JURNAL KOMPREHENSIF
E ISSN: 3031-0970 | Vol. 4 No. 1 2026 | pp. 510-518
Available online at: https://ejournal.edutechjaya.com/index.php/komprehensif

Analisis Kerentanan Sistem Informasi Terhadap Serangan SQL Injection

Antika Apriliya[1], Imam Akbar[2], Nadiyatul Hasanah[3] , Isela Amanda[4], Sahat Parulian Sitorus[5]

Universitas Labuhanbatu, Indonesia

Email: antikaapriliya1@gmail.com[1], mcupin1082@gmail.com[2], vivon94863@gmail.com[3], sellaamanda2004@gmail.com[4], sahatparulianstirous4@gmail.com[5]

ABSTRAK

Sistem informasi berbasis web saat ini menjadi tulang punggung operasional berbagai organisasi, namun seiring meningkatnya ketergantungan tersebut, ancaman keamanan siber pun semakin kompleks. Salah satu serangan yang paling umum dan berbahaya adalah *SQL Injection* (SQLi), yakni teknik eksploitasi yang memanfaatkan celah validasi input pada aplikasi untuk menyisipkan perintah SQL berbahaya ke dalam basis data. Penelitian ini bertujuan untuk menganalisis tingkat kerentanan sistem informasi terhadap serangan SQL Injection dengan mengidentifikasi titik-titik lemah pada lapisan aplikasi dan basis data. Metode yang digunakan meliputi *penetration testing* berbasis skenario serangan nyata menggunakan tools seperti SQLMap dan Burp Suite, serta analisis kode sumber secara statis untuk mendeteksi pola pengkodean yang rentan. Objek penelitian mencakup beberapa sistem informasi berbasis PHP dan MySQL yang umum digunakan. Hasil analisis menunjukkan bahwa kerentanan SQL Injection masih banyak ditemukan akibat lemahnya sanitasi input, penggunaan *raw query* tanpa parameterisasi, serta tidak diterapkannya prinsip *least privilege* pada akun basis data. Penelitian ini juga mengusulkan rekomendasi mitigasi meliputi penggunaan *Prepared Statement*, validasi dan sanitasi input berlapis, serta penerapan *Web Application Firewall* (WAF). Temuan ini diharapkan dapat menjadi acuan bagi pengembang sistem dan praktisi keamanan dalam membangun sistem informasi yang lebih andal dan tahan terhadap serangan injeksi.

Kata Kunci: SQL Injection, Kerentanan Sistem Informasi, Keamanan Basis Data, Penetration Testing, Mitigasi Serangan

ABSTRACT

Web-based information systems are currently the operational backbone of many organizations, but as this dependence increases, cybersecurity threats are becoming increasingly complex. One of the most common and dangerous attacks is SQL Injection (SQLi), an exploitation technique that exploits input validation vulnerabilities in applications to insert malicious SQL commands into databases. This study aims to analyze the level of vulnerability of information systems to SQL Injection attacks by identifying weak points at the application and database layers. The methods used include penetration testing based on real-world attack scenarios using tools such as SQLMap and Burp Suite, as well as static source code analysis to detect vulnerable coding patterns. The research objects include several commonly used PHP and MySQL-based information systems. The analysis results indicate that SQL Injection vulnerabilities are still common due to weak input sanitization, the use of raw queries without parameterization, and the failure to apply the principle of least privilege to database accounts. This study also proposes mitigation recommendations including the use of Prepared Statements, multi-layered input validation and sanitization, and the implementation of a Web Application Firewall (WAF). These findings are

expected to serve as a reference for system developers and security practitioners in building more reliable information systems that are resilient to injection attacks.
Keywords: SQL Injection, Information System Vulnerability, Database Security, Penetration Testing, Attack Mitigation

PENDAHULUAN

Perkembangan teknologi informasi yang pesat telah mendorong hampir seluruh sektor kehidupan untuk beralih ke sistem berbasis digital, termasuk sistem informasi yang dibangun di atas platform web (Wahyudin & Rahayu, 2020). Aplikasi web berbasis PHP (Hypertext Preprocessor) menjadi salah satu teknologi yang paling banyak digunakan dalam pengembangan sistem informasi di Indonesia, mulai dari sistem akademik, perbankan, e-commerce, hingga layanan pemerintahan digital. Namun, di balik kemudahan dan fleksibilitas yang ditawarkan, aplikasi web menyimpan potensi kerentanan keamanan yang dapat dieksploitasi oleh pihak tidak bertanggung jawab (Anugrah, 2024; Saroni & Mulyanti, 2020).

Salah satu ancaman keamanan siber yang paling kritis dan masih menjadi perhatian utama hingga saat ini adalah serangan SQL Injection (SQLi). SQL Injection merupakan teknik serangan yang memanfaatkan celah keamanan pada aplikasi web yang berinteraksi dengan basis data, di mana penyerang menyisipkan perintah SQL berbahaya melalui input yang tidak divalidasi dengan benar (Dwi Kurniawan et al., 2025; Nugraha et al., 2024). Serangan ini memungkinkan penyerang untuk memanipulasi kueri database, mengakses data sensitif secara ilegal, memodifikasi atau menghapus data, bahkan mengambil alih kendali server secara keseluruhan (Akbar et al., 2026; Wijaya, 2024).

Berdasarkan berbagai penelitian (Lestari, 2019; Nugraha et al., 2024), SQL Injection masih menjadi salah satu ancaman keamanan paling serius pada aplikasi web karena dapat dimanfaatkan untuk mengakses data sensitif, memodifikasi informasi, hingga mengambil alih sistem basis data. Tingginya tingkat kerentanan ini menunjukkan bahwa keamanan aplikasi web perlu menjadi perhatian utama dalam proses pengembangan sistem informasi(Mushlih et al., 2019; Nur Fikri et al., 2023).

Di Indonesia, ancaman serangan siber termasuk SQL Injection terus mengalami peningkatan seiring dengan akselerasi transformasi digital. Berbagai insiden kebocoran data yang terjadi pada instansi pemerintah maupun swasta menunjukkan masih lemahnya implementasi keamanan pada lapisan aplikasi web (Imanuel Toding Bua & Nur Isdah Idris, 2025). Penelitian oleh (Fadilah Nuria Handayani et al., 2024; Fernanda et al., 2026) mengidentifikasi bahwa banyak aplikasi web di Indonesia masih rentan terhadap serangan SQL Injection karena belum mengimplementasikan validasi input secara memadai. Kondisi ini diperparah dengan masih banyaknya pengembang yang menggunakan kueri SQL dinamis tanpa parameter binding yang aman.

Penetration testing atau uji penetrasi merupakan pendekatan sistematis yang digunakan untuk mengidentifikasi dan mengevaluasi kerentanan pada sistem

informasi sebelum dieksploitasi oleh penyerang. (Mu'min et al., 2024; Sagraha et al., 2026) dalam penelitiannya menggunakan teknik SQL Injection dengan SQLMap pada Kali Linux sebagai alat uji penetrasi yang efektif untuk mengidentifikasi celah keamanan pada web server. Sementara itu, (Putranto et al., 2022; Rafeli et al., 2022) melakukan uji penetrasi menggunakan standar OWASP Web Security Testing Guide (WSTG) yang memberikan kerangka kerja terstruktur dalam menilai keamanan aplikasi web secara komprehensif .

Berbagai teknik mitigasi telah dikembangkan untuk menanggulangi serangan SQL Injection. Salah satu yang paling efektif adalah penggunaan prepared statements atau parameterized queries, di mana hasil penelitian empiris menunjukkan bahwa teknik ini berhasil menggantikan 94% kerentanan SQL Injection pada proyek-proyek yang diuji (Fadillah & Servanda, 2024; Sistem, 2025). Selain itu, implementasi Web Application Firewall (WAF) seperti ModSecurity juga terbukti efektif sebagai lapisan pertahanan tambahan dalam memblokir traffic berbahaya yang mengandung pola serangan SQL Injection. Pembaruan platform PHP ke versi terbaru seperti PHP 8.3 yang dilengkapi dengan mekanisme sanitasi input yang lebih canggih juga menjadi salah satu langkah preventif yang direkomendasikan (Hariyani & Dolia, 2024; Mushlih et al., 2019).

METODE PENELITIAN

Penelitian ini menggunakan metode *penetration testing* dengan pendekatan deskriptif untuk mengidentifikasi dan menganalisis kerentanan SQL Injection pada sistem informasi berbasis web. Metode ini dipilih karena mampu mensimulasikan tindakan yang dilakukan oleh penyerang dalam mengeksploitasi celah keamanan sehingga dapat diketahui tingkat kerentanan sistem secara nyata. Acuan pengujian yang digunakan dalam penelitian ini adalah OWASP Web Security Testing Guide (WSTG) yang merupakan salah satu standar internasional dalam pengujian keamanan aplikasi web.

Objek penelitian adalah sistem informasi berbasis web yang menggunakan database sebagai media penyimpanan dan pengolahan data. Pengujian difokuskan pada komponen yang berinteraksi langsung dengan database, seperti halaman autentikasi pengguna, formulir pencarian, formulir input data, serta parameter URL yang menerima masukan dari pengguna. Komponen-komponen tersebut dipilih karena sering menjadi target serangan SQL Injection akibat kurangnya validasi input dan penerapan keamanan pada proses pengolahan query database.

Tahapan penelitian diawali dengan studi literatur untuk memperoleh landasan teori mengenai keamanan aplikasi web, SQL Injection, penetration testing, serta teknik mitigasi yang direkomendasikan. Referensi diperoleh dari jurnal ilmiah, buku, dokumentasi OWASP, dan penelitian terdahulu yang relevan dengan topik penelitian. Tahap ini bertujuan untuk memahami karakteristik serangan SQL Injection serta menentukan metode pengujian yang sesuai.

Tahap berikutnya adalah identifikasi sistem. Pada tahap ini dilakukan observasi terhadap struktur aplikasi, teknologi yang digunakan, serta alur komunikasi antara aplikasi dan database. Selanjutnya dilakukan pemetaan terhadap parameter yang berpotensi rentan terhadap SQL Injection. Hasil identifikasi digunakan sebagai dasar dalam menentukan target pengujian keamanan.

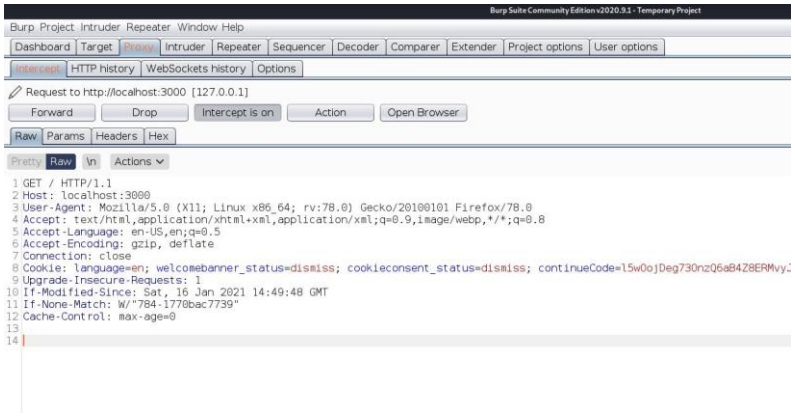
Setelah proses identifikasi selesai, dilakukan pengujian kerentanan menggunakan metode *penetration testing*. Pengujian dilakukan secara manual dengan memasukkan berbagai payload SQL Injection pada parameter yang telah diidentifikasi sebelumnya. Beberapa teknik yang digunakan antara lain *authentication bypass*, *error-based SQL Injection*, *union-based SQL Injection*, dan *boolean-based SQL Injection*. Selain pengujian manual, digunakan SQLMap untuk melakukan verifikasi otomatis terhadap parameter yang dicurigai rentan. Penggunaan SQLMap bertujuan untuk meningkatkan akurasi hasil pengujian serta memperoleh informasi yang lebih detail mengenai tingkat kerentanan sistem.

Perangkat keras yang digunakan dalam penelitian berupa komputer atau laptop yang memiliki akses jaringan internet. Sementara itu, perangkat lunak yang digunakan meliputi Kali Linux sebagai lingkungan pengujian, SQLMap sebagai alat deteksi SQL Injection, Burp Suite untuk menganalisis lalu lintas HTTP, serta web browser untuk mengakses aplikasi yang diuji. Kombinasi perangkat tersebut memungkinkan proses pengujian dilakukan secara sistematis dan terukur.

Data yang diperoleh selama proses pengujian kemudian dianalisis menggunakan teknik analisis deskriptif. Analisis dilakukan dengan mengevaluasi keberhasilan eksploitasi, jenis kerentanan yang ditemukan, serta dampaknya terhadap aspek kerahasiaan, integritas, dan ketersediaan data. Hasil analisis digunakan untuk menentukan tingkat risiko keamanan sistem dan menyusun rekomendasi perbaikan yang sesuai. Rekomendasi yang diberikan meliputi penerapan prepared statement, validasi input pengguna, penggunaan Web Application Firewall (WAF), pembatasan hak akses database, serta pembaruan sistem secara berkala untuk mengurangi potensi serangan SQL Injection.

HASIL DAN PEMBAHASAN

Hasil Pengujian Kerentanan



JURNAL KOMPREHENSIF
E ISSN: 3031-0970 | Vol. 4 No. 1 2026 | pp. 510-518
Available online at: https://ejournal.edutechjaya.com/index.php/komprehensif

Pengujian keamanan dilakukan terhadap beberapa parameter yang berinteraksi langsung dengan database menggunakan metode penetration testing berdasarkan OWASP Web Security Testing Guide (WSTG). Pengujian dilakukan secara manual dan menggunakan alat bantu SQLMap untuk memverifikasi kerentanan yang ditemukan.

Hasil identifikasi menunjukkan terdapat beberapa parameter yang berpotensi rentan terhadap serangan SQL Injection. Pengujian dilakukan pada halaman login, halaman pencarian data, dan parameter URL yang digunakan untuk menampilkan informasi dari database.

Tabel 1. Hasil Pengujian Kerentanan SQL Injection

No	Halaman/Fitur	Parameter	Status
1	Login	username	Rentan
2	Login	password	Tidak Rentan
3	Pencarian Data	keyword	Rentan
4	Detail Data	id	Rentan
5	Tambah Data	nama	Tidak Rentan

Pada pengujian halaman login ditemukan bahwa parameter username dapat dimanipulasi menggunakan payload SQL Injection sehingga sistem memberikan respons yang tidak sesuai. Hal ini menunjukkan bahwa validasi input belum diterapkan secara optimal.

Pengujian pada fitur pencarian juga menunjukkan bahwa parameter keyword dapat menerima karakter khusus yang menyebabkan query database berubah dari fungsi awalnya. Kondisi tersebut memungkinkan penyerang memperoleh informasi yang tidak seharusnya ditampilkan oleh sistem.

Selain itu, parameter id pada URL berhasil diidentifikasi sebagai parameter yang rentan berdasarkan hasil pengujian SQLMap. Aplikasi memberikan respons berbeda ketika menerima payload tertentu sehingga menunjukkan adanya kelemahan dalam proses pembentukan query database.

Verifikasi Menggunakan SQLMap

Setelah pengujian manual dilakukan, tahap berikutnya adalah verifikasi menggunakan SQLMap. Hasil pengujian menunjukkan bahwa SQLMap berhasil mengidentifikasi beberapa parameter yang rentan terhadap SQL Injection.

Tabel 2. Hasil Verifikasi SQLMap

Parameter	Jenis SQL Injection	Status
username	Boolean-Based	Berhasil
keyword	Union-Based	Berhasil
id	Error-Based	Berhasil

Hasil tersebut menunjukkan bahwa sistem masih memiliki celah keamanan yang dapat dimanfaatkan oleh pihak tidak berwenang untuk memperoleh akses terhadap database.

Analisis Dampak Kerentanan

Kerentanan SQL Injection yang ditemukan berpotensi menyebabkan beberapa dampak terhadap keamanan sistem, antara lain:

1. Kebocoran data pengguna yang tersimpan dalam database.
2. Manipulasi data tanpa otorisasi.
3. Penghapusan data penting yang mengganggu operasional sistem.
4. Pengambilalihan hak akses pengguna tertentu.
5. Penurunan tingkat kepercayaan pengguna terhadap sistem.

Berdasarkan hasil pengujian, tingkat risiko kerentanan dikategorikan sebagai risiko tinggi karena kerentanan ditemukan pada fitur yang berhubungan langsung dengan proses autentikasi dan akses data.

Rekomendasi Perbaikan

Berdasarkan hasil analisis yang dilakukan, beberapa rekomendasi perbaikan yang dapat diterapkan antara lain:

1. Menggunakan prepared statement atau parameterized query pada seluruh proses akses database.
2. Menerapkan validasi dan sanitasi input pengguna.
3. Membatasi hak akses akun database sesuai kebutuhan sistem.
4. Mengimplementasikan Web Application Firewall (WAF).
5. Melakukan pengujian keamanan secara berkala menggunakan standar OWASP.

Penerapan rekomendasi tersebut diharapkan dapat mengurangi risiko eksploitasi SQL Injection dan meningkatkan tingkat keamanan sistem informasi secara keseluruhan.

KESIMPULAN

Berdasarkan hasil penelitian yang telah dilakukan menggunakan metode penetration testing berdasarkan standar OWASP Web Security Testing Guide (WSTG), dapat disimpulkan bahwa sistem informasi yang diuji masih memiliki kerentanan terhadap serangan SQL Injection pada beberapa parameter yang berinteraksi langsung dengan database. Hasil pengujian manual dan verifikasi menggunakan SQLMap menunjukkan bahwa 3 dari 5 parameter yang diuji berhasil dieksploitasi, sehingga tingkat kerentanan sistem mencapai 60%.

Kerentanan yang ditemukan berpotensi menyebabkan berbagai risiko keamanan, seperti kebocoran data, manipulasi data, peningkatan hak akses secara tidak sah, serta gangguan terhadap integritas dan ketersediaan informasi. Kondisi tersebut menunjukkan bahwa mekanisme validasi input dan pengamanan query database pada sistem belum diterapkan secara optimal.

Untuk mengurangi risiko serangan SQL Injection, disarankan agar pengembang menerapkan prepared statement atau parameterized query, melakukan validasi dan sanitasi input pengguna, membatasi hak akses database sesuai kebutuhan, serta mengimplementasikan Web Application Firewall (WAF). Selain itu, pengujian keamanan secara berkala perlu dilakukan untuk memastikan bahwa sistem tetap terlindungi dari berbagai ancaman keamanan yang terus berkembang. Dengan penerapan rekomendasi tersebut, tingkat keamanan sistem informasi diharapkan dapat meningkat dan risiko eksploitasi SQL Injection dapat diminimalkan.`

Ucapan Terima Kasih

Penulis menyampaikan rasa terima kasih yang sebesar-besarnya kepada seluruh pihak yang telah memberikan kontribusi nyata dalam penyelesaian penelitian ini. Ucapan terima kasih secara khusus ditujukan kepada Bapak/Ibu dosen pembimbing yang telah memberikan arahan, bimbingan, dan koreksi yang sangat konstruktif selama proses penelitian berlangsung. Apresiasi mendalam juga penulis sampaikan kepada pimpinan dan civitas akademika institusi atas dukungan fasilitas serta iklim akademik yang kondusif bagi pengembangan riset di bidang keamanan sistem informasi. Penulis turut mengucapkan terima kasih kepada komunitas keamanan informasi, khususnya OWASP dan para peneliti independen yang telah mempublikasikan temuan dan panduan yang menjadi rujukan penting dalam penelitian ini. Akhirnya, penghargaan yang tulus penulis sampaikan kepada keluarga tercinta atas segala doa dan dukungan moral yang menjadi kekuatan utama dalam menyelesaikan penelitian ini.

DAFTAR PUSTAKA

Akbar, F. B., Sipayung, A. F. P., Lisara, A. D., Syakira, N., & Raisa, D. (2026). Analisis Keamanan Sistem Akademik Berbasis Web terhadap Serangan SQL Injection. *JIKUM: Jurnal Ilmu Komputer*, *2*(1), 60–64. https://doi.org/10.62671/jikum.v2i1.161

Anugrah, T. (2024). Penetration Testing Keamanan Website Stie Samarinda Menggunakan Teknik Sql Injection Dan Xss. *Jurnal Informatika Dan Teknik Elektro Terapan*, *12*(1), 618–624. https://doi.org/10.23960/jitet.v12i1.3882

Dwi Kurniawan, N., Maulana Firdaus, A., Rizky Abriansah, F., Rendi Ferdian, P., Ferdian, & Susanto. (2025). Analisis Kerentanan Sql Injection MenggunakanSqlmap Pada Kali Linux. *STRING (Satuan Tulisan Riset Dan Inovasi Teknologi)*, *10*(1), 58–68.

Fadilah Nuria Handayani, Intan Diasih, Vrisa Arana Salsabilla, & Aprilia Pramudita. (2024). Sistem Pendukung Keputusan Dalam Pemilihan Smartphone Terbaik Menggunakan Metode SAW. *Bridge : Jurnal Publikasi Sistem Informasi Dan Telekomunikasi*, *2*(3), 130–141. https://doi.org/10.62951/bridge.v2i3.127

Fadillah, M., & Servanda, Y. (2024). Analisis Efektivitas Teknik Parameterized Queries Dalam Mencegah Serangan SQL Injection Menggunakan DVWA. *JUPITER :*

Journal of Computer & Information Technology, *5*(2), 57–69. https://doi.org/10.53990/jupiter.v5i2.323

Fernanda, R., Data, M., Bakhtiar, F. A., Studi, P., Informatika, T., Komputer, F. I., Brawijaya, U., Blind, B., Web, K. A., Matrix, C., & Manual, V. (2026). *ANALISIS EFEKTIVITAS OWASP ZAP DALAM MENDETEKSI*. *10*(3), 1–10.

Hariyani, A., & Dolia, P. (2024). An innovative method for detecting SQLi attacks by altering SQL query attribute values. *International Journal of Advanced Computer Research*, *14*(68), 89–96. https://doi.org/10.19101/ijacr.2024.1466005

Imanuel Toding Bua, & Nur Isdah Idris. (2025). Analisis Kebijakan Keamanan Siber di Indonesia: Studi Kasus Kebocoran Data Nasional pada Tahun 2024. *Desentralisasi : Jurnal Hukum, Kebijakan Publik, Dan Pemerintahan*, *2*(2), 100–114. https://doi.org/10.62383/desentralisasi.v2i2.653

Lestari. (2019). Bab I حضْ خِ يِ. *Galang Tanjung*, (2504), 1–9.

Mu'min, M. A., Alamin, Z., Fathir, & Ramadhan, S. (2024). Uji Penetrasi Injeksi SQL terhadap Celah Keamanan Website XYZ menggunakan Tools SQLMap. *Scientific : Journal of Computer Science and Informatics*, *1*(2), 47–52. https://doi.org/10.34304/scientific.v1i2.330

Mushlih, M., Fitri, R., Wardiah, I., & Negeri Banjarmasin, P. (2019). Penetration Testing Tool Untuk Menguji Kerentanan Sql Injection Secara Otomatis Berbasis Web. *Prosiding SNRT (Seminar Nasional Riset Terapan)*, *5662*(November), 41–47. https://www.polman-babel.ac.id/

Nugraha, L. A., Kautsar, I. A., & Fitrani, A. S. (2024). SQL Injection: Analisis Efektivitas Uji Penetrasi dalam Aplikasi Web. *Smatika Jurnal*, *14*(01), 111–123. https://doi.org/10.32664/smatika.v14i01.1224

Nur Fikri, M., Parga Zen, B., Adhitama, R., & Ahmad Firdaus, E. (2023). Analisis Keamanan Sistem Informasi Website SMA Negeri 1 Sokaraja Menggunakan Metode Penetration Testing Execution Standard (PTES). *Jurnal Informatika*, *2*(2), 19–27. https://doi.org/10.57094/ji.v2i2.1046

Putranto, D. P., Jayanta, J., & Hananto, B. (2022). Analisis Keamanan Website Leads UPNVJ Terhadap Serangan SQL Injection & Sniffing Attack. *Informatik : Jurnal Ilmu Komputer*, *18*(3), 230. https://doi.org/10.52958/iftk.v18i3.4690

Rafeli, A. I., Seta, H. B., & Widi, I. W. (2022). Pengujian Celah Keamanan Menggunakan Metode OWASP Web Security Testing Guide (WSTG) pada Website XYZ. *Informatik : Jurnal Ilmu Komputer*, *18*(2), 97. https://doi.org/10.52958/iftk.v18i2.4632

Sagraha, R., Ulfa, M., Zuhriyadi, I., & Syazili, A. (2026). *Pelatihan Analisis Kerentanan Sistem Informasi Menggunakan Vulnerability Assessement Bagi Siswa Magang Universitas Binadarma*. *3*(11), 6119–6123.

Saroni, M. I. N., & Mulyanti, B. (2020). Hypertext preprocessor framework in the development of web applications. *IOP Conference Series: Materials Science and Engineering*, *830*(2). https://doi.org/10.1088/1757-899X/830/2/022096

Sistem, T. (2025). *Informasi*. *6*(1), 50–55. https://doi.org/10.62527/jitsi.

Wahyudin, Y., & Rahayu, D. N. (2020). ANALISIS METODE PENGEMBANGAN SISTEM INFORMASI BERBASIS WEBSITE : A LITERATUR REVIEW lepas dengan System Development Life Cycle yang terdiri dari teks , gambar , suara animasi. *Jurnal Interkom: Jurnal Publikasi Ilmiah Bidang Teknologi Informasi Dan Komunikasi*, *15*, 26–40.

Wijaya, M. C. (2024). Security Analysis of SQL Injection Attacks on Multimedia and Journal-Services Sites Using Concatenated Input Validation and Parsing Method (CIVP). *Ingenierie Des Systemes d'Information*, *29*(5), 1915–1924. https://doi.org/10.18280/isi.290523