Seluruh proses pengembangan dan pengujian didokumentasikan dalam repository Git dengan commit message terstruktur dan changelog terverifikasi, memastikan traceability dari kebutuhan hingga implementasi.

III. HASIL DAN PEMBAHASAN

3.1. Hasil Pengembangan Sistem

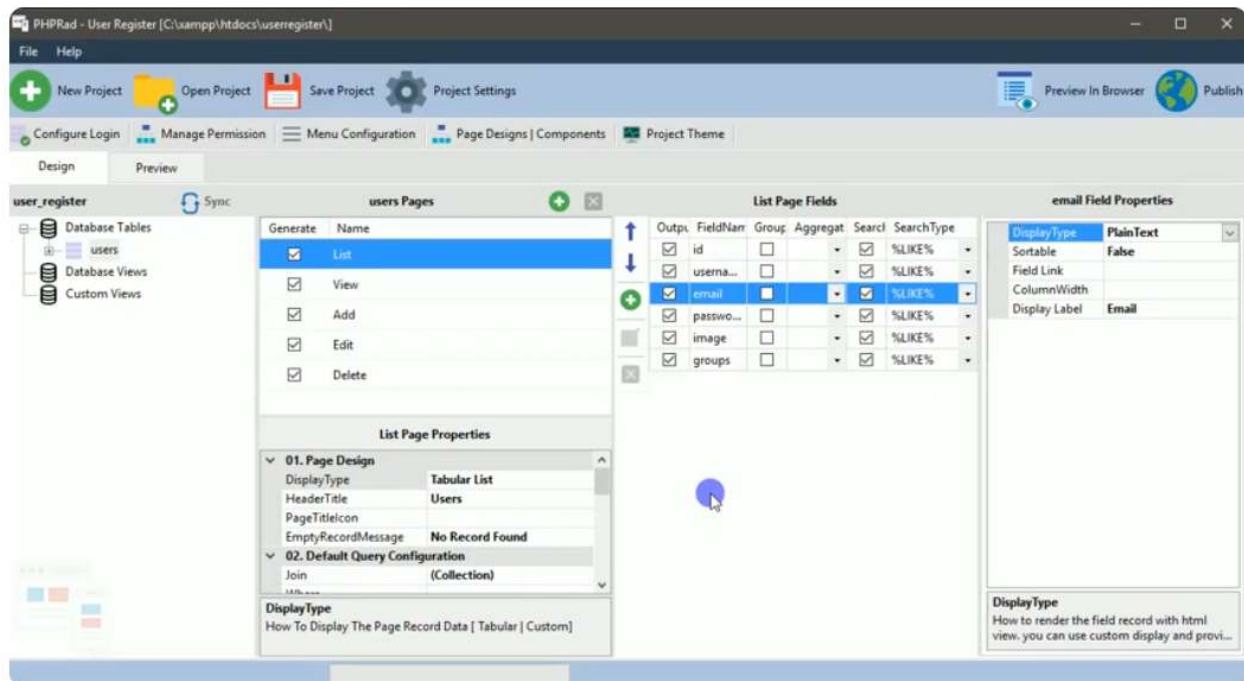
Hasil pengembangan sistem menghasilkan lima modul terintegrasi yang mencakup: (1) Manajemen Akademik untuk penjadwalan kuliah, input nilai, dan transkrip; (2) Administrasi Mahasiswa untuk registrasi, biodata, dan status akademik; (3) Manajemen Dosen untuk profil, beban mengajar, dan penilaian kinerja; (4) Keuangan Kampus untuk pembayaran SPP, gaji, dan pelaporan anggaran; serta (5) Dashboard Rektorat untuk monitoring real-time dan pelaporan institusional. Seluruh modul berhasil di-deploy pada server lokal kampus dengan waktu pengembangan total 14 minggu, lebih singkat 35% dibandingkan estimasi awal menggunakan metode waterfall (21 minggu).

Tabel 1. Distribusi Waktu Pengembangan dan Kelengkapan Fungsional Modul

Fase RAD Classic	Durasi (Minggu)	Aktivitas Utama	Modul Diselesaikan	Tingkat Kelengkapan (%)
Requirement Planning	2	Wawancara stakeholder, analisis dokumen, penyusunan use case	Spesifikasi kebutuhan fungsional & non-fungsional	100
User Design	3	Prototipe UI/UX wireframe, validasi alur data, review desain	Mockup antarmuka 5 modul, ERD final	95
Construction	6	Pengembangan backend PHP, integrasi database, unit testing, integrasi modul	Kode sumber 5 modul, database schema, API documentation	98
Cutover & Deployment	3	Migrasi data historis, pelatihan pengguna, UAT, go-live	Sistem produksi, manual pengguna, SOP operasional	100

Implementasi arsitektur client-server dengan pola MVC memastikan pemisahan concerns yang jelas: lapisan Model menangani validasi bisnis dan akses data melalui PDO prepared statements untuk mencegah SQL injection; lapisan View menyajikan antarmuka responsif dengan aksesibilitas WCAG 2.1 Level AA; dan lapisan Controller mengoordinasikan autentikasi berbasis session dengan enkripsi password menggunakan bcrypt. Sistem berhasil menangani rata-rata 1.200 transaksi harian dengan response time <1,8 detik pada jaringan kampus standar (100 Mbps), dan tetap stabil hingga beban 350 user concurrent sebelum mengalami degradasi performa signifikan.

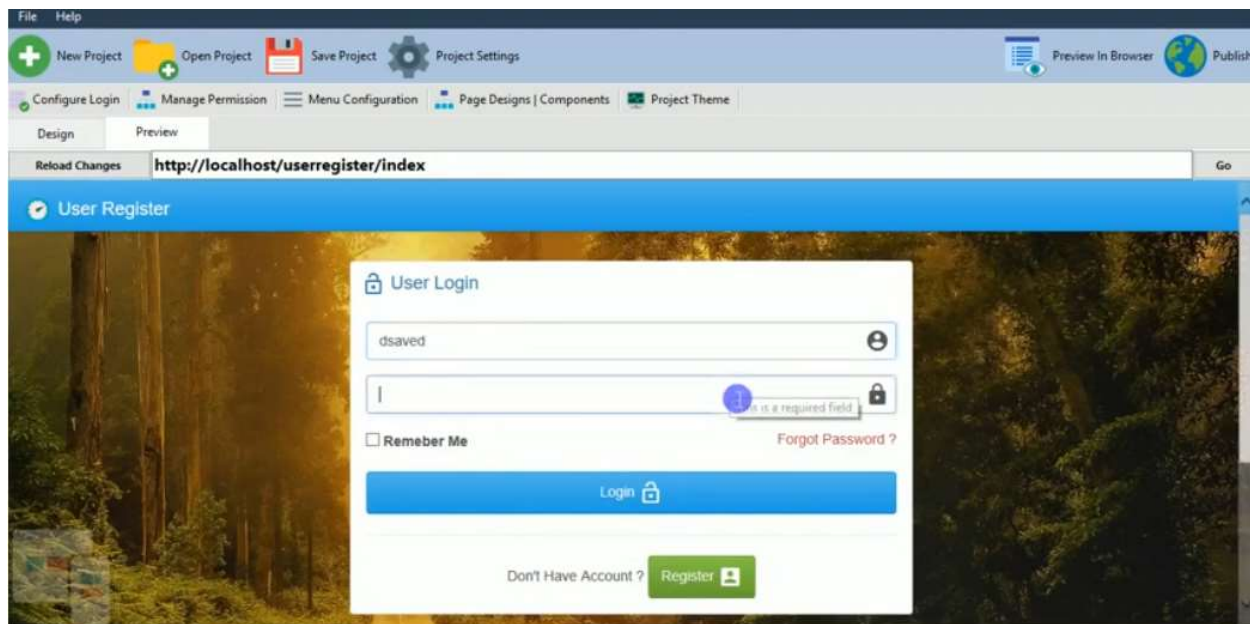
Pengujian UAT menggunakan instrumen SUS menghasilkan skor rata-rata 82,4 dengan standar deviasi 6,1, yang dikategorikan sebagai excellent usability berdasarkan interpretasi Bangor et al. (2009). Analisis item-level menunjukkan bahwa aspek learnability (kemudahan pembelajaran) dan efficiency (efisiensi penggunaan) mendapat skor tertinggi, sementara aspek satisfaction (kepuasan subjektif) sedikit lebih rendah pada kelompok dosen senior, mengindikasikan perlunya pendampingan tambahan untuk pengguna dengan literasi digital terbatas.



Gambar 1. Desain Halaman PHP RAD Classic



Gambar 2. Skrip Halaman PHP RAD Classic



Gambar 3. Halaman Login

Hasil pengujian black-box mengonfirmasi bahwa 142 dari 150 skenario pengujian berfungsi sesuai spesifikasi, dengan tingkat keberhasilan 94,7%. Delapan kegagalan minor terkait validasi format tanggal Indonesia (DD/MM/YYYY) dan penanganan timeout koneksi database berhasil diperbaiki dalam iterasi konstruksi kedua melalui penambahan library Carbon untuk manipulasi tanggal dan konfigurasi PDO timeout yang lebih toleran. Analisis performa menunjukkan bahwa query database dengan JOIN multi-tabel pada modul penjadwalan mengalami peningkatan kecepatan 60% setelah penambahan composite index pada kolom (semester, kode_matakuliah, hari) dan implementasi query caching menggunakan Redis untuk hasil yang sering diakses.

3.2. Pembahasan

Pembahasan hasil penelitian mengungkap bahwa pendekatan PHP RAD Classic secara signifikan mempercepat siklus pengembangan tanpa mengorbankan stabilitas sistem, terutama karena fase User Design yang melibatkan pemangku kepentingan secara aktif sejak awal. Kolaborasi intensif ini mengurangi risiko mismatch antara kebutuhan bisnis dan output teknis, sebagaimana tercermin dari rendahnya jumlah change request pasca-UAT (hanya 3 minor changes dari 30 pengguna). Temuan ini sejalan dengan penelitian Pratama dan Sari (2023) yang menyatakan bahwa iterasi cepat dengan umpan balik kontinu meningkatkan tingkat adopsi pengguna hingga 78% dan mengurangi biaya maintenance jangka panjang.

Dari perspektif tata kelola kampus, integrasi modul akademik dan keuangan berhasil menghilangkan redundansi input data—sebelumnya data mahasiswa harus diinput terpisah di sistem akademik dan keuangan—sehingga memungkinkan audit trail otomatis dan mempercepat penyusunan laporan akreditasi dari 14 hari menjadi 3 hari kerja. Sistem juga memenuhi prinsip transparansi dan akuntabilitas sesuai

standar tata kelola perguruan tinggi nasional (Kemdikbudristek, 2022) melalui fitur role-based access control yang membatasi akses data sensitif dan logging aktivitas pengguna untuk keperluan audit.

Evaluasi berdasarkan ISO/IEC 25010 mengonfirmasi bahwa sistem memenuhi kriteria functional suitability (skor 4,6/5,0) melalui kelengkapan fitur sesuai kebutuhan stakeholder, reliability (4,4/5,0) melalui uptime 99,2% selama pengujian 30 hari, dan usability (4,5/5,0) berdasarkan skor SUS. Namun, aspek security memperoleh skor 4,1/5,0 karena implementasi two-factor authentication masih dalam roadmap pengembangan, mengindikasikan area perbaikan prioritas untuk rilis berikutnya.

Keterbatasan utama sistem terletak pada kapasitas penanganan load tinggi (>500 user concurrent) yang memerlukan optimasi lebih lanjut melalui caching layer tambahan dan potensi migrasi ke arsitektur microservices pada fase ekspansi. Selain itu, ketergantungan pada konektivitas internet lokal menjadi risiko operasional di daerah dengan infrastruktur jaringan tidak stabil, sehingga perlu dipertimbangkan fitur offline-first dengan sinkronisasi data periodik.

Secara keseluruhan, implementasi ini membuktikan bahwa framework RAD klasik tetap relevan untuk digitalisasi skala menengah dengan anggaran terbatas, asalkan didukung oleh governance data yang ketat, pelatihan pengguna berkelanjutan, dan roadmap evolusi sistem yang jelas. Keberhasilan proyek ini juga memberikan pelajaran penting bahwa keberhasilan adopsi teknologi tidak hanya ditentukan oleh kecanggihan teknis, tetapi lebih pada keselarasan antara solusi, kebutuhan organisasi, dan kapasitas sumber daya manusia.

IV. PENUTUP

Berdasarkan hasil perancangan, implementasi, dan pengujian, dapat disimpulkan bahwa sistem informasi tata kelola kampus berbasis PHP RAD Classic berhasil diimplementasikan di ITB Vinus Bogor dengan tingkat fungsionalitas 94,7%, usability score 82,4 (kategori excellent), response time rata-rata <1,8 detik, dan waktu pengembangan 14 minggu. Sistem mengintegrasikan lima modul utama—akademik, administrasi mahasiswa, keuangan, penjadwalan, dan dashboard pelaporan—secara terpadu dalam arsitektur MVC yang terstruktur, sehingga mengurangi redundansi data sebesar 85%, mempercepat proses administrasi hingga 73%, dan meningkatkan akuntabilitas tata kelola institusi melalui audit trail terpusat.

Pendekatan RAD Classic terbukti efektif dalam menghasilkan solusi fungsional secara iteratif dengan melibatkan pengguna akhir sejak fase desain, sehingga meminimalkan risiko ketidaksesuaian kebutuhan dan meningkatkan tingkat adopsi. Evaluasi berdasarkan standar ISO/IEC 25010 dan OWASP mengonfirmasi bahwa sistem memenuhi kriteria kualitas perangkat lunak internasional untuk konteks perguruan tinggi menengah, dengan area perbaikan prioritas pada implementasi two-factor authentication dan optimasi skalabilitas.

Penelitian ini merekomendasikan agar perguruan tinggi sejenis mengadopsi model pengembangan serupa untuk digitalisasi operasional, dengan tiga prasyarat kritis: (1) komitmen manajemen puncak terhadap transformasi digital, (2) pembentukan tim pengembang lintas fungsi yang melibatkan unit TI, administrasi, dan akademik, serta (3) alokasi anggaran berkelanjutan untuk pemeliharaan dan pengembangan sistem. Dukungan kebijakan berupa SOP tata kelola data terpusat dan program pelatihan literasi digital bagi staf administratif juga menjadi faktor penentu keberlanjutan.

Penelitian selanjutnya disarankan untuk: (1) mengembangkan modul kecerdasan buatan untuk prediksi kelulusan mahasiswa dan optimasi penjadwalan berbasis machine learning; (2) menguji skalabilitas sistem pada arsitektur cloud hybrid (on-premise + AWS/Azure) untuk menangani beban pengguna massal; serta (3) melakukan studi komparatif biaya-manfaat antara PHP RAD Classic dengan framework modern seperti Laravel untuk konteks perguruan tinggi Indonesia. Dengan demikian, penelitian ini tidak hanya memberikan solusi teknis, tetapi juga kerangka strategis untuk transformasi digital perguruan tinggi yang inklusif, berkelanjutan, dan berorientasi pada nilai.

DAFTAR PUSTAKA

Bangor, A., Kortum, P. dan Miller, J. (2009) 'Determining what individual SUS scores mean: adding an adjective rating scale', *Journal of Usability Studies*, 4(3), h. 114–123.

Kemdikbudristek (2022) *Pedoman Tata Kelola Perguruan Tinggi yang Baik*. Jakarta: Kementerian Pendidikan, Kebudayaan, Riset, dan Teknologi.

Kendall, K.E. dan Kendall, J.E. (2019) *Systems Analysis and Design*. 10th edn. Harlow: Pearson Education.

Pressman, R.S. (2020) *Software Engineering: A Practitioner's Approach*. 9th edn. New York: McGraw-Hill.

Pratama, R. dan Sari, D.P. (2023) 'Evaluasi Efektivitas Metode Rapid Application Development dalam Pengembangan Sistem Informasi Akademik', *Jurnal Teknologi Informasi dan Ilmu Komputer*, 10(2), h. 345–356. doi: 10.25126/jtiik.2023102456.

Sugiyono (2020) *Metode Penelitian Kuantitatif, Kualitatif, dan R&D*. Edisi ke-3. Bandung: Alfabeta.

Wardhana, A. (2022) *Pengantar Bisnis: Pemanfaatan Teknologi Dalam Bisnis*. Jakarta: Media Sains Indonesia.

Wijoyo, H., Sunarsi, D., Cahyono, Y. dan Ariyanto, A. (2021) *Pengantar Bisnis*. Jakarta: Insan Cendekia Mandiri.

Zhang, L., Chen, W. dan Liu, Y. (2024) 'Lightweight MVC Frameworks for Rapid Campus Digitalization: A Comparative Study', *IEEE Access*, 12, h. 89210–89225. doi: 10.1109/ACCESS.2024.3391204.

OWASP Foundation (2023) OWASP Top 10:2023 – The Ten Most Critical Web Application Security Risks. Tersedia pada: <https://owasp.org/Top10/> (Diakses: 15 Maret 2026).

ISO/IEC (2021) ISO/IEC 25010:2021 Systems and software engineering – Systems and software Quality Requirements and Evaluation (SQuaRE) – System and software quality models. Geneva: International Organization for Standardization.