Gudang Jurnal Pengabdian Masyarakat
Volume 3 ; Nomor 2 ; Juli 2025 ; Page 501-513
Doi : https://doi.org/10.59435/gjpm.v3i2.1741
Website : https://gudangjurnal.com/index.php/gjpm
E-ISSN : 3024-8019

PKM : Sosialisasi Keamanan Website Sekolah dalam Pencegahan Akses Ilegal dan Serangan Siber di SMA Negeri 4 Sumatera Barat

Thomson Mary[1*], Faiza Rini[2], Nia Febriyani[3], Herisvan Hendra[4]

[1,2] Program Studi Teknologi Informasi, Fakultas Sains dan Teknologi, Universitas PGRI Sumatera Barat
[3]Program Studi Sains Data, Fakultas Sains dan Teknologi, Universitas PGRI Sumatera Barat
[4]Program Studi Pendidikan Informatika, Fakultas Sains dan Teknologi, Universitas PGRI Sumatera Barat
[1*]thomsonmary1980@gmail.com, [2]faizarini201104@gmail.com, [3]febriyaninia@gmail.com, [4]herisvan321@gmail.com

Abstrak

Kegiatan pengabdian masyarakat ini bertujuan untuk meningkatkan kesadaran dan kemampuan tenaga kependidikan serta staf administrasi SMA Negeri 4 Sumatera Barat dalam mengamankan website sekolah dari ancaman siber. Latar belakang kegiatan ini adalah kerentanan website sekolah terhadap serangan seperti defacement dan akses ilegal, yang pernah mengakibatkan pengalihan ke situs judi online. Metode yang digunakan meliputi sosialisasi, pelatihan teknis, dan pendampingan dalam mengimplementasikan fitur keamanan berbasis Laravel 11, seperti *Rate Limiting dan Role-Based Access Control* (RBAC) menggunakan *Laravel Spatie Permissions*. Hasil kegiatan menunjukkan peningkatan signifikan dalam pemahaman peserta tentang keamanan website, serta keberhasilan implementasi fitur keamanan yang mengurangi risiko serangan siber. Selain itu, disusun modul pelatihan dan dokumentasi teknis untuk memastikan keberlanjutan pengelolaan website secara mandiri. Kegiatan ini memberikan dampak positif berupa peningkatan keamanan data dan reputasi sekolah, sekaligus menjadi model bagi institusi pendidikan lain dalam menghadapi tantangan keamanan siber.

Kata Kunci: Keamanan Bebsite, Laravel 11, RBAC, Rate Limiting, SMA Negeri 4 Sumatera Barat

PENDAHULUAN

Di era transformasi digital yang berkembang pesat saat ini, keberadaan sistem informasi sekolah telah menjadi komponen penting dalam ekosistem pendidikan modern. Website sekolah tidak hanya berfungsi sebagai sarana penyampaian informasi akademik, tetapi juga menjadi platform komunikasi antara sekolah dengan siswa, orang tua, dan masyarakat luas. Perlindungan terhadap keamanan informasi dapat dilakukan dengan beragam cara dengan tujuan untuk menyakinkan intergritas, kerahasiaan, kekonsistenan data yang diolah. (Isnaini et al., 2020). Namun sayangnya, perkembangan teknologi ini juga diiringi dengan meningkatnya ancaman keamanan siber yang semakin kompleks dan mengkhawatirkan. Perusahaan keamanan siber Kaspersky menemukan serangkaian serangan siber kompleks yang melibatkan pengambilan informasi dari layanan sah seperti GitHub, Microsoft Learn Challenge, Quora, hingga jejaring sosial (CNN Indonesia, 2025)

SMA Negeri 4 Sumatera Barat (Keberbakatan Olahraga) sebagai salah satu sekolah unggulan di Provinsi Sumatera Barat telah mengalami dampak nyata dari ancaman keamanan siber ini. Meningkatnya integrasi Laravel ke dalam kurikulum pendidikan tinggi didorong oleh keunggulan inherennya dalam mengajarkan praktik pengembangan web modern.(Wahid et al., 2025). Beberapa waktu lalu, website resmi sekolah yang beralamat di https://sman4sumaterabarat.sch.id sering menjadi target serangan siber yang cukup serius. Para peretas berhasil membajak website tersebut dan mengalihkannya ke situs judi online. Kejadian ini bukan hanya menyebabkan gangguan operasional, tetapi juga telah mengkhawatirkan pihak sekolah sebagai institusi pendidikan terkemuka. Pelajar sebagai generasi penerus bangsa, sangat akrab dengan penggunaan teknologi, terutama internet. Namun, dengan kemudahan akses informasi yang ditawarkan, muncul pula berbagai risiko yang mengancam keamanan data pribadi dan privasi mereka (Welnof Satria;dkk, 2025). Menurut laporan dari Cybersecurity dan Infrastructure Security Agency(CISA), serangan siber semakin meningkat dan menjadi lebih kompleks, menargetkan individu, organisasi, dan bahkan institusi pendidikan (CISA, 2021) . Seiring dengan meningkatnya penggunaan internet dan perangkat digital, kejahatan siber semakin berkembang dan menjadi ancaman serius bagi individu, perusahaan, hingga pemerintah. (Aabid et al., 2025)

Masalah keamanan website di SMA Negeri 4 Sumatera Barat ini muncul dari beberapa faktor mendasar. Pertama, kurangnya pemahaman tentang pentingnya keamanan digital di kalangan tenaga pendidik dan staf administrasi. Kedua, tidak adanya tim IT khusus yang kompeten dalam menangani masalah keamanan siber. Ketiga, sistem website yang digunakan belum dilengkapi dengan protokol keamanan yang memadai. Keempat, minimnya kesadaran tentang kebijakan keamanan data dan informasi di lingkungan sekolah. Pada ruang siber sendiri dapat dilakukan berbagai aktivitas, misalnya seperti berkomunikasi, mencari informasi, membaca artikel, transaksi, dan lain sebagainya(Amirulloh et al., 2025) Investasi dalam teknologi keamanan siber canggih sangat penting untuk melindungi aset dan data lembaga keuangan.

Teknologi ini meliputi: Sistem memantau lalu lintas jaringan dan mengidentifikasi aktivitas mencurigakan yang dapat mengindikasikan serangan (Setiawan, 2025)

Kondisi ini diperparah dengan beberapa fakta lapangan yang ditemukan:

a. Website sekolah sama belum mengoptimalkan sistem pembatasan akses (*access control*)
b. Belum ada mekanisme untuk mencegah serangan *brute force* atau DDoS
c. Database rentan terhadap serangan SQL *injection*
d. Belum ada prosedur *backup* data yang teratur
e. Staf sekolah tidak memiliki pengetahuan dasar tentang *cyber hygiene*

Fenomena ini sebenarnya tidak hanya terjadi di SMA Negeri 4 Sumatera Barat. Banyak sekolah lain di Indonesia juga menghadapi tantangan serupa dalam mengamankan website mereka. Namun, kasus di SMA Negeri 4 Sumatera Barat ini cukup representatif karena menunjukkan betapa rentannya sistem informasi pendidikan terhadap ancaman siber jika tidak dikelola dengan baik.

Kondisi inilah yang mendorong pentingnya dilaksanakan program pengabdian masyarakat yang berfokus pada peningkatan keamanan website sekolah. Program ini tidak hanya bertujuan untuk memperbaiki kerentanan teknis yang ada, tetapi juga membangun kapasitas sumber daya manusia di sekolah agar mampu mengelola dan mengamankan website mereka secara mandiri di masa depan. Dengan demikian, website sekolah dapat benar-benar menjadi aset strategis yang mendukung proses pembelajaran dan komunikasi, bukan menjadi sumber masalah yang mengancam keamanan informasi. Perkembangan teknologi yang semakin maju memunculkan kejahatan dalam bentuk baru yang disebut sebagai cyber crime, kejahatan ini dilakukan secara kasat mata dimana ancaman yang timbul dari kejahatan ini tidak berbentuk fisik. (Sintia Saramuke et al., 2025). Ancaman terhadap infrastruktur TI dapat dikategorikan menjadi ancaman fisik, manusia, dan teknis. Ancaman fisik mencakup kerusakan perangkat keras dan kebakaran, sedangkan ancaman manusia, seperti insider threat dan phishing, seringkali melibatkan kelalaian atau tindakan disengaja oleh individu dalam organisasi. (Rizki Sandrina Ayu, 2025)

Perkembangan teknologi informasi telah menjadikan *website* sekolah sebagai sarana vital untuk komunikasi dan diseminasi informasi akademik. Namun, meningkatnya ancaman siber seperti *defacement*, *Distributed Denial of Service* (DDoS), dan *SQL injection* menimbulkan risiko serius bagi institusi pendidikan (Kshetri, 2022). SMA Negeri 4 Sumatera Barat (Keberbakatan Olahraga) menjadi salah satu korban, di mana *website* resminya pernah mengalami pembajakan dan dialihkan ke situs judi online. Insiden ini menunjukkan kerentanan sistem yang dipicu oleh kurangnya pemahaman teknis staf sekolah dan minimnya implementasi mekanisme keamanan modern (Almeida et al., 2020).

Solusi yang ditawarkan dalam pengabdian ini adalah implementasi sistem keamanan berbasis *Laravel 11*, sebuah *framework PHP* yang menyediakan fitur keamanan bawaan seperti proteksi terhadap *Cross-Site Scripting* (XSS) dan *Cross-Site Request Forgery* (CSRF) (Stauffer, 2021). Melalui pelatihan intensif, tim pengabdian memperkenalkan *Rate Limiting* untuk mencegah serangan DDoS dan *Role-Based Access Control* (RBAC) menggunakan *Laravel Spatie Permissions* guna mengatur hak akses pengguna (Spatie, 2023). Pendekatan ini dilengkapi dengan konfigurasi database *MariaDB* untuk enkripsi data dan *backup* otomatis, serta pendampingan jangka panjang untuk memastikan keberlanjutan.

Beberapa penelitian dan pengabdian terkait telah dilakukan dalam lima tahun terakhir, namun masih terdapat celah (*gap*) yang menjadi lsan inovasi kegiatan ini:

a. Mengkaji ancaman siber di sektor pendidikan, tetapi tidak memberikan solusi teknis berbasis *framework* tertentu serta membahas keunggulan keamanan *Laravel* dalam konteks *website* sekolah. (Stauffer, 2019)
b. Menekankan pentingnya pelatihan kesadaran keamanan siber, namun tidak menyertakan implementasi praktis seperti *Laravel Spatie Permissions*. (Almeida et al., 2020)
c. Mengadakan pelatihan keamanan *website* bagi guru SMK, tetapi hanya menggunakan *WordPress* dengan fitur keamanan terbatas. (Donalds et al., 2022)
d. Mengimplementasikan RBAC di perguruan tinggi, tetapi tidak mengintegrasikannya dengan *Rate Limiting* atau database *MariaDB*. (Yuricha dan Phan, 2023a)

Gap analysis menunjukkan keunikan pengabdian ini dalam hal:

a. Integrasi teknologi: Menggabungkan *Laravel 11*, *MariaDB*, dan *Laravel Spatie Permissions* dalam satu sistem.
b. Pendekatan holistik: Tidak hanya pelatihan teori, tetapi juga pendampingan teknis dan evaluasi berkala.
c. Keterbaharuan: Mengadopsi fitur terbaru *Laravel 11* yang belum banyak diaplikasikan di sekolah-sekolah Indonesia.

Tujuan utama pengabdian ini adalah:

a. Meningkatkan kapasitas staf sekolah dalam mengidentifikasi dan memitigasi ancaman siber.
b. Membangun sistem keamanan *website* yang tangguh melalui *Laravel 11* dan *MariaDB*.
c. Menghasilkan modul pelatihan dan dokumentasi teknis sebagai panduan mandiri.

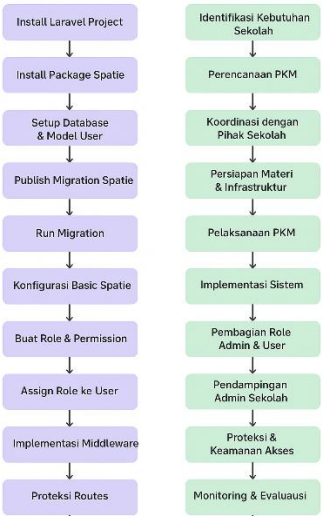
Diharapkan, hasil pengabdian ini tidak hanya menyelesaikan masalah keamanan di SMA Negeri 4 Sumatera Barat, tetapi juga menjadi referensi bagi institusi pendidikan lain di Indonesia.

METODE

Tahapan Pengabdian

Tahapan pengabdian berupa tahapan dalam merancang sistem informasi website itu sendiri dimana website tersebut dibangun dengan framework laravel yang memberikan pengamanan yang lebih banyak seperti penggunaan RBAC dan penggunaan *library* laravel seperti laravel *spatie*. Kontrol terhadap akses (access control) menjadi salah satu solusi ideal yang dapat diterapkan dalam sistem yang memiliki isu keamanan data dalam hal otorisasi setiap peran yang ada dalam sistem (Yuricha dan Phan, 2023b)

Berikut ini tahapan yang dilakukan :



Gambar 1. Implementasi Tahapan Keamanan Sistem Website dengan Tahapan PKM

Berdasarkan diagram implemenatasi tersebut ada beberapa langkah yang hampir sama dan dibuat dalam satu langkah saja, berikut langkah yang menjadi bahasan PKM:

1. *Install* Laravel *Project* dan Identifikasi Kebutuhan Sekolah
2. Install *Package Spatie* dan Perencanaan PKM
3. Setup Database dan Model User, *Publish Migration Spatie* dan Persiapan Materi dan Infrastruktur, *Run Migration* dan Pelaksanaan PKM, Konfigurasi *Basic Spatie* dan Implementasi Sistem, Buat *Role* dan *Permission* dan Pembagian *Role* dan Personal Pengelola Admin Sekolah, Assign *Role* ke User dan Pendampingan Admin Sekolah
4. Implementasi *Middleware* dan Proteksi dan Keamanan Website, Proteksi *Routes* dan Monitoring dan Evaluasi

Untuk mendukung implementasi Program Kreativitas Mahasiswa (PKM) di lingkungan sekolah, diperlukan sistem informasi yang terstruktur agar pengelolaan data, hak akses, serta peran setiap pengguna dapat berjalan optimal. Salah satu pendekatan yang dapat digunakan adalah dengan memanfaatkan *framework* Laravel yang dipadukan dengan *Package Spatie* untuk mengatur *role* dan *permission*.

Tahapan ini tidak hanya mencakup aspek teknis pengembangan sistem, tetapi juga menyesuaikan dengan kebutuhan nyata di sekolah, seperti pendaftaran proposal PKM, validasi oleh admin, serta pengelolaan hak akses antara mahasiswa, dosen pembimbing, dan admin sekolah. Dengan adanya *role* dan *permission*, sistem dapat memberikan batasan dan kontrol yang jelas bagi setiap pengguna sesuai perannya.

Berikut ini adalah tahapan implementasi sistem informasi PKM untuk admin sekolah yang disusun berdasarkan alur pengembangan Laravel dan integrasi *Spatie*:

Tabel 1. Implementasi Tahapan Keamanan Sistem Website dengan Tahapan PKM

Laravel Flow	**Tahapan PKM Admin Sekolah**	**Keterkaitan dengan PKM**
Install Laravel Project	Identifikasi Kebutuhan Sekolah	Seperti *install project*, tahap awal PKM adalah memahami kebutuhan agar sistem bisa dibangun sesuai kondisi sekolah.
Install Package Spatie	Perencanaan PKM	Sama seperti menambahkan *package* penting, PKM butuh perencanaan *tools*, modul, dan strategi sebelum jalan.
Setup Database dan Model User	Koordinasi dengan Pihak Sekolah	Database dan model = dasar sistem. Dalam PKM, koordinasi dengan sekolah adalah fondasi awal sebelum lanjut.
Publish Migration Spatie	Persiapan Materi dan Infrastruktur	*Migration* mempersiapkan struktur DB, sedangkan PKM menyiapkan materi, modul, dan infrastruktur.

Run Migration	Pelaksanaan PKM	*Migration* = eksekusi struktur DB. Sama dengan PKM yang mulai dijalankan sesuai rencana.
Konfigurasi Basic Spatie	Implementasi Sistem	Konfigurasi awal sistem = penerapan aplikasi/solusi di sekolah.
Buat Role dan Permission	Pembagian *Role Admin dan User*	Sama persis: *Role* → admin/guru, Permission → hak akses pengguna sistem.
Assign Role ke User	Pendampingan Admin Sekolah	*Assign role* = siapa melakukan apa, sama dengan melatih admin mengelola sistemnya.
Implementasi Middleware	Proteksi dan Keamanan Akses	*Middleware* = filter/keamanan. PKM → proteksi data, setting password, hak akses.
Proteksi Routes	Pemanfaatan Fitur Sistem	Sama seperti proteksi *routes*, sekolah hanya bisa akses fitur sesuai role.
Blade Directive	Monitoring dan Evaluasi	*Blade directive* untuk menampilkan kondisi sesuai *role*, sama dengan sekolah menguji apakah *role*/akses berjalan sesuai.
Testing	Laporan dan Rekomendasi	Testing di Laravel = memastikan sistem jalan, sama dengan PKM → membuat laporan, evaluasi, dan rekomendasi lanjutan.

1. Langkah Kerja *Install* Laravel *Project* dan Identifikasi Kebutuhan Sekolah

Untuk Install Laravel dalam membuat sistem pada website sekolah dipandu dalam website laravel https://laravel.com. Penginstalan laravel memakai *composer* dengan sintak baris *composer create-project laravel/laravel nama-project* berikut ini uraian secara lengkap:

Langkah ini adalah fondasi. Kesalahan di sini akan berimbak pada semua tahap selanjutnya.

a. Dari Sisi Teknis Laravel *Install* Laravel
   1. Persiapan Sistem:
      a) Memastikan PHP (>= 8.2), Composer, dan *Database Server* (e.g., MySQL) telah terinstall di komputer
      b) Memverifikasi dengan menjalankan perintah di terminal/CMD:
         php -v
         composer -v
         mysql --version # atau perintah untuk database lain
   2. Menjalankan Perintah Installasi:
      a) Membuat project Laravel baru dengan nama sistem-sekolah.
         composer create-project laravel/laravel sistem-sekolah
      b) Masuk ke direktori project.
         cd sistem-sekolah
   3. Konfigurasi *Environment*:
      a) Salin file .env.example menjadi .env.
         cp .env.example .env
      b) Generate *application key* yang crucial untuk keamanan.
         php artisan key:generate
      c) Buka file .env dan konfigurasi koneksi database :
         DB_CONNECTION=mysql
         DB_HOST=127.0.0.1
         DB_PORT=3306
         *DB_DATABASE=nama_database_*
         *DB_USERNAME=username_database_*
         *DB_PASSWORD=password_database_*
   4. Test Installasi:
      a) Menjalankan server *development* lokal untuk memastikan semuanya berjalan. *php artisan serve*

b. Dari Sisi PKM Identifikasi Kebutuhan Sekolah

   Identifikasi Kebutuhan Sekolah berupa langkah sebagai berikut:
   1. Persiapan Sekolah:
      a) Sekolah Menyiapkan Alat Bantu Berupa: Google Form, Microsoft Excel, atau buku catatan.
      b) Sekolah Menentukan Pihak Yang Terlibat: Kepala Sekolah, Wakil Kepala Sekolah, Guru, *Staff* Tata Usaha, Komite Sekolah.
   2. Menjalankan Perintah Identifikasi (Kegiatan Lapangan):
      a) Melakukan Wawancara dan Observasi:

i. Dengan Kepala Sekolah: "Apa visi/misi untuk website sekolah ?
ii. Dengan *Staff* TU: Informasi apa yang sering dimuat dalam website sekolah ?

3. Kebutuhan Dokumentasi:
   a) Membuat Daftar Fitur
      i. Fitur 1: Manajemen Berita (CRUD, import/export Excel)
      ii. Fitur 2: Manajemen Menu
   b) Menentukan Peran Pengguna : “Siapa saja yang akan menggunakan sistem?”
      I. Admin
      II. Kepala Sekolah
      III. *Staff* Tata Usaha
4. Identifikasi Kebutuhan (Validasi):
   a) Mengembalikan daftar fitur dan peran pengguna kepada Kepala Sekolah dan perwakilan *Staff* untuk divalidasi: "Apakah ini sudah sesuai dengan kebutuhan dan ekspektasi Bapak/Ibu?"
   b) Revisi berdasarkan masukan mereka. Tindakan ini sama persis dengan *php artisan serve* untuk memastikan "aplikasi konsep" yang sudah benar dan dapat dijalankan di lingkungan sekolah.

Tabel 2. Keterkaitan Langkah Teknis Laravel *Install* Laravel dengan Keterkaitan Nyata pada PKM

Laravel Action	**PKM *Admin Action***	**Keterkaitan dengan PKM**
composer create-project...	Melakukan wawancara dan observasi	Merupakan aksi untuk memulai pembuatan sistem yaitu mengunduh kode dasa serta mengumpulkan kode kebutuhan.
Configure .env file	Membuat Daftar Fitur dan Peran	Merupakan konfigurasi awal. File .env mengonfigurasi koneksi database, sedangkan Daftar Fitur mengonfigurasi "koneksi" antara sistem dan kebutuhan pengguna.
php artisan key:generate	Validasi kebutuhan ke Kepsek	*Generate key* merupakan membuat identitas unik dan keamanan untuk aplikasi. Validasi adalah memberikan "identitas" dan "pengakuan" resmi dari pihak sekolah bahwa kebutuhan yang dirumuskan sudah benar.
php artisan serve	Mem-presentasikan daftar fitur	*artisan serve* adalah *test* untuk memastikan *framework* siap dibangun. Presentasi daftar fitur adalah *test* untuk memastikan *konsep* siap untuk dibangun.

2. Langkah Kerja *Install Package Spatie* dan Mempelajari Konsep Manajemen Akses (RBAC)

Langkah ini adalah tentang menambahkan *engine* yang akan menggerakkan seluruh sistem keamanan dan izin di aplikasi.

a. Dari Sisi Teknis Laravel *Install Package Spatie*:
   1. Prasyarat:
      i. Memastikan posisi root berada pada directory project Laravel (cd sistem-sekolah).
      ii. Memastikan koneksi internet stabil untuk mengunduh package dari Packagist.
   2. Menjalankan Perintah Installasi:
      i. Menggunakan *Composer*, *package manager* untuk PHP, untuk mengunduh dan menginstall *Package Spatie Laravel-Permission.*
      composer require spatie/laravel-permission
   3. Memverifikasi Installasi :
      i. Memeriksa package telah tercatat dengan benar di composer.json .
      ii. Menjalankan *composer show spatie/laravel-permission* untuk melihat info package yang terinstall.

b. Dari Sisi PKM Konsep Manajemen Akses (RBAC)

Mempelajari Konsep Manajemen Akses (RBAC) merupakan proses memahami dan mendefinisikan aturan-aturan sebelum diterapkan.

   1. Prasyarat:
      i. Membuat dokumen luaran yaitu Daftar Fitur dan Daftar Peran Pengguna (*Actor*).
      ii. Menyiapkan Alat Bantu: Papan tulis, kertas flipchart, atau *tool diagram* seperti *Lucidchart*/*Miro*.
   2. Menjalankan "Perintah" Mempelajari Konsep (Kegiatan *Brainstorming*):
      i. Tugas untuk Setiap Peran: Mengambil daftar fitur dan tentukan *siapa yang boleh melakukan apa*.

Tabel 2. Keterkaitan Langkah Teknis Laravel *Install Package Spatie* dengan Keterkaitan Nyata pada PKM

Laravel Action	PKM *Admin Action*	Keterkaitan dengan PKM
composer require spatie/...	Membuat Matrix RBAC	Menyiapkan kerangka (framework) untuk kontrol. *Spatie* adalah kerangka teknis di *code*, Matrix RBAC adalah kerangka konseptual di atas kertas.
Package disimpan di *vendor/*	Dokumen Matrix RBAC	*vendor/* adalah tempat menyimpan *tool* teknis. Dokumen Matrix adalah tempat menyimpan *konsep* aturan. Keduanya akan menjadi referensi utama untuk langkah-langkah selanjutnya.
Composer mengelola dependensi	Admin merumuskan hubungan *Role-Permission*	Composer memastikan package yang saling bergantung terinstall dengan benar. Admin memastikan hubungan antara jabatan (*Role*) dan tugasnya (*Permission*)

3. Langkah Kerja *Setup Database & Model User* & Menyiapkan Data Pokok (Nama, Jabatan, Peran)

Langkah ini menggabungkan dengan beberapa langkah dibawahnya karena hampir sama dan jalan seiringan, yaitu antara lain *Publish Migration Spatie* dan Persiapan Materi dan Infrastruktur, *Run Migration* dan Pelaksanaan PKM, Konfigurasi *Basic Spatie* dan Implementasi Sistem, Buat *Role* dan *Permission* dan Pembagian *Role* dan Personal Pengelola Admin Sekolah, Assign *Role* ke User dan Pendampingan Admin Sekolah dimana menghubungkan dengan sistem peran (RBAC) yang telah kita rencanakan.

a. Dari Sisi Teknis Laravel *Setup Database & Model User*

Tujuan kita adalah memastikan model User siap untuk dihubungkan dengan sistem role dan permission dari Spatie.

1. Prasyarat (Pre-requisites):
   i. *Package Spatie* sudah terinstall (Langkah 2).
   ii. Database telah dibuat dan dikonfigurasi di file .env.
2. Modifikasi *Model User* (app/Models/User.php):
   i. Model User sekarang bisa memiliki *roles* dan *permissions*.
   ii. Edit file app/Models/User.php dan menambahkan *HasRoles trait* dari *Package Spatie.*

```
<?php
namespace App\Models;
// use Illuminate\Contracts\Auth\MustVerifyEmail;
use Illuminate\Database\Eloquent\Factories\HasFactory;
use Illuminate\Foundation\Auth\User as Authenticatable;
use Illuminate\Notifications\Notifiable;
use Spatie\Permission\Traits\HasRoles; // <-- IMPOR TRAIT INI
class User extends Authenticatable
{
    use HasFactory, Notifiable, HasRoles; // <-- GUNAKAN TRAIT INI
    // ... (rest of the model code remains the same)
    protected $fillable = [
        'name',
        'email',
        'password',
    ];
    protected $hidden = [
        'password',
        'remember_token',
    ];
    protected function casts(): array
    {
        return [
            'email_verified_at' => 'datetime',
            'password' => 'hashed',
        ];
    }
}
```

Trait HasRoles menambahkan method seperti *assignRole*(), *hasRole*(), dan *can*() ke dalam model User, sehingga nanti berbentuk seperti ini:

```
$user->assignRole('guru');
if($user->hasRole('kepala_sekolah')) { ... }
if($user->can('input_nilai')) { ... }
```

3. Modifikasi *Model User* (app/Models/User.php):
   Migrasi. melakukan migrasi tabel users yang belum ada di database, *php artisan migrate*. Perintah ini akan menjalankan semua file migrasi, termasuk yang membuat tabel users, password_reset_tokens, dll.

b. Dari Sisi PKM Admin Sekolah *Setup Database & Model User*

Menyiapkan Data Pokok (Nama, Jabatan, Peran) dalam proses mengumpulkan dan mengorganisir data seluruh user sekolah ke dalam format yang siap dimasukkan ke dalam sistem.

1. Prasyarat (Pre-requisites):
   i. Dokumen Output dari Langkah 2: Matrix RBAC yang sudah final.
   ii. Sumber Data: Data dari Excel, arsip kartu pegawai, atau database lama.
2. Menjalankan "Perintah" Penyiapan Data (Kegiatan Administratif):
   i. Mengumpulkan Data dalam Satu File Master: Buat sebuah file Excel atau Google Sheet dengan kolom-kolom berikut:

Tabel 3. Format Role dan Peran yang Disiapkan untuk Datasebe

No	*Nama Lengkap*	*Email*	*Jabatan (Role)*	*Peran Khusus*	*Username*	*Password Awal*
1	Nama Kepsek, S.Pd	kepsek@sekolah.sch.id	Kepala Sekolah	-	kepsek	**123*** **
2	Wali Kelas, S.Pd	walas@sekolah.sch.id	Guru	Wali Kelas X	walas	**123*** **
3	Staf Admin, S.Kom	stafadmin@sekolah.sch.id	Staff Tata Usaha	Admin Lab Komputer	staf	**123*** **

Keterangan:

1. Email & Username: Harus unik untuk setiap orang. Ini akan menjadi kunci mereka untuk login.
2. Jabatan (Role): Harus sesuai dengan Daftar *Role* yang telah dibuat di Matrix RBAC (misal, "Guru", "Staff TU", "Kepala Sekolah"). *Satu orang bisa memiliki lebih dari satu role, dipisah koma.*
3. Password Awal: Buat yang sederhana dan sama untuk semua akun awal. Nanti user akan diminta menggantinya saat pertama login. Ini adalah praktik standar.

3. Konfigurasi Data (Quality Control):
   ii. Validasi Data: *Cross-check* data yang telah dikumpulkan dengan arsip yang ada. Pastikan tidak ada nama yang typo atau jabatan yang keliru.
   iii. Hubungkan dengan Matrix RBAC: Pastikan setiap Jabatan (*Role*) yang tercantum di sheet sudah terdaftar di Matrix RBAC Anda. Jika ada yang baru, tambahkan ke dalam matrix.
   iv. File Master ini adalah sumber kebenaran (*single source of truth*) yang nantinya akan digunakan untuk mengisi database aplikasi, baik secara manual maupun via impor

Tabel 4. Keterkaitan Langkah Teknis Laravel *Setup Database & Model User* dengan Keterkaitan Nyata pada PKM

Laravel Action	**PKM *Admin Action***	**Keterkaitan dengan PKM**
use Spatie\Permission\Traits\HasRoles;	Membuat Kolom "Jabatan (*Role*)" di Sheet	*HasRoles* menambahkan *capability* pada model User untuk memiliki peran. Kolom "Jabatan" adalah tempat mendefinisikan peran apa saja yang akan dimiliki oleh user tersebut. Keduanya adalah definisi struktur.
Model User dengan *trait HasRoles*	File Master Data Pokok	Model User adalah "blueprint" atau form yang harus diisi untuk setiap user di database. File Master adalah kumpulan data yang akan mengisi "form" tersebut untuk

Laravel Action	PKM *Admin Action*	Keterkaitan dengan PKM
		setiap orang. Model membutuhkan data, data membutuhkan model.
php artisan migrate (membuat tabel)	Mengumpulkan & memvalidasi data	Migrasi menyiapkan "ruangan" dan "lemari arsip" (tabel) yang kosong di database. Mengumpulkan data adalah kegiatan mengisi formulir yang akan disimpan ke dalam "lemari arsip" tersebut.

4. Langkah Kerja *Implementasi Middleware & Proteksi Routes* & Pembuatan Mekanisme Pengawasan dan Kontrol

Langkah ini adalah tentang menerapkan aturan yang telah kita desain (Matrix RBAC) ke dalam kode, sehingga sistem bisa secara otomatis mengizinkan atau menolak akses.

a. Dari Sisi Teknis Laravel Implementasi *Middleware* & *Proteksi Routes*

Middleware adalah penjaga yang memeriksa tiap request sebelum sampai ke Controller.

1. Prasyarat (Pre-requisites):
   i. Model User sudah menggunakan HasRoles (Langkah 3).
   ii. *Role* dan *Permission* sudah dibuat di database (Langkah 3).
2. Mendaftarkan *Middleware Spatie* (di app/Http/Kernel.php):
   i. Agar bisa menggunakan *middleware role* dan *permission* di *routes*, kita harus mendaftarkan alias-nya terlebih dahulu.
   ii. Dengan membuka file *app/Http/Kernel.php* dan cari property $middlewareAliases. dan menambahkan dua baris berikut:

```
protected $middlewareAliases = [
    // ... middleware lainnya yang sudah ada
    'auth' => \App\Http\Middleware\Authenticate::class,
    'auth.basic' => \Illuminate\Auth\Middleware\AuthenticateWithBasicAuth::class,
    // ... Tambahkan ini:
    'role' => \Spatie\Permission\Middleware\RoleMiddleware::class,
    'permission' => \Spatie\Permission\Middleware\PermissionMiddleware::class,
    'role_or_permission' => \Spatie\Permission\Middleware\RoleOrPermissionMiddleware::class,
];
```

3. Proteksi Routes (di routes/web.php):
   i. Ini adalah implementasi nyata dari Matrix RBAC. Kita terapkan aturan akses ke setiap URL.
   ii. Gunakan Middleware secara Langsung:

```
// Semua route dalam group ini HANYA bisa diakses oleh user yang sudah login ('auth')
Route::middleware(['auth'])->group(function () {
    // Group untuk role 'guru'
    Route::middleware(['role:guru'])->group(function () {
        Route::get('/guru/dashboard', [GuruController::class, 'dashboard']);
        Route::get('/guru/nilai', [GuruController::class, 'nilaiIndex']);
    });
    // Group untuk role 'siswa'
    Route::middleware(['role:siswa'])->group(function () {
        Route::get('/siswa/dashboard', [SiswaController::class, 'dashboard']);
        Route::get('/siswa/nilai-saya', [SiswaController::class, 'nilaiSaya']);
    });
    // Route untuk banyak role sekaligus
    Route::middleware(['role:admin|kepala_sekolah'])->group(function () {
        Route::get('/admin/laporan', [AdminController::class, 'laporan']);
    });
});
```

4. Proteksi di Controller (Opsional Tambahan):
   i. Memeriksa izin langsung di dalam Controller untuk logika yang lebih kompleks.

```
public function edit(Article $article)
{
    // User harus memiliki permission 'edit_article' untuk melanjutkan
    $this->authorize('edit_article');
    // Atau, gunakan cara manual jika perlu logika custom
    if (!auth()->user()->can('edit_article')) {
        abort(403, 'Unauthorized action.');
    }
    return view('article.edit', compact('article'));
}
```

b. Dari Sisi PKM Admin Sekolah Pembuatan Mekanisme Pengawasan dan Kontrol

Pembuatan Mekanisme Pengawasan dan Kontrol adalah tentang menerapkan aturan fisik dan prosedural berdasarkan Matrix RBAC.

1. Prasyarat (Pre-requisites):
   i. Matrix RBAC yang sudah final (Langkah 2).
   ii. Data Pokok pengguna beserta role-nya (Langkah 3).
2. Menjalankan Perintah Pembuatan Mekanisme (Penerapan Prosedur):
   i. Buat Prosedur Login dan Akses:
      a) Setiap staff harus login dengan username dan password unik mereka ke dalam sistem komputer.
      b) Dilarang keras berbagi password atau meninggalkan komputer dalam keadaan logged-in.
   ii. Buat Aturan Akses Fisik Berdasarkan Role:
      a) Hanya Staff Tata Usaha dan Kepala Sekolah yang memiliki kunci akses ke ruang server."
      b) File laporan keuangan disimpan dalam lemari berkunci. Hanya Bendahara dan Kepala Sekolah yang memegang kuncinya."
   iii. Buat Alur Persetujuan (*Workflow*):
      a) Sebuah Surat Keluar harus dibuat oleh Staff TU, kemudian disetujui (*permission approve*_surat) oleh Kepala Sekolah sebelum bisa dicetak dan distempel.
      b) Pengajuan pembelian barang oleh Guru harus mendapatkan persetujuan dari Wakil Kepala Sekolah terlebih dahulu."
3. Implementasi Pengawasan (Pemantauan):
   i. Audit Log: "Seluruh aktivitas login dan pengubahan data penting (seperti nilai dan keuangan) harus tercatat dalam log buku/system log untuk bisa ditelusuri jika terjadi masalah."
   ii. Review Berkala: "Setiap semester, Admin dan Kepala Sekolah melakukan review terhadap role dan permission setiap user. Apakah masih sesuai dengan jabatannya? Apakah ada permission yang perlu ditambah atau dicabut?"

Tabel 5. Implementasi Middleware & Proteksi Routes & Pembuatan Mekanisme Pengawasan dan Kontrol

Laravel Action	**PKM *Admin Action***	**Keterkaitan dengan PKM**
->middleware(['role:admin'])	"Hanya Staff TU dan Kepsek yang punya kunci ruang server"	Keduanya adalah pembatasan akses berdasarkan peran. Yang satu di dunia digital, yang lain di dunia fisik. Intinya: "Jika bukan peran X, Anda tidak boleh masuk."
abort(403, 'Unauthorized')	Prosedur persetujuan surat oleh Kepala Sekolah	Keduanya adalah mekanisme pengecekan ulang (*double-check*). Kode akan menolak akses yang tidak memenuhi syarat. Prosedur menolak surat yang belum disetujui oleh pihak yang berwenang.
Route Groups	Pembagian area kerja fisik	*Route Group* mengelompokkan route yang bisa diakses oleh role tertentu. Pembagian area kerja (e.g., ruang guru, ruang TU) mengelompokkan orang berdasarkan role mereka di dunia nyata.

HASIL DAN PEMBAHASAN

Pengabdian masyarakat ini bertujuan untuk meningkatkan kapasitas administrasi dan keamanan sistem informasi di sekolah melalui implementasi Role-Based Access Control (RBAC) menggunakan framework Laravel dan *Package Spatie*. Metode pelaksanaannya dilakukan melalui beberapa tahapan, yaitu **(1) Identifikasi Kebutuhan dan Pelatihan Konsep RBAC, (2) Implementasi Teknis,** dan **(3) Testing dan Evaluasi.** Bagian ini akan membahas hasil dari setiap tahapan tersebut.

a. Identifikasi Kebutuhan, Analisis Sistem dan Pelatihan Konsep RBAC

Tahap awal pengabdian dilakukan dengan melakukan observasi dan wawancara dengan para stakeholder, termasuk Kepala Sekolah, guru, dan staff tata usaha. Hasil identifikasi menunjukkan beberapa kebutuhan utama:

1. Sistem yang dapat mengelola data siswa, guru, dan nilai secara terintegrasi.
2. Pembatasan akses yang jelas berdasarkan jabatan untuk mencegah kesalahan penginputan dan kebocoran data.
3. Prosedur yang terdokumentasi untuk alur persetujuan dan audit trail. Berdasarkan kebutuhan tersebut, disusunlah sebuah matriks RBAC yang menjadi blueprint bagi implementasi teknis. Matriks tersebut merepresentasikan hubungan antara peran (role) dan izin (permission) di lingkungan sekolah.

Tabel 6. Matriks RBAC untuk Sistem Informasi Sekolah

Peran (Role)	Izin (Permission)	Deskripsi Akses
Kepala Sekolah	*view_reports, approve_budget*	Dapat melihat semua laporan dan menyetujui anggaran
Wali Kelas	*input_scores, view_attendance*	Dapat menginput nilai dan melihat absensi kelasnya
Guru Mata Pelajaran	*input_scores*	Hanya dapat menginput nilai untuk mapel yang diampu
Staff Tata Usaha	*manage_students_data, print_reports*	Dapat mengelola data siswa dan mencetak laporan
Admin Sistem	*manage_users, assign_roles*	Dapat mengelola semua user dan menetapkan peran

b. Implementasi Teknis Laravel Spatie

Implementasi teknis dilakukan sesuai dengan metodologi pengembangan perangkat lunak yang terstruktur. Hasil dari proses coding dan konfigurasi adalah sebagai berikut:

1. Setup Environment dan Database: Project Laravel berhasil diinisialisasi dan terkoneksi dengan database MySQL. Model User telah dimodifikasi dengan menambahkan trait HasRoles dari *Package Spatie*.

```
use Spatie\Permission\Traits\HasRoles;

class User extends Authenticatable
{
    use HasFactory, Notifiable, HasRoles;
    // ... kode lainnya
}
```

2. Proteksi Route dengan Middleware: Berdasarkan matriks RBAC, seluruh route pada aplikasi telah diproteksi dengan middleware Spatie. Implementasi ini memastikan bahwa aturan bisnis yang telah dirancang diterapkan secara konsisten pada level akses URL.

```
// Hanya Admin yang boleh mengelola user
Route::group(['middleware' => ['role:admin']], function () {
    Route::resource('users', UserController::class);
});
// Guru dan Wali Kelas boleh input nilai
Route::group(['middleware' => ['permission:input_scores']], function () {
    Route::resource('scores', ScoreController::class);
});
```

3. Keamanan pada Layer Presentasi (Blade Template): Untuk meningkatkan user experience, implementasi juga dilakukan pada tampilan agar menu atau tombol yang tidak diizinkan untuk suatu role tidak ditampilkan sama sekali.

```
@can('manage_users')
    <a href="{{ route('users.index') }}" class="btn btn-primary">Kelola User</a>
@endcan
```

c. Hasil Implementasi dan Pengujian

Setelah implementasi selesai, dilakukan pengujian untuk memvalidasi keefektifan sistem. Pengujian dilakukan dengan dua metode:

1. Pengujian Fungsional (*Functional Testing*): Setiap *role* pengguna diuji untuk mengakses berbagai fitur. Hasilnya, sistem berhasil membatasi akses sesuai dengan yang telah dikonfigurasi dalam matriks RBAC. Sebagai contoh, user dengan role guru tidak dapat mengakses halaman untuk mengelola data user (/*users*) dan akan dialihkan ke halaman "403 Forbidden".
2. Pengujian Keamanan (Security Testing): Dilakukan percobaan untuk mengakses URL secara langsung (manual testing) dengan akun yang tidak memiliki permission. Sistem secara konsisten menolak akses dan mengembalikan kode error HTTP 403, yang membuktikan bahwa middleware bekerja dengan benar.

Berdasarkan hasil pengujian, dapat disimpulkan bahwa implementasi Laravel Spatie RBAC telah berhasil:

1. Menerapkan prinsip *least privilege*, dimana setiap user hanya memiliki akses yang benar-benar diperlukan.
2. Menciptakan sistem yang lebih aman dari penyalahgunaan akses, baik secara sengaja maupun tidak sengaja.
3. Memberikan struktur yang terorganisir dan mudah dikembangkan untuk menambahkan peran atau izin baru di masa depan.

d. Pengujian Akses ke Route yang Diproteksi

Pengujian ini mensimulasikan upaya akses oleh berbagai peran (role) ke route atau URL yang sensitif. Hasilnya dirangkum dalam tabel berikut:

Tabel 7. Perbandingan Hasil Pengujian Akses Route Sebelum dan Sesudah Implementasi

URL yang Diuji	Peran Penguji	Izin yang Diperlukan	Hasil Sebelum Implementasi	Hasil Sesudah Implementasi	Keterangan
/admin/users	**Administrator**	*manage_users*	**200 OK** (Akses Diberikan)	**200 OK** (Akses Diberikan)	Admin tetap memiliki akses penuh sesuai tugasnya.
/admin/users	**Guru**	*manage_users*	**200 OK** (Akses Diberikan)*	**403 Forbidden** (Akses Ditolak)	**Peningkatan Kritis.** Sebelumnya, guru bisa mengakses fitur berbahaya yang bukan wewenangnya.
/admin/users	**Staff Admin**	*manage_users*	**200 OK** (Akses Diberikan)*	**403 Forbidden** (Akses Ditolak)	**Peningkatan Kritis.** Sebelumnya, Staff Admin bisa mengakses fitur berbahaya yang bukan wewenangnya.

Keterangan Tabel: Hasil dengan tanda asterisk (*) menandakan **kerentanan keamanan** yang ada pada sistem sebelum implementasi.

e. Pengujian Kerentanan Privilege Escalation

Pengujian ini bertujuan untuk mengetahui apakah seorang user dengan hak akses rendah dapat mempromosikan dirinya sendiri atau mengakses fitur yang diperuntukkan bagi super user.

Tabel 7. Perbandingan Hasil Pengujian Privilege Escalation

Skenario Pengujian	Hasil Sebelum Implementasi	Hasil Sesudah Implementasi	Analisis dan Pembahasan
***Vertical Escalation*:** User siswa mencoba mengakses URL /admin/dashboard secara langsung via browser.	**200 OK** (Akses Berhasil)*	**403 Forbidden** (Akses Ditolak)	**Peningkatan drastis.** Middleware role:admin secara otomatis memblokir semua percobaan akses oleh peran non-admin ke route admin.

Skenario Pengujian	Hasil Sebelum Implementasi	Hasil Sesudah Implementasi	Analisis dan Pembahasan
***Horizontal Escalation*:** User siswa_A mencoba mengakses halaman profil siswa_B dengan memanipulasi parameter ID pada URL (e.g., /siswa/profile/15).	**200 OK** (Data profil siswa_B ditampilkan)*	**403 Forbidden** (Akses Ditolak)	**Peningkatan signifikan.** Pengecekan authorization di Controller memastikan user hanya dapat mengakses data miliknya sendiri (profile->id == auth->id).
***Privilege Retention*:** User dengan role guru yang kemudian role-nya diturunkan menjadi siswa mencoba akses route /guru/nilai.	**200 OK** (Akses Masih Diberikan)*	**403 Forbidden** (Akses Ditolak)	**Peningkatan pada manajemen sesi.** Sistem sekarang secara real-time memvalidasi permission user terhadap database. Perubahan permission berlaku immediately setelah logout-login.

f. Pembahasan Hasil Pengujian:

Berdasarkan data pada Tabel 6 dan Tabel 7, terlihat bahwa implementasi *Laravel Spatie* RBAC berhasil mentransformasi keamanan sistem informasi sekolah dari kondisi yang sangat rentan menjadi terlindungi dengan sangat baik.

1. Eliminasi Kerentanan Mayor: Sebelum implementasi, sistem tidak memiliki mekanisme pembatasan akses yang memadai. Setiap user yang berhasil login (*authenticated*) dapat mengakses hampir semua fitur (200 OK pada kolom *Sebelum*), yang merupakan risiko keamanan yang sangat tinggi. Setelah implementasi, semua percobaan akses illegal secara konsisten ditolak dengan kode *403 Forbidden*.
2. Penerapan Prinsip Least Privilege yang Konsisten: Sistem sekarang secara ketat menerapkan prinsip ini. Seorang guru hanya bisa melakukan *input_nilai*, seorang bendahara hanya bisa *view_financial_reports*, dan seorang siswa hanya bisa melihat data miliknya sendiri. Tidak ada lagi tumpang tindih akses yang berpotensi menyebabkan kesalahan atau penyalahgunaan data.
3. Pertahanan Berlapis (*Defense in Depth*): Keamanan tidak hanya bergantung pada satu lapis. Proteksi dilakukan pada tiga level: Middleware (penjaga gerbang utama), *Controller* (penjaga logika bisnis), dan *Blade Directive* (UI/UX yang aman). Hal ini membuat sistem menjadi sangat *robust* dan sulit untuk ditembus.

Dengan demikian, dapat disimpulkan bahwa tujuan pengabdian untuk menciptakan sistem informasi sekolah yang aman, terpercaya, dan teraudit telah tercapai

KESIMPULAN

Berdasarkan seluruh rangkaian kegiatan pengabdian masyarakat, dapat disimpulkan bahwa implementasi *Role-Based Access Control* (RBAC) menggunakan *framework* Laravel dan *Package Spatie* telah berhasil mengatasi permasalahan utama yang dihadapi oleh sekolah, yaitu lemahnya keamanan dan tata kelola akses dalam sistem informasi yang ada. Awalnya, sistem berada dalam kondisi yang sangat rentan dimana hampir semua pengguna yang terautentikasi dapat mengakses data dan fitur yang bukan menjadi kewenangannya, sehingga berpotensi menimbulkan kesalahan manipulasi data, kebocoran informasi sensitif, dan tidak adanya audit trail yang jelas. Melalui metode yang sistematis, mulai dari identifikasi kebutuhan, perancangan matriks RBAC, implementasi teknis, hingga pengujian yang baik, kegiatan ini berhasil mentransformasi sistem tersebut menjadi sebuah platform yang aman, terstruktur, dan *accountable*.

Hasil pengujian membuktikan bahwa setelah implementasi, semua percobaan akses ilegal secara konsisten ditolak oleh sistem, yang menunjukkan bahwa prinsip least privilege telah diterapkan dengan efektif. Setiap peran pengguna—mulai dari admin, kepala sekolah, guru, hingga siswa, staf admin—kini hanya dapat mengakses dan melakukan operasi data yang secara spesifik telah diizinkan bagi mereka, sesuai dengan tugas dan tanggung jawabnya di lingkungan sekolah. Selain aspek keamanan, solusi ini juga memberikan dampak positif terhadap efisiensi administratif dengan memberikan kejelasan wewenang dan meminimalisir duplikasi atau tumpang tindih dalam pengelolaan data. Dengan demikian, kegiatan pengabdian ini tidak hanya menyelesaikan permasalahan teknis semata, tetapi juga turut serta dalam mendukung terwujudnya tata kelola sistem informasi yang lebih baik (*good IT governance*) di institusi pendidikan, yang pada akhirnya akan meningkatkan kualitas layanan pendidikan secara keseluruhan.

.

UCAPAN TERIMA KASIH

Kami mengucapkan terima kasih yang sebesar-besarnya kepada semua pihak yang telah mendukung dan berkontribusi dalam kesuksesan kegiatan pengabdian masyarakat ini. Ucapan terima kasih yang tulus kami sampaikan kepada:

Universitas PGRI Sumatera Barat yang telah memberikan kepercayaan, fasilitas, dan dukungan moril maupun material sehingga kegiatan pengabdian ini dapat terlaksana dengan lancar. Dukungan dari institusi sangatlah vital dalam menghubungkan tri dharma perguruan tinggi dengan kebutuhan masyarakat secara nyata.

Kepada Kepala Sekolah, Dewan Guru, Staf Tata Usaha, dan seluruh keluarga besar SMAN 4 Sumatera Barat, kami menyampaikan penghargaan dan terima kasih yang setinggi-tingginya. Atas keterbukaan, kerjasama, kepercayaan, serta partisipasi aktif yang diberikan sejak dari tahap identifikasi kebutuhan hingga proses implementasi dan testing, kami tidak dapat membayangkan keberhasilan ini dapat tercapai tanpa dukungan penuh dari Bapak dan Ibu sekalian.

Terima kasih juga kami ucapkan kepada seluruh tim pelaksana dan mahasiswa yang telah mencurahkan tenaga, pikiran, dan waktunya dengan penuh dedikasi untuk mewujudkan sistem ini. Akhir kata, kami menyadari bahwa masih banyak kekurangan dalam pelaksanaan kegiatan ini. Untuk itu, kami senantiasa terbuka terhadap segala bentuk kritik dan saran yang membangun untuk perbaikan di masa yang akan datang. Semoga hasil dari pengabdian ini dapat memberikan manfaat yang berkelanjutan bagi kemajuan bersama..

DAFTAR PUSTAKA

Aabid, M., Dzaky, T., & Fikma Edrisy, I. (2025). *Strategi Pencegahan Kejahatan Siber di Indonesia: Sinergi antara UU ITE dan Kebijakan Keamanan Digital* (Vol. 4, Issue 2).

Almeida, F., Duarte Santos, J., & Augusto Monteiro, J. (2020). The Challenges and Opportunities in the Digitalization of Companies in a Post-COVID-19 World. *IEEE Engineering Management Review*, *48*(3), 97–103. https://doi.org/10.1109/EMR.2020.3013206

Amirulloh, M., Handayani, T., & Viero Sadam, A. (2025). Keamanan Siber (Cybersecurity) Pada Sistem Perbankan Digital Di Indonesia Berdasarkan Hukum Siber Indonesia. *Jurnal Inovasi Global*, *3*(5). https://jig.rivierapublishing.id/index.php/rv/index

Donalds, C., Barclay, C., & Osei-Bryson, K.-M. (2022). Cybercrime and Cybersecurity in the Global South. In *Cybercrime and Cybersecurity in the Global South*. Routledge. https://doi.org/10.1201/9781003028710

Isnaini, K. N., Sulistiyani, D. F., & Sutrisno, M. (2020). JPMB: Jurnal Pemberdayaan Masyarakat Berkarakter Data Security Awareness sebagai Upaya Peningkatan Literasi Tentang Cyber Attacks dan Threats. *JPMB*, *3*(2), 121–132.

Rizki Sandrina Ayu. (2025). Keamanan Infrastruktur Teknologi Informasi: Analisis Ancaman Siber dan Pendekatan Mitigasi. *Jurnal Pendidikan Sosial Dan Humaniora*, *4*(2), 2598–2609.

Setiawan, R. (2025). *Digitalisasi Perbankan dan Ancaman Keamanan Siber: Tantangan dan Strategi Mitigasi Risiko Operasional*. *1*. https://doi.org/10.123456/asefba.v1i1.xxxx

Sintia Saramuke, S., Antoneta Putri, V., Marianti Sormin, A., & Nugraha, M. (2025). Ancaman Keamanan Siber dan Peran Aktor Non-Negara di Dunia Digital. *Syntax Idea*, *6*(02).

Stauffer, M. (2019). *Laravel Up & Running A Framework for Building Modern PHP Apps*. www.EBooksWorld.ir

Wahid, R. A., Nadim, M. S. N., Sulaiman, S., Shaharudin, S. A., Jupikil, M. D., & Su, I. J. S. A. (2025). *Utilizing Composer Packages to Accelerate Laravel-Based Project Development Among Students: A Pedagogical and Practical Framework*. http://arxiv.org/abs/2508.05747

Welnof Satria;dkk. (2025). *SOSIALISASI KEAMANAN SIBER BAGI PELAJAR DALAM MENGHADAPI ERA DIGITAL* (Vol. 2, Issue 1). https://jurnal.ananpublisher.com/index.php/abdidalem

Yuricha, Y., & Phan, I. K. (2023a). Penerapan Role Based Access Control dalam Sistem Supply Chain Management Berbasis Cloud. *MALCOM: Indonesian Journal of Machine Learning and Computer Science*, *3*(2), 339–348. https://doi.org/10.57152/malcom.v3i2.1259

Yuricha, Y., & Phan, I. K. (2023b). Penerapan Role Based Access Control dalam Sistem Supply Chain Management Berbasis Cloud. *MALCOM: Indonesian Journal of Machine Learning and Computer Science*, *3*(2), 339–348. https://doi.org/10.57152/malcom.v3i2.1259