

# Blueprint

## Strategis Pengembangan Sistem Informasi Penjaminan Mutu Internal (E-SPMI) Enterprise: Adaptasi Arsitektur Modular Monolith dalam Ekosistem Pendidikan Tinggi Indonesia

### 1. Pendahuluan: Reorientasi Strategis Penjaminan Mutu di Era Digital

Transformasi pendidikan tinggi di Indonesia sedang mengalami pergeseran paradigma yang fundamental, bergerak dari kepatuhan administratif semata menuju budaya mutu yang substantif dan berkelanjutan. Momentum ini dipertegas dengan diterbitkannya **Peraturan Menteri Pendidikan Tinggi, Sains, dan Teknologi Republik Indonesia Nomor 39 Tahun 2025 tentang Penjaminan Mutu Pendidikan Tinggi**.<sup>1</sup> Regulasi ini tidak hanya menggantikan aturan sebelumnya tetapi juga meredefinisi status akreditasi menjadi Terakreditasi, Terakreditasi Unggul, dan Tidak Terakreditasi, yang menuntut perguruan tinggi untuk memiliki mekanisme deteksi dini dan pengendalian mutu yang jauh lebih responsif dan terintegrasi dibandingkan sebelumnya.<sup>2</sup>

Dalam konteks arsitektur perangkat lunak, tantangan ini bukan sekadar masalah digitalisasi formulir kertas menjadi format elektronik. Ini adalah tantangan rekayasa sistem yang kompleks yang melibatkan orkestrasi proses bisnis siklus PPEPP (Penetapan, Pelaksanaan, Evaluasi, Pengendalian, dan Peningkatan), integrasi data heterogen dari berbagai silo (SIAKAD, Keuangan, HRIS), serta pemenuhan standar keamanan Sistem Pemerintahan Berbasis Elektronik (SPBE) yang ketat.<sup>3</sup> Sistem SPMI modern harus berfungsi sebagai *nerve center* atau pusat syaraf universitas, yang mampu menyajikan data *real-time* mengenai kesehatan institusi berdasarkan 8 Indikator Kinerja Utama (IKU) dan standar melampaui SN-Dikti.<sup>5</sup>

Laporan ini menyajikan analisis mendalam dan blueprint teknis pengembangan aplikasi E-SPMI berbasis arsitektur **Modular Monolith**. Pendekatan ini dipilih karena menawarkan keseimbangan optimal antara modularitas fungsional—yang memungkinkan adaptasi cepat terhadap perubahan regulasi—with kesederhanaan operasional yang meminimalkan beban infrastruktur, sebuah pertimbangan krusial bagi mayoritas perguruan tinggi di Indonesia.<sup>7</sup> Dokumen ini dirancang bagi para pengambil keputusan teknis dan strategis, memberikan panduan komprehensif mulai dari analisis kebutuhan regulasi, desain basis data, hingga strategi *deployment* yang aman dan skalabel.

---

## 2. Analisis Kebutuhan Komprehensif dan Kepatuhan Regulasi

Pengembangan perangkat lunak di sektor pendidikan tinggi terikat erat pada *regulatory compliance*. Analisis kebutuhan tidak dapat dilakukan hanya dengan melihat kebutuhan pengguna saat ini (*current state*), tetapi harus memproyeksikan kebutuhan regulasi masa depan (*future state*).

### 2.1 Dekonstruksi Permendiktisaintek No. 39 Tahun 2025 dan Implikasi Sistem

Regulasi terbaru membawa implikasi teknis yang signifikan terhadap *business logic* aplikasi. Sistem tidak bisa lagi bersifat statis; ia harus memiliki *rule engine* yang dinamis.

Aspek Regulasi	Implikasi pada Arsitektur Sistem	Referensi
<b>Status Akreditasi Baru</b> (Unggul vs Terakreditasi)	Sistem membutuhkan algoritma simulasi skor (Prediktif) yang mampu menghitung posisi prodi secara <i>real-time</i> berdasarkan data yang masuk, memberikan peringatan dini jika status "Unggul" terancam.	2
<b>Siklus PPEPP</b>	<i>Workflow engine</i> harus bersifat <i>sequential strict enforcement</i> . Tahap Evaluasi (Audit) tidak boleh dibuka jika tahap Pelaksanaan (Input Bukti) belum dikunci. Sistem harus mencatat <i>audit trail</i> pada setiap transisi fase.	3
<b>Standar Melampaui SN-Dikti</b>	Struktur database standar harus bersifat polimorfik atau hirarkis tak terbatas	5

	( <i>n-level hierarchy</i> ), memungkinkan universitas menambahkan standar turunan (misal: Standar AI, Standar Green Campus) tanpa mengubah skema database.	
<b>Rekognisi Internasional</b>	Modul data harus mendukung <i>multilingual support</i> (i18n) dan kompatibilitas format data dengan standar akreditasi internasional (seperti AUN-QA atau ABET).	2
<b>Integrasi PDDikti</b>	Diperlukan <i>middleware</i> atau <i>service worker</i> yang berjalan di latar belakang untuk melakukan sinkronisasi dua arah dengan PDDikti Feeder, memastikan data internal konsisten dengan data di kementerian.	9

## 2.2 Analisis Kebutuhan Fungsional Berbasis Aktor (Actor-Goal Analysis)

Menggunakan pendekatan *User-Centered Design*, kita memetakan kebutuhan fungsional berdasarkan peran aktor dalam ekosistem kampus.

### 2.2.1 Lembaga Penjaminan Mutu (LPM) - The Administrator

Sebagai pemilik proses bisnis, LPM membutuhkan kendali penuh atas konfigurasi sistem. Kebutuhan kritis mereka meliputi manajemen periode audit yang fleksibel (bisa per semester atau per tahun) dan kemampuan untuk melakukan *cloning* standar dari tahun sebelumnya untuk efisiensi. LPM juga memerlukan fitur "Matrix Auditor" untuk mengelola penugasan auditor silang antar fakultas guna menghindari konflik kepentingan.<sup>13</sup>

### 2.2.2 Auditor Internal - The Evaluator

Auditor sering kali adalah dosen yang memiliki beban kerja tinggi. Sistem harus meminimalisir

beban kognitif mereka. Fitur *Desk Evaluation* harus menyajikan dokumen bukti dan instrumen penilaian dalam satu layar (*split view*) untuk mempercepat verifikasi. Kebutuhan krusial lainnya adalah fitur *Auto-Save* dan kemampuan bekerja dalam mode *offline-first* (sinkronisasi saat online) mengingat konektivitas di beberapa area kampus mungkin tidak stabil saat visitasi lapangan.<sup>15</sup>

### 2.2.3 Auditee (Kaprodi/Upps) - The Executor

Pain point utama auditee adalah input data yang berulang. Arsitektur sistem harus mengadopsi prinsip *One-Click Evidence*. Jika data jumlah mahasiswa sudah ada di SIAKAD, auditee tidak boleh diminta menginput angka tersebut secara manual. Sistem harus menyajikannya secara otomatis, dan auditee hanya perlu memvalidasi. Fitur *Repository* dokumen juga harus mendukung *versioning*, sehingga auditee bisa melacak revisi dokumen kurikulum dari tahun ke tahun.<sup>17</sup>

### 2.2.4 Pimpinan Universitas (Rektor/Dekan) - The Strategist

Pimpinan membutuhkan *helicopter view*. Mereka tidak tertarik pada detail checklist audit, melainkan pada tren dan risiko. Dashboard eksekutif harus menyajikan visualisasi data 8 IKU, peta risiko persebaran temuan audit (misalnya: "Fakultas Teknik memiliki kepatuhan rendah pada standar penelitian"), dan rekomendasi tindak lanjut berbasis AI/Analitik Sederhana.<sup>6</sup>

## 2.3 Analisis Kebutuhan Non-Fungsional (NFR)

Kebutuhan non-fungsional sering kali menjadi penentu kegagalan atau keberhasilan sistem dalam jangka panjang.

1. **Scalability:** Sistem harus mampu menangani lonjakan trafik (*spike*) pada minggu-minggu menjelang penutupan periode audit, di mana ribuan dosen mungkin mengakses sistem secara bersamaan untuk mengunggah bukti.
2. **Interoperability:** Sistem harus mengekspos API (RESTful/GraphQL) yang aman untuk berkomunikasi dengan sistem lain (HRIS, SIAKAD, Keuangan) tanpa *tight coupling*.<sup>20</sup>
3. **Security (SPBE Compliance):** Sesuai standar SPBE, sistem harus menerapkan enkripsi data *at-rest* dan *in-transit*, serta mendukung *Single Sign-On* (SSO) universitas untuk kenyamanan dan keamanan akses.<sup>4</sup>
4. **Auditability:** Karena ini adalah sistem audit, maka sistem itu sendiri harus diaudit. Setiap aksi (Create, Read, Update, Delete) harus tercatat dalam *Log Activity* yang tidak bisa dimanipulasi (*immutable logs*).<sup>23</sup>

---

## 3. Arsitektur Sistem: Pendekatan Modular Monolith

Dalam merancang sistem berskala enterprise untuk universitas, arsitek dihadapkan pada dikotomi klasik: Monolith vs Microservices. Microservices menawarkan isolasi sempurna namun membawa kompleksitas infrastruktur yang masif (*distributed tracing, network latency*,

*data consistency*). Untuk konteks mayoritas universitas di Indonesia, di mana tim IT sering kali terbatas, **Modular Monolith** adalah solusi arsitektural yang paling pragmatis dan *robust*.<sup>7</sup>

### 3.1 Rasionalisasi Arsitektur Modular Monolith

Modular Monolith menstrukturkan aplikasi sebagai satu unit *deployment* tunggal (sebuah artefak.jar atau folder aplikasi PHP), namun secara internal kode diorganisir secara ketat ke dalam modul-modul bisnis yang terisolasi.

- **Keunggulan Operasional:** Tim IT universitas hanya perlu memelihara satu *pipeline* CI/CD dan satu set server database, mengurangi *overhead* DevOps secara drastis dibandingkan mengelola puluhan container microservices.<sup>26</sup>
- **Refactoring Safety:** Karena kode berada dalam satu repositori (*monorepo*), perubahan pada kontrak API internal lebih mudah dikelola dan di-*refactor*.
- **Microservices-Ready:** Jika di masa depan modul tertentu (misal: Modul IKU yang berat komputasinya) perlu di-*scale* secara independen, arsitektur modular memudahkan ekstraksi modul tersebut menjadi service terpisah tanpa menulis ulang kode dari nol.<sup>8</sup>

### 3.2 Diagram Kontainer dan Batasan Modul

Sistem E-SPMI akan dibagi menjadi beberapa modul vertikal yang merepresentasikan domain bisnis SPMI:

1. **Modul Core & Identity:** Menangani otentikasi (SSO), manajemen pengguna, peran (RBAC), dan struktur organisasi universitas.
2. **Modul Standar (The Law):** Mengelola master data standar, indikator, dan pemetaan dokumen. Modul ini bersifat "read-heavy".
3. **Modul Pelaksanaan (The Doer):** Menangani input bukti kinerja, integrasi data transaksional dari SIAKAD, dan logbook kegiatan dosen.
4. **Modul Audit (The Checker):** Menangani logika penjadwalan audit, instrumen checklist, input temuan, dan kalkulasi skor.
5. **Modul Pengendalian (The Fixer):** Manajemen tiket Permintaan Tindakan Koreksi (PTK) dan Rapat Tinjauan Manajemen (RTM).
6. **Modul Analytics (The Insight):** Agregasi data untuk dashboard IKU dan pelaporan eksekutif.

### 3.3 Stack Teknologi Pilihan

Mempertimbangkan ketersediaan talenta (*talent pool*) di Indonesia dan karakteristik performa yang dibutuhkan, stack teknologi berikut direkomendasikan:

- **Backend Framework: Laravel 11 (PHP 8.3).** Laravel dipilih karena ekosistemnya yang sangat matang di Indonesia, fitur *Eloquent ORM* yang ekspresif untuk relasi data kompleks, dan dukungan bawaan untuk *Queue* dan *Job* yang kuat.<sup>28</sup> Alternatif untuk skala sangat besar adalah **Spring Boot (Java)**, namun *development velocity* Laravel biasanya lebih tinggi untuk tim kecil-menengah.<sup>31</sup>

- **Database: PostgreSQL 16.** Dipilih karena dukungannya yang superior terhadap tipe data JSONB (berguna untuk menyimpan formulir audit yang strukturnya dinamis) dan fitur *Full-Text Search* untuk pencarian dokumen.<sup>29</sup>
  - **Frontend: Vue.js 3** atau **React** dengan arsitektur *Inertia.js* (untuk Laravel). Ini memungkinkan pembangunan *Single Page Application* (SPA) yang responsif tanpa kompleksitas pemisahan repo frontend-backend yang berlebihan.
  - **Caching & Queue: Redis.** Wajib digunakan untuk *caching* konfigurasi standar dan menangani antrian proses berat seperti *report generation* atau sinkronisasi data PDDikti.
  - **Infrastructure: Docker** untuk kontainerisasi aplikasi, dijalankan di atas orkestrator sederhana (Docker Swarm) atau Kubernetes jika universitas sudah memilikinya.
- 

## 4. Blueprint Fungsional dan Spesifikasi Fitur Mendetail

Bagian ini menguraikan spesifikasi fungsional sistem E-SPMI yang dipetakan langsung ke siklus PPEPP, memastikan setiap tahapan penjaminan mutu terfasilitasi secara digital.

### 4.1 Modul Penetapan Standar (P) - Fondasi Sistem

Modul ini berfungsi sebagai "Konstitusi" mutu universitas. Fleksibilitas adalah kunci di sini.

- **Manajemen Standar Hirarkis (N-Level):**
  - Sistem harus mendukung struktur pohon: Standar Utama (misal: Pendidikan) -> Standar Turunan (misal: Proses Pembelajaran) -> Indikator -> Sub-Indikator.
  - Setiap *node* standar harus memiliki atribut metadata: Kode Standar, Deskripsi, Referensi Regulasi (SN-Dikti/Renstra), dan Status (Aktif/Non-aktif/Draft).<sup>5</sup>
- **Indikator Kinerja & Target Dinamis:**
  - Administrator dapat mendefinisikan tipe data indikator: Kuantitatif (Angka), Kualitatif (Teks), Boolean (Ya/Tidak), atau Pilihan Ganda.
  - Target indikator dapat diset berbeda per jenjang (D3, S1, S2) atau per rumpun ilmu (Saintek/Soshum). Ini mengakomodasi keragaman prodi dalam satu universitas.
- **Versioning & Cloning:**
  - Fitur "Clone Periode" memungkinkan LPM menyalin seluruh set standar tahun 2024 ke tahun 2025 dengan satu klik, kemudian hanya mengedit bagian yang berubah. Ini menghemat ratusan jam kerja administratif.
  - Sistem menyimpan *snapshot* standar per periode audit. Perubahan standar di tengah periode audit yang sedang berjalan harus dicegah atau dikunci versinya.

### 4.2 Modul Pelaksanaan & Monitoring (P) - Integrasi & Bukti

Tujuan modul ini adalah mengurangi beban administrasi dosen melalui integrasi data cerdas.

- **Repository Bukti Terpusat:**
  - Mekanisme upload dokumen dengan *tagging* otomatis. Dokumen yang diupload sekali (misal: Dokumen Kurikulum) dapat ditautkan ke berbagai standar berbeda

- tanpa perlu upload ulang (*Many-to-Many Relationship*).
- Dukungan *preview* dokumen langsung di browser (PDF Viewer) untuk memudahkan auditor dan auditee.<sup>32</sup>
- **Integrasi Data Transaksional (SIAKAD/HRIS Bridge):**
  - Sistem secara periodik (via *Cron Job*) menarik data dari SIAKAD untuk mengisi capaian indikator kuantitatif otomatis. Contoh: Indikator "Rata-rata kehadiran dosen" otomatis terisi angka "15.4 pertemuan" dari data presensi SIAKAD.
  - Jika data tidak sesuai, auditee dapat mengajukan "Sanggah Data" dengan melampirkan bukti manual, yang kemudian harus diverifikasi oleh admin.<sup>17</sup>
- **Evaluasi Diri (Self-Assessment):**
  - Sebelum audit visitasi, prodi wajib mengisi Evaluasi Diri. Sistem menghitung skor pra-audit untuk memberikan gambaran kesiapan prodi.<sup>14</sup>

## 4.3 Modul Audit Mutu Internal (E) - Inti Proses Bisnis

Modul ini mengelola interaksi antara Auditor dan Auditee, meniru proses visitasi namun dalam bentuk digital.

- **Perencanaan & Penjadwalan Audit:**
  - Algoritma penugasan auditor yang mencegah konflik kepentingan (misal: Dosen Teknik tidak boleh mengaudit Prodi Teknik).
  - Fitur notifikasi jadwal audit ke kalender akademik digital dosen.
- **Kertas Kerja Audit Digital (Digital Working Paper):**
  - Tampilan antarmuka *split-screen*: Kiri menampilkan Pernyataan Standar & Bukti Prodi, Kanan menampilkan Form Penilaian Auditor.
  - Auditor memberikan status temuan: **Sesuai, Observasi (OB), Ketidaksesuaian Minor, atau Ketidaksesuaian Mayor.**<sup>3</sup>
  - Fitur "Catatan Auditor" yang wajib diisi jika status adalah OB atau KTS, memaksa auditor memberikan alasan yang jelas.
- **Kalkulasi Skor & Grading:**
  - Sistem otomatis menghitung skor kepatuhan berdasarkan bobot standar.
  - Simulasi predikat akreditasi (Baik, Baik Sekali, Unggul) berdasarkan algoritma matriks penilaian BAN-PT atau LAM yang diinjeksikan ke dalam sistem.<sup>15</sup>

## 4.4 Modul Pengendalian & Peningkatan (P & P) - Siklus Berkelanjutan

Menjamin bahwa temuan audit ditindaklanjuti dan menjadi dasar peningkatan standar.

- **Manajemen PTK (Permintaan Tindakan Koreksi):**
  - Setiap temuan KTS secara otomatis men-generate "Tiket PTK" yang memiliki *due date*.
  - Auditee menginput *Root Cause Analysis* (Akar Masalah) dan Rencana Perbaikan.
  - Auditor memverifikasi bukti perbaikan secara digital untuk menutup (close) tiket PTK.
- **Rapat Tinjauan Manajemen (RTM) Online:**

- Modul untuk mencatat agenda RTM, daftar hadir, dan *Action Plan* tingkat universitas.
  - Hasil RTM (misal: keputusan menaikkan anggaran penelitian) ditautkan kembali ke Modul Standar untuk revisi target tahun depan.<sup>16</sup>
  - **Dashboard Business Intelligence:**
    - Visualisasi 8 IKU: Persentase lulusan bekerja, jumlah penelitian, kerjasama mitra kelas dunia, dll.<sup>6</sup>
    - Analisis Tren Mutu: Grafik garis yang menunjukkan kenaikan/penurunan skor mutu prodi selama 5 tahun terakhir.
- 

## 5. Desain Basis Data (Database Schema) & Manajemen Data

Kualitas sistem E-SPMI sangat bergantung pada struktur datanya. Skema database harus didesain untuk integritas referensial yang kuat namun fleksibel untuk perubahan item standar.

### 5.1 Skema Relasional Utama (Entity Relationship)

Berikut adalah narasi desain tabel-tabel inti dalam sistem, disusun dengan prinsip normalisasi<sup>33</sup>:

1. **Grup Master Standar:**
  - mst\_instruments: Tabel *header* yang menyimpan versi instrumen audit (misal: "Instrumen AMI 2025").
  - mst\_metrics: Tabel hirarkis (menggunakan pola *Adjacency List* dengan kolom *parent\_id*) yang menyimpan seluruh butir standar, indikator, dan sub-indikator. Kolom *data\_type* menentukan apakah input berupa angka, teks, atau file.
2. **Grup Organisasi & User (RBAC):**
  - ref\_units: Menyimpan data Fakultas, Prodi, Biro, dan Lembaga.
  - users: Data autentikasi pengguna.
  - model\_has\_roles: Tabel pivot (many-to-many) yang menghubungkan user dengan role (Auditor, Auditee, Admin) dalam konteks unit tertentu.
3. **Grup Transaksi Audit:**
  - trx\_audits: Tabel utama transaksi audit yang menghubungkan ref\_units (Auditee), mst\_instruments (Standar), dan Periode Audit.
  - trx\_audit\_details: Menyimpan nilai penilaian per butir indikator. Relasi 1-to-many ke mst\_metrics.
  - trx\_evidences: Menyimpan path file bukti atau link URL yang diupload auditee untuk setiap detil audit.
  - trx\_findings: Tabel khusus untuk menyimpan temuan (KTS/OB), analisis akar masalah, dan status perbaikan (Open/Closed).

## 5.2 Strategi Integrasi Data & API (SIAKAD/HRIS)

Integrasi data sering menjadi titik kegagalan sistem. Strategi **Anti-Corruption Layer (ACL)** diterapkan untuk melindungi inti sistem E-SPMI dari ketidakkonsistenan data sistem eksternal.<sup>20</sup>

- **Pola Integrasi:** Menggunakan REST API untuk komunikasi *real-time* (misal: cek login user) dan mekanisme *ETL* (*Extract, Transform, Load*) terjadwal via Background Jobs untuk sinkronisasi data besar (misal: data IPK seluruh mahasiswa).
- **Data Warehouse Mini:** E-SPMI sebaiknya memiliki tabel *staging* atau *data warehouse* kecil untuk menyimpan agregasi data dari SIAKAD (misal: tabel *dw\_student\_performance*). Ini mencegah E-SPMI membebani database operasional SIAKAD dengan query berat saat audit berlangsung.<sup>24</sup>

---

## 6. Rencana Kerja dan Implementasi (Implementation Roadmap)

Implementasi E-SPMI adalah proyek *marathon*, bukan *sprint*. Rencana kerja berikut dirancang untuk durasi 9 bulan, dengan pendekatan *Iterative & Incremental* untuk memitigasi risiko kegagalan adopsi.

Fase	Durasi	Aktivitas Utama & Output	Referensi
I. Inisiasi & Analisis	Bulan 1	Pembentukan Tim Task Force, Studi Regulasi Permendiktisaintek 39/2025, Analisis Kesenjangan Proses ( <i>Gap Analysis</i> ). <b>Output:</b> Dokumen SRS (Software Requirement Specs) & Project Charter.	39
II. Desain	Bulan 2	Perancangan ERD Database, Desain	35

<b>Arsitektur</b>		UI/UX (Prototyping), Penentuan Kontrak API dengan Tim SIAKAD. <b>Output:</b> Blueprint Teknis, Mockup UI High-Fidelity.	
<b>III. Pengembangan Core</b>	Bulan 3-4	Coding Modul Master Standar, Modul Manajemen User (SSO), dan Modul Repository Dokumen. Fokus pada fondasi data. <b>Output:</b> Versi Alpha (Internal Testing).	8
<b>IV. Pengembangan Audit</b>	Bulan 5-6	Coding Modul Transaksi Audit (Desk Eval & Visitasi), Kalkulasi Skor, dan Integrasi Data SIAKAD. <b>Output:</b> Versi Beta (Siap Pilot).	15
<b>V. Pilot &amp; UAT</b>	Bulan 7	<i>Conference Room Pilot</i> (Simulasi Audit) dengan 1 Fakultas terpilih. <i>User Acceptance Testing</i> (UAT). Perbaikan Bugs. <b>Output:</b> Berita Acara UAT, Sistem Stabil.	39

<b>VI. Deployment &amp; Training</b>	Bulan 8	Setup Server Produksi, Migrasi Data, Pelatihan Auditor & Auditee secara masif <i>(Training of Trainers).</i> <b>Output:</b> Sistem Live, User Manual.	32
<b>VII. Evaluasi Pasca-Implementasi</b>	Bulan 9	Monitoring performa server saat beban puncak, survei kepuasan pengguna, Rapat Evaluasi Proyek. <b>Output:</b> Laporan Akhir Proyek.	41

## 7. Keamanan, Tata Kelola (SPBE), dan Mitigasi Risiko

Dalam kerangka Sistem Pemerintahan Berbasis Elektronik (SPBE), keamanan informasi adalah mandat, bukan opsi. Sistem E-SPMI menyimpan data sensitif kinerja institusi yang bersifat rahasia strategis.<sup>4</sup>

### 7.1 Strategi Keamanan Berlapis (Defense in Depth)

- **Keamanan Aplikasi:** Penerapan *Role-Based Access Control* (RBAC) yang granular. Dosen biasa tidak boleh melihat data audit prodi lain. Implementasi proteksi CSRF, XSS, dan SQL Injection pada level framework (Laravel/Spring Security).<sup>42</sup>
- **Enkripsi Data:** Data sensitif (password, NIK, gaji dosen jika ada) harus dienkripsi di database (AES-256). Jalur komunikasi wajib menggunakan HTTPS/TLS 1.3.<sup>22</sup>
- **Audit Trail (Jejak Digital):** Tabel log harus mencatat "Who, What, When, Where" untuk setiap perubahan data kritis. Log ini harus bersifat *append-only* untuk menjamin integritas forensik.<sup>23</sup>

### 7.2 Infrastruktur dan Pemulihan Bencana (Disaster Recovery)

- **Backup Strategy:** Backup database harian (Incremental) dan mingguan (Full) ke lokasi fisik terpisah (*Off-site backup* atau Cloud Storage berbeda region).
- **High Availability:** Jika anggaran memungkinkan, gunakan konfigurasi *Load Balancer*

dengan minimal 2 node aplikasi server untuk mencegah *Single Point of Failure*.<sup>45</sup>

---

## 8. Penutup: Menuju Ekosistem Mutu Digital yang Adaptif

Pengembangan sistem E-SPMI yang komprehensif adalah langkah strategis bagi universitas untuk merespons dinamika regulasi Permendiktisaintek No. 39 Tahun 2025. Dengan mengadopsi arsitektur **Modular Monolith**, universitas mendapatkan fleksibilitas untuk berevolusi tanpa terbebani kompleksitas infrastruktur yang berlebihan.

Kunci keberhasilan sistem ini bukan hanya pada kecanggihan kode programnya, melainkan pada integrasinya yang mulus dengan proses bisnis universitas (Siklus PPEPP) dan kemampuannya untuk menyajikan data yang dapat ditindaklanjuti (*actionable insights*) bagi pimpinan. E-SPMI yang dirancang dengan blueprint ini tidak hanya akan menjadi alat kepatuhan administratif, tetapi menjadi mesin penggerak utama dalam perjalanan universitas mencapai predikat Unggul dan rekognisi global. Implementasi yang disiplin sesuai *roadmap*, didukung oleh komitmen pimpinan (*tone from the top*) dan keamanan siber yang tangguh, akan memastikan investasi teknologi ini memberikan *Return on Investment* (ROI) maksimal berupa peningkatan budaya mutu institusi yang nyata.

### Karya yang dikutip

1. PERATURAN MENTERI PENDIDIKAN TINGGI, SAINS, DAN TEKNOLOGI REPUBLIK INDONESIA NOMOR 39 TAHUN 2025 TENTANG PENJAMINAN MUTU PENDIDIKAN TINGGI - Universitas Potensi Utama, diakses Januari 22, 2026,  
<https://lpm.potensi-utama.ac.id/pengumuman/peraturan-menteri-pendidikan-tinggi-sains-dan-teknologi-republik-indonesia-nomor-39-tahun-2025-tentang-penjaminan-mutu-pendidikan-tinggi>
2. Permendiktisaintek No. 39 Tahun 2025: Standar Baru Penjaminan Mutu Pendidikan Tinggi, diakses Januari 22, 2026,  
<https://bpm.unair.ac.id/permendiktisaintek-no-39-tahun-2025-standar-baru-penjaminan-mutu-pendidikan-tinggi/>
3. Mengenal Siklus Sistem Penjaminan Mutu Internal (SPMI) Di ..., diakses Januari 22, 2026,  
<https://sevima.com/mengenal-siklus-sistem-penjaminan-mutu-internal-spmi-di-perguruan-tinggi/>
4. Keamanan Informasi Dalam Sistem Pemerintahan Berbasis Elektronik (SPBE)-1\_1749529839.pdf - Wiyata Kinarya, diakses Januari 22, 2026,  
[https://wiyatakinarya.kemendikdasmen.go.id/kms/get-temporary-pdf/Keamanan%20Informasi%20Dalam%20Sistem%20Pemerintahan%20Berbasis%20Elektronik%20%28SPBE%29-1\\_1749529839.pdf?path=https%253A%252F%252Fwiyatakinarya.kemendikdasmen.go.id%252Fcloud%252Fs%252Fdrdi9gFBByEiDPsX%252Fdownload%252FKeamanan%2BInformasi%2BDalam%2BSistem%2BPemerintahan%2BB](https://wiyatakinarya.kemendikdasmen.go.id/kms/get-temporary-pdf/Keamanan%20Informasi%20Dalam%20Sistem%20Pemerintahan%20Berbasis%20Elektronik%20%28SPBE%29-1_1749529839.pdf?path=https%253A%252F%252Fwiyatakinarya.kemendikdasmen.go.id%252Fcloud%252Fs%252Fdrdi9gFBByEiDPsX%252Fdownload%252FKeamanan%2BInformasi%2BDalam%2BSistem%2BPemerintahan%2BB)

[erbasis%2BElektronik%2B%2528SPBE%2529-1\\_1749529839.pdf](#)

5. Standar Nasional Pendidikan Tinggi (SN Dikti) - Badan Penjaminan Mutu UM, diakses Januari 22, 2026,  
<https://bpm.um.ac.id/wp-content/uploads/2020/06/Standar-Nasional-Pendidikan-Tinggi-SN-Dikti.pdf>
6. Buku Panduan Indikator Kinerja Utama PTN v2 (2021) - LPM USK, diakses Januari 22, 2026,  
<https://lpm.usk.ac.id/storage/2024/08/Buku-Panduan-Indikator-Kinerja-Utama-PTN-versi2-Kemdikbud-2021.pdf>
7. Modular Monolith Architecture in Cloud Environments: A Systematic Literature Review, diakses Januari 22, 2026, <https://www.mdpi.com/1999-5903/17/11/496>
8. Behold the Modular Monolith: The Architecture Balancing Simplicity and Scalability, diakses Januari 22, 2026,  
<https://dev.to/naveens16/behold-the-modular-monolith-the-architecture-balancing-simplicity-and-scalability-2d4>
9. Permendiktisaintek Nomor 39 Tahun 2025 tentang: Penjaminan Mutu Pendidikan Tinggi - lldikti vi, diakses Januari 22, 2026,  
<https://lldikti6.id/wp-content/uploads/2025/11/01.-MATERI-1-SNDIKTI-Permen-39-2025.pdf>
10. SIKLUS PPEPP - Lembaga Penjaminan Mutu, diakses Januari 22, 2026,  
<https://bpm.uai.ac.id/siklus-ppepp/>
11. dokumen sn dikt dan pelampauan sndikti | 3 - FSH UNISNU, diakses Januari 22, 2026,  
<https://syariah.unisnu.ac.id/assets/media/dokumen-sn-dikti-pelampauan-standar-mutu-fsh.pdf>
12. Penjaminan Mutu Pendidikan Tinggi - Permendikbudriset No. 53 Tahun 2023, diakses Januari 22, 2026,  
<https://peraturan.bpk.go.id/Details/265158/permendikbudriset-no-53-tahun-2023>
13. 1 PEDOMAN PENERAPAN SIKLUS PPEPP SPMI USK - LPM USK, diakses Januari 22, 2026,  
<https://lpm.usk.ac.id/storage/2025/11/DOKUMEN-PPEPP-SPMI-USK-FINAL.pdf>
14. Aplikasi eSPMI: Best Practices SPMI (Sistem Penjaminan Mutu Internal) di Perguruan Tinggi, diakses Januari 22, 2026,  
<https://ecampuz.com/aplikasi-espml-best-practices-spmi-di-perguruan-tinggi/>
15. Rancang Bangun Sistem Informasi Audit Mutu Internal Berbasis IAPS 4.0 - E-Journal STMIK Dharmapala Riau, diakses Januari 22, 2026,  
<https://ojs.stmikdharmapalariau.ac.id/index.php/jkb/article/download/7/6>
16. Rancang Bangun Sistem Audit Mutu Internal Guna Optimalisasi Kinerja Penjaminan Mutu Perguruan Tinggi, diakses Januari 22, 2026,  
<https://sistemas.ftik.unisi.ac.id/index.php/stmsi/article/viewFile/1374/390>
17. Implementation of Academic Information System (SIAKAD) to Support Institutional Accreditation at the Air Force Academy, diakses Januari 22, 2026,  
<https://srhformosapublisher.org/index.php/fjas/article/download/253/362/1597>
18. Implementation of Academic Information System (SIAKAD) to Support Institutional Accreditation at the Air Force Academy - ResearchGate, diakses

- Januari 22, 2026,  
[https://www.researchgate.net/publication/394584615 Implementation of Academic Information System SIAKAD to Support Institutional Accreditation at the Air Force Academy](https://www.researchgate.net/publication/394584615_Implementation_of_Academic_Information_System_SIAKAD_to_Support_Institutional_Accreditation_at_the_Air_Force_Academy)
19. 8 Indikator Kinerja Utama Perguruan Tinggi - SEVIMA, diakses Januari 22, 2026,  
<https://sevima.com/8-iku/>
20. 7 best practices for building and maintaining API integrations - Merge.dev, diakses Januari 22, 2026,  
<https://www.merge.dev/blog/api-integration-best-practices>
21. ERP API Integration Guides & Resources, diakses Januari 22, 2026,  
<https://www.getknit.dev/blog/erp-api-integration-guides-resources>
22. SK MANAJEMEN KEAMANAN INFORMASI.pdf - Diskominfo Badung, diakses Januari 22, 2026,  
<https://diskominfo.badungkab.go.id/storage/diskominfo/file/SK%20MANAJEMEN%20KEAMANAN%20INFORMASI.pdf>
23. Auditing schema tables - IBM, diakses Januari 22, 2026,  
<https://www.ibm.com/docs/en/sim/7.0.2?topic=reference-auditing-schema-tables>
24. Master Database Audit: Your Essential Guide - Data-Sleek, diakses Januari 22, 2026, <https://data-sleek.com/blog/database-audit-essential-guide/>
25. What Is a Modular Monolith? - Milan Jovanović, diakses Januari 22, 2026, <https://www.milanjovanovic.tech/blog/what-is-a-modular-monolith>
26. Modular Monolith Architecture with .NET | ABP.IO, diakses Januari 22, 2026, <https://abp.io/architecture/modular-monolith>
27. Understanding modular ERP: Benefits, challenges, and implementation - Priority Software, diakses Januari 22, 2026, <https://www.priority-software.com/resources/modular-erp/>
28. Digitalisasi Sistem Penjaminan Mutu Internal (SPMI) untuk Institusi Pendidikan Berbasis Website - jurnal mdp, diakses Januari 22, 2026, <https://jurnal.mdp.ac.id/index.php/msc/article/download/11183/2315/>
29. Laravel vs Spring Boot: Top Differences - GeeksforGeeks, diakses Januari 22, 2026, <https://www.geeksforgeeks.org/blogs/laravel-vs-spring-boot/>
30. Laravel vs. Spring Boot: Best Framework for Your Web Apps? - Lucent Innovation, diakses Januari 22, 2026, <https://www.lucentinnovation.com/resources/technology-posts/laravel-vs-spring-boot>
31. Spring Boot vs Laravel - DEV Community, diakses Januari 22, 2026, <https://dev.to/ysf0009/spring-boot-vs-laravel-2l7d>
32. Permudah pengelolaan Penjaminan Mutu internal anda - Jamitu.id, diakses Januari 22, 2026, <https://app.jamitu.id/>
33. DESIGN AND DEVELOPMENT OF AN INTERNAL QUALITY AUDIT ..., diakses Januari 22, 2026, [https://ejournal.nusamandiri.ac.id/index.php/pilar/article/download/4729/1155/1546\\_1](https://ejournal.nusamandiri.ac.id/index.php/pilar/article/download/4729/1155/1546_1)
34. Audit data schema overview - Progress Documentation, diakses Januari 22, 2026, <https://docs.progress.com/bundle/openedge-security-and-auditing/page/Audit-d>

[ata-schema-overview.html](#)

35. Sistem Informasi Lembaga Penjamin Mutu (LPM) pada AMIK Luwuk Banggai Berbasis Web, diakses Januari 22, 2026,  
<https://journalcenter.org/index.php/jupsim/article/download/5357/4154/21429>
36. RANCANG BANGUN MODEL DATA UNTUK MANAJEMEN DOKUMEN SISTEM PENJAMINAN MUTU INTERNAL | SINTECH (Science and Information Technology) Journal, diakses Januari 22, 2026,  
<https://jurnal.instiki.ac.id/index.php/sintechjournal/article/view/734>
37. Merancang Perangkat Lunak Sistem Penjaminan Mutu ... - Neliti, diakses Januari 22, 2026, <https://media.neliti.com/media/publications/443041-none-49388c0a.pdf>
38. How To Conduct Data Quality Audits: A Step-by-Step Guide - Monte Carlo Data, diakses Januari 22, 2026,  
<https://www.montecarlodata.com/blog-how-to-conduct-data-quality-audits/>
39. Tahapan Manajemen Proyek Sistem Informasi - IDS Digital College, diakses Januari 22, 2026, <https://ids.ac.id/tahapan-manajemen-proyek-sistem-informasi/>
40. 6 Tahapan pada IT Project Management. Simak Selengkapnya! - Universitas Bakrie, diakses Januari 22, 2026,  
<https://bakrie.ac.id/articles/804-6-tahapan-pada-it-project-management-simak-selengkapnya.html>
41. SISTEM AUDIT MUTU INTERNAL POLITEKNIK ENGINERING INDORAMA - Direktori Jurnal Nasional STIE Jayakarta, diakses Januari 22, 2026,  
<https://journal.stiejayakarta.ac.id/index.php/JMBJayakarta/article/download/188/104>
42. Role-Based Access Control (RBAC) - GW Information Technology, diakses Januari 22, 2026, <https://it.gwu.edu/role-based-access-control-rbac>
43. What is Role-Based Access Control (RBAC) and How It Works? - Splashtop, diakses Januari 22, 2026,  
<https://www.splashtop.com/blog/role-based-access-control>
44. KEAMANAN SPBE PADA TRANSFORMASI DIGITAL, diakses Januari 22, 2026,  
<https://spbe.pontianak.go.id/storage/materi/November2023/HqEr20IJHJm8xXzTqzT0.pdf>
45. manajemen dan standar keamanan sistem pemerintahan berbasis elektronik - :: Jabarprov-csirt, diakses Januari 22, 2026,  
[https://csirt.jabarprov.go.id/assets/event/Dwi-Kardono-\(BSSN\)-Manajemen-Keamanan-Aplikasi-SPBE-dan-Hardening.pdf](https://csirt.jabarprov.go.id/assets/event/Dwi-Kardono-(BSSN)-Manajemen-Keamanan-Aplikasi-SPBE-dan-Hardening.pdf)