



OPTIMALISASI KINERJA CMS WORDPRESS MELALUI PENERAPAN REPLIKASI DATABASE MYSQL

Neil Kevin Hulu^{1*}, Tommy², Divi Handoko³

1,2,3) Prodi Teknik Informatika, Fakultas Teknik dan Komputer, Universitas Harapan Medan, Indonesia

*Corresponding Email: neilkevinhulu@gmail.com

Abstrak

Internet sebagai salah satu media dalam penyampaian informasi turut mengubah bentuk dan proses perpindahan data dari data fisik menjadi data digital. Termasuk blog dan situs web yang menyediakan ribuan informasi perdetiknya. Performa dan kinerja blog-blog yang handal menjadi isu penting untuk blogger yang menggunakan hosting sendiri. Umumnya terdapat dua alternatif cara yang dapat digunakan untuk optimasi database pada content management system (CMS) wordpress, yaitu dengan phpMyAdmin atau dengan WP-Optimize. Tetapi dari pilihan tersebut masih banyak kekurangan dalam hasil akhir optimisasinya. Adapun tujuan penelitian ini adalah untuk mempercepat kinerja sistem CMS Wordpress dalam membaca database. Metode dalam penelitian ini perancangan dan implementasi sistem dengan menggunakan wordpress. Hasil dari penelitian ini dengan adanya perubahan yang dilakukan sehingga mempercepat sistem CMS Wordpress dalam membaca database.

Kata Kunci: Database, Wordpress, Optimasi, Kinerja, Performa.

Abstract

The internet as a medium for conveying information has also changed the form and process of transferring data from physical data to digital data. Including blogs and web sites that provide thousands of information per second. Reliable performance and performance of blogs is an important issue for bloggers who use self-hosting. Generally, there are two alternative methods that can be used to optimize the database on the WordPress content management system (CMS), namely with phpMyAdmin or with WP-Optimize. But from these choices there are still many shortcomings in the final optimization result. The purpose of this research is to speed up the performance of the Wordpress CMS system in reading databases. The method in this study is the design and implementation of the system using wordpress. The results of this study with the changes made so as to speed up the Wordpress CMS system in reading the database.

Keywords: Database, Wordpress, Optimization, Performance, Performance.

PENDAHULUAN

Perkembangan pesat teknologi informasi dan internet telah mengubah cara kita menyampaikan dan mengakses informasi. Salah satu dampak signifikan dari perkembangan ini adalah migrasi data dari bentuk fisik ke digital, yang memungkinkan penyampaian informasi melalui berbagai platform, seperti blog dan situs web. Setiap detik, ribuan informasi dapat diakses melalui platform-platform ini (Surentu et al., 2020). Hal ini tentunya menciptakan tantangan tersendiri dalam menjaga performa dan kinerja blog atau situs web, terutama bagi pengguna yang mengandalkan hosting sendiri (Purba et al., 2020).



Content Management System (CMS) WordPress menjadi salah satu platform yang populer digunakan untuk membuat dan mengelola situs web, termasuk blog. Seiring dengan meningkatnya jumlah pengguna dan volume data yang ditangani, performa situs web WordPress menjadi isu penting yang perlu dioptimalkan (Khaliq et al., 2023). Pada umumnya, terdapat dua alternatif utama yang digunakan untuk optimasi database WordPress, yaitu melalui phpMyAdmin atau menggunakan plugin WP-Optimize. Meskipun kedua metode ini sudah cukup populer, keduanya masih memiliki kekurangan dalam hasil akhirnya, khususnya dalam hal efisiensi dan kecepatan (Wijoyo et al., 2020).

Salah satu solusi yang dapat diterapkan untuk mengatasi masalah ini adalah dengan menambahkan metode replikasi database ke dalam sistem web tersebut. Replikasi database memungkinkan distribusi beban kerja antara beberapa server, yang dapat mengurangi beban pada server utama dan meningkatkan kinerja sistem secara keseluruhan. Dengan menerapkan replikasi database pada CMS WordPress, diharapkan dapat tercapai optimasi performa yang lebih baik dan stabil (Abdurrahman et al., 2022).

Replikasi Database Management Systems (DBMS) menggunakan MySQL merupakan teknik yang efektif untuk mendistribusikan data dan objek-objek database dari satu database ke database lain yang terpisah secara fisik. Teknik ini memungkinkan data untuk disalin dan disebar ke lokasi-lokasi berbeda melalui jaringan lokal atau internet (Muliawan, 2023). Dalam penelitian ini, sistem database server MySQL WordPress akan diimplementasikan menggunakan metode replikasi master-slave, di mana plugin HyperDB digunakan untuk mengelola replikasi antar server. Dengan menambahkan beberapa server database ke dalam sistem, aplikasi WordPress dapat membaca data dari beberapa server sekaligus, yang pada gilirannya dapat mempercepat kinerja sistem dalam mengakses database (M. Siregar et al., 2021).

Replikasi MySQL dipilih dalam penelitian ini karena memberikan manfaat yang signifikan bagi sistem yang lebih banyak melakukan aktivitas membaca data dibandingkan menulis, seperti yang terjadi pada banyak instalasi WordPress. Dengan menggunakan satu master server dan beberapa slave servers, kita dapat menambahkan slave servers tanpa memberikan beban yang berat pada sistem, sehingga memungkinkan peningkatan kinerja yang



lebih baik dan lebih efisien.

Berdasarkan penjelasan di atas, penelitian ini bertujuan untuk mengoptimalkan kinerja CMS WordPress dengan menggunakan replikasi database MySQL. Diharapkan hasil penelitian ini dapat memberikan kontribusi dalam meningkatkan performa server hosting, mempercepat pemuatan media informasi, dan memperbaiki pengelolaan konten dalam sistem web.

KAJIAN TEORI

CMS WordPress

WordPress adalah sebuah sistem manajemen konten (CMS) yang paling populer di dunia (Ramadhani et al., 2024). CMS ini memungkinkan pengguna untuk membuat, mengelola, dan mempublikasikan konten di website tanpa memerlukan keahlian coding (Aqsal & Sutrisno, 2022). WordPress berbasis PHP dan MySQL, sehingga sangat fleksibel dan dapat digunakan untuk berbagai jenis website, dari blog pribadi hingga situs e-commerce besar (Irfannudin & Hamdani, 2020).

Beberapa fitur utama WordPress termasuk (Ademarbun et al., 2024):

1. Antarmuka Pengguna yang Ramah: Desain yang intuitif memudahkan pengelolaan konten bahkan untuk pemula.
2. Plugin dan Tema: Pengguna dapat mengubah tampilan dan fungsi website dengan memilih berbagai tema dan plugin yang tersedia.
3. SEO Friendly: WordPress dilengkapi dengan berbagai alat untuk optimasi mesin pencari (SEO), yang membantu meningkatkan visibilitas website.
4. Komunitas Aktif: Terdapat banyak sumber daya dan dukungan dari komunitas global untuk membantu pengembangan dan pemecahan masalah.
5. Keamanan: Meskipun WordPress cukup aman, pengelolaannya perlu dipastikan selalu.

Database

Database adalah sistem yang digunakan untuk mengorganisir, menyimpan, dan mengelola data dalam bentuk terstruktur, sehingga memudahkan pengguna untuk mengakses, memodifikasi, dan menganalisis data tersebut (Syahputri & Nasution, 2023). Data dalam database biasanya disusun dalam tabel yang terdiri dari baris (*record*) dan kolom (*field*), dengan



setiap tabel memiliki kunci utama (*primary key*) untuk mengidentifikasi record secara unik(Nurhayati & Nasution, 2023).

Beberapa komponen utama dalam database meliputi(Ramadhan & Mukhaiyar, 2020):

1. DBMS (Database Management System): Perangkat lunak yang digunakan untuk mengelola dan mengoperasikan database. Contoh DBMS yang populer adalah MySQL, PostgreSQL, Oracle, dan Microsoft SQL Server.
2. Model Database: Menentukan bagaimana data disusun dan diakses. Model yang paling umum adalah model relasional, di mana data disimpan dalam tabel-tabel yang saling berhubungan.
3. Query: Bahasa atau perintah yang digunakan untuk mengambil, mengubah, atau menghapus data dari database. SQL (Structured Query Language) adalah bahasa yang paling umum digunakan untuk berinteraksi dengan database relasional.
4. Keamanan: Database harus dilindungi dengan kontrol akses untuk memastikan hanya pengguna yang berwenang yang dapat mengakses data tertentu.
5. Backup dan Recovery: Proses untuk mencadangkan data dan memulihkan data yang hilang atau rusak.

MySQL

MySQL adalah sebuah sistem manajemen basis data relasional (RDBMS) yang bersifat open-source(U. K. Siregar et al., 2024). Diciptakan oleh perusahaan Swedia MySQL AB dan saat ini dikembangkan oleh Oracle Corporation, MySQL menggunakan Structured Query Language (SQL) untuk mengelola dan mengakses data dalam basis data(Anggoro et al., 2021).

Berikut adalah beberapa konsep dasar tentang MySQL(Anggoro et al., 2021):

1. Relasional: MySQL menggunakan tabel untuk menyimpan data. Setiap tabel terdiri dari baris dan kolom, yang memungkinkan pengaturan data dengan cara yang terstruktur.
2. SQL: Bahasa yang digunakan untuk berinteraksi dengan basis data. SQL memungkinkan pembuatan, pembacaan, pembaruan, dan penghapusan data.
3. CRUD: Operasi dasar dalam manajemen basis data, yaitu Create (membuat), Read (membaca), Update (memperbarui), dan Delete (menghapus) data.
4. Primary Key: Kolom atau kumpulan kolom yang digunakan untuk mengidentifikasi

setiap baris dalam tabel secara unik.

5. Foreign Key: Kolom yang digunakan untuk menghubungkan tabel satu dengan yang lain, membentuk hubungan antara data.

MySQL sering digunakan untuk aplikasi web, dan mendukung berbagai platform dan bahasa pemrograman. Karena bersifat open-source, MySQL sangat populer di kalangan pengembang dan pengguna yang membutuhkan basis data yang cepat, andal, dan dapat diskalakan (Widia & Asriningtias, 2021).

Replikasi Database

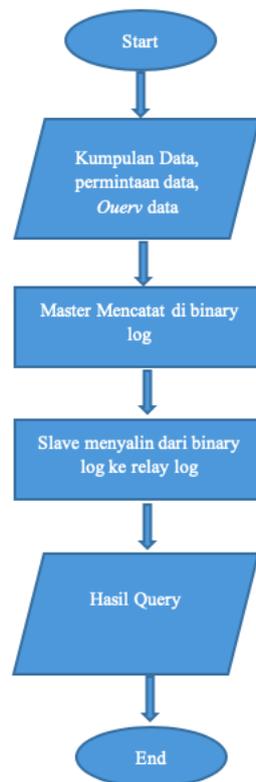
Replikasi Database adalah proses di mana data dari satu database (disebut master) disalin atau direplikasi ke satu atau lebih database lain (disebut slave atau replica) (Heryanto & Hartati, 2020). Tujuan utama dari replikasi ini adalah untuk meningkatkan ketersediaan dan keandalan data, serta meningkatkan performa sistem (Jaelani, 2021).

Berikut adalah konsep dasar tentang replikasi database (Iskandar et al., 2024):

1. Master-Slave Replication: Data diubah dan diperbarui di database master dan perubahan tersebut kemudian direplikasi ke database slave. Slave hanya bersifat read-only.
2. Multi-Master Replication: Setiap node dalam replikasi dapat berfungsi sebagai master. Perubahan data dapat dilakukan di mana saja dan kemudian disinkronkan ke semua node.
3. Synchronous vs Asynchronous Replication: Dalam replikasi sinkron, perubahan data harus diterapkan di semua node secara bersamaan sebelum operasi dianggap selesai. Dalam replikasi asinkron, perubahan di master segera direplikasi ke slave, tetapi tidak harus menunggu konfirmasi dari slave untuk menyelesaikan operasi.
4. Conflict Resolution: Dalam replikasi multi-master, jika dua node melakukan perubahan pada data yang sama secara bersamaan, sistem harus mampu menyelesaikan konflik tersebut.
5. Data Consistency: Replikasi dapat menyebabkan masalah konsistensi data jika tidak dikelola dengan baik, terutama dalam replikasi asinkron.

METODE PENELITIAN

Metode yang digunakan dalam pengolahan database adalah replikasi database. Di dalam replikasi database juga terdapat beberapa database didalamnya, dimana database tersebut ada yang berfungsi sebagai master (utama) dan ada juga yang berfungsi sebagai slave (kedua), seperti yang dijelaskan pada Gambar 1.



Gambar 1. Flowchart Replikasi Database

Prinsip kerja dari replikasi database adalah data ketika ada *query* dari pengguna (*user*), permintaan *query* tersebut akan diolah oleh replikasi database yang didalamnya terdapat *master* database dan *slave* database. Sesuai dengan Gambar 3. ketika mendapat perintah untuk melakukan *query* maka data akan diolah menggunakan metode replikasi database. Dimana database *master* akan mencatat perubahan perintah yang diterima pada *binary log*. Kemudian database *slave* akan menyalin aktivitas dari *binary log* kedalam *relay log* yang ada pada database *slave*. Di mulai dengan susunan pekerja, yang disebut *thread slave I/O*. *Thread I/O* membuka koneksi klien agar bisa terhubung ke *master*. Setelah itu *slave* akan merespon proses



relay log dan selanjutnya mengambil semua data di *master*, dan *thread* ini diam menunggu *master* memberi sinyal lagi ketika ada events baru. Lalu *thread I/O* ini mencatat aktivitas ke *relay log slave*. Thread ini membaca dan merespon aktivitas dari *relay log*, sehingga memperbarui data *slave* agar sesuai dengan *master*.

Selanjutnya melakukan implementasi proses mengonversi dari tahap perancangan sistem kedalam bentuk sistem berdasarkan dengan data-data yang telah di pelajari dari studi literatur dan observasi yang telah dilakukan. Dan akan dilakukan pengujian terhadap sistem yang telah dibangun untuk mengetahui apakah penelitian berhasil atau tidak.

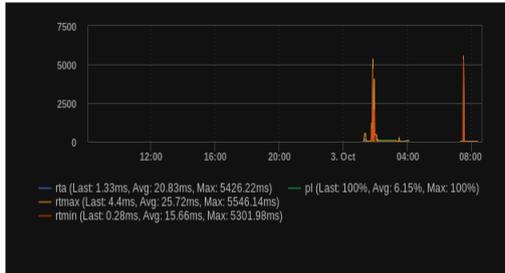
Setelah implementasi sistem telah berhasil di setiap server yang dibangun, di tahap ini dilakukan pengujian pada seluruh server sebagai sebuah sistem private cloud.

Pada tahap selanjutnya sistem telah berhasil di bangun. Pemeliharaan meliputi perbaikan kekurangan-kekurangan ataupun error saat pengembangan dan pengembangan sistem

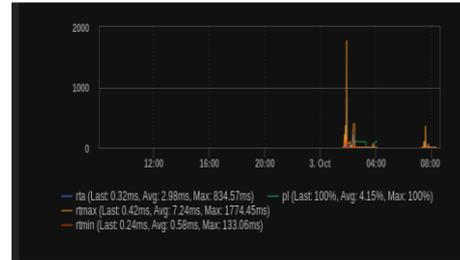
HASIL DAN PEMBAHASAN

Pada kegiatan pengujian performa tentunya ada beberapa titik yang dijadikan sebagai acuan untuk standar. Dalam pengujian kali ini ada acuan yang digunakan untuk sebagai standar performansi tentunya dengan menggunakan spesifikasi laptop yang sama dari segi prosesor, RAM, sistem operasi dan aplikasi yang berjalan. Yang pertama adalah dari segi RTA (*Round Trip Average*) merupakan waktu yang dicapai pada saat satu kali pengiriman dan penerimaan data dari *server*. Kedua adalah dari presentase *service* ketika berjalannya proses *query* pada suatu database *server*, yang dapat meningakibatkan terjadinya *server down*.

Setelah dilakukan konfigurasi, maka dilakukan pengujian antara database replikasi dengan single database dan diperoleh hasil pengujian berupa grafik, serta diagram seperti pada Gambar 2.



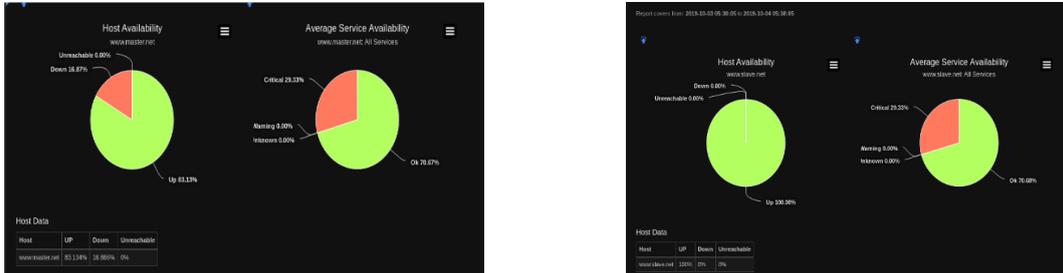
a. Grafik Keluaran Replikasi Database



b. Grafik Keluaran Single Database

Gambar 2. Grafik Keluaran Database

Dari hasil keluaran grafik pada *Nagiosxi* saat melakukan *query* database dapat dilihat pada Gambar 2(a). bahwa pada replikasi database tingkat *RTAmax* (*Round Trip Average*) yang dicapai adalah 5426.22ms, RTA adalah merupakan waktu yang dicapai pada saat melakukan pengiriman dan penerimaan data dari *server*. Sedangkan pada Gambar 2 (b) dapat dilihat bahwa ketika *single* database melakukan *query* hasil *RTAmax* yang dicapai pada saat melakukan *query* adalah 1774.45ms, dimana hasil tersebut lebih rendah dibandingkan dari database yang terreplikasi. Dengan database yang digunakan sama yakni radius dan juga melakukan *query* yang sama hasil yang didapatkan berbeda, jauh hal ini dikarenakan untuk *server* database yang menggunakan replikasi dalam melakukan *query* karena terdapat 2 database didalamnya yakni *master* database dan *slave* database, yang fungsinya sendiri digunakan adalah sebagai penggandaan dari database yang ada. Dari adanya 2 database *server* inilah yang menyebabkan proses pencarian *query* pada replikasi database membutuhkan waktu yang lebih lama sehingga tingkat *RTAmax* yang dicapa lebih tinggi daripada *single* database. Karena untuk melakukan pencarian sebuah *query* dilakukan pada kedua database yakni *master* dan *slave*, dan hal ini dilakukan secara bergantian sehingga waktu yang dibutuhkan juga lama. Berbeda dengan yang terjadi pada *single* database, *RTAmax* yang dicapai oleh *single* database hasilnya lebih rendah dikarenakan pada saat terjadi pencarian sebuah *query*, *single* database hanya melakukan pencarian pada satu database saja, sehingga RTA yang dibutuhkan tidak terlalu tinggi. Jadi dapat disimpulkan dari hasil RTA yang ada pada grafik, *single* database hasilnya berada 1 tingkat lebih unggul daripada replikasi database, karena RTA dari *single* database nilainya lebih rendah dibandingkan RTA dari replikasi database.



a. Diagram Service dari Replikasi Database b. Diagram Service Single Database

Gambar 3. Diagram Service Database

Dari Gambar 3 (a) yang merupakan hasil diagram dari replikasi database dapat dilihat bahwa kemungkinan terjadinya *server down* masih dapat terjadi dengan presentase sebesar 16,68%, tetapi masih dapat mengatasi tingkat serangan *query* sebesar 29% dengan lancar, tanpa adanya pemberitahuan bahaya. Sedangkan pada Gambar 3(b) *single* database dapat dilihat bahwa performa ketika berjalan hampir tidak pernah mengalami *server down* 0.00%, walaupun menghadapi tingkat serangan *query* sebesar 29% yang sama dengan database yang sudah menerapkan replikasi.

Diluar dari RTA, untuk tingkat keilangan data jika terjadi kerusakan ataupun *server down* sudah dapat dipastikan bahwa pada *single* database, tidak dapat diselamatkan karena hanya terdiri dari satu database saja sehingga tidak ada database lain yang menggantikan tugas dari database tersebut ketika terjadi kerusakan, berbeda dengan database yang sudah tereplikasi ketika terjadi kesalahan ataupun *server down* dari salah satu database (*master*), maka database yang lain (*slave*) akan bisa digunakan karena sebelumnya sudah terdapat replikasi dari database utama. Berikut adalah bentuk pengujian dari sistem replikasi. Pengujiannya dilakukan dengan cara ditambahkan database baru pada sisi *master*, dan dicek pada sisi *slave* apakah juga terbentuk database baru sesuai pada *master* atau tidak.

```
MariaDB [(none)]> show databases
-> ;
+-----+
| Database |
+-----+
| information_schema |
| mysql |
| performance_schema |
+-----+
3 rows in set (0.03 sec)

MariaDB [(none)]> █
```

```
MariaDB [(none)]> show databases;
+-----+
| Database |
+-----+
| information_schema |
| mysql |
| performance_schema |
+-----+
3 rows in set (0.00 sec)

MariaDB [(none)]> _
```

a. Database Master Sebelum Replikasi b. Database Slave Sebelum Replikasi

Gambar 4. Database Master

```
MariaDB [(none)]> create database radius;
Query OK, 1 row affected (0.00 sec)

MariaDB [(none)]> show databases;
+-----+
| Database |
+-----+
| information_schema |
| mysql |
| performance_schema |
| radius |
+-----+
4 rows in set (0.00 sec)

MariaDB [(none)]>
```

```
MariaDB [(none)]> show databases;
+-----+
| Database |
+-----+
| information_schema |
| mysql |
| performance_schema |
| radius |
+-----+
4 rows in set (0.00 sec)

MariaDB [(none)]>
```

a. Database Master Setelah Replikasi

b. Database Slave Setelah Replikasi

Gambar 5. Database Master

Dapat dilihat pada Gambar 4. menjelaskan keadaan dari database *master* dan *slave* ketika belum direplikasi. Setelah dilakukan proses replikasi, ketika pada sisi *master* dibuat database baru maka pada sisi juga akan otomatis terbentuk database yang sama seperti pada sisi master seperti pada Gambar 5. Apabila hal ini terjadi dapat diartikan bahwa metode replikasi yang dibuat sudah berhasil berjalan. Dengan diterapkannya sistem replikasi ini, mampu mengurangi terjadinya kehilangan data apabila salah satu database mengalami kerusakan, dan tidak perlu melakukan penggandaan secara manual karena sudah otomatis tergandakan dengan menggunakan metode replikasi. Sehingga dapat memberikan kemudahan kepada pengguna untuk menggandakan database pada suatu instansi ataupun perusahaan berukuran cukup besar karena tidak perlu melakukan penggandaan secara manual yang mungkin saja bisa terjadi kelalaian atau *human error*. Pada tabel 1, dijelaskan perbedaan performa dari hasil pengukuran menggunakan *NagiosXI* ketika dilakukan *query* terhadap replikasi database dan *single* database.

Tabel 1. Hasil Perbandingan Performa Replikasi Database dan Single Database

Hasil Pengujian	Replikasi Database	Singel Database
Round Trip Max	5545.24 ms	1774.45 ms
Round Trip Min	5301.98 ms	133.06 ms
Critical	29 %	29 %
Server Up	83.134 %	100 %
Server Down	16.866 %	0 %

SIMPULAN

Berdasarkan hasil penelitian yang telah dilakukan, dapat disimpulkan bahwa penggunaan replikasi database MySQL pada CMS WordPress dapat memberikan dampak yang signifikan



terhadap performa sistem. Meskipun replikasi database menunjukkan RTA yang lebih tinggi dibandingkan dengan single database, yakni sebesar 5426.22ms dibandingkan dengan 1774.45ms pada single database, namun replikasi database tetap memberikan keunggulan dalam hal pengelolaan data. Dalam hal ini, meskipun terjadi serangan query dengan persentase 29%, replikasi database mengalami server down dengan presentase lebih rendah (16,68%) dibandingkan dengan single database yang tidak mengalami server down. Selain itu, metode replikasi database juga dapat mengurangi risiko kehilangan data akibat human error dalam penggandaan manual serta menghemat waktu bagi pekerja dalam mengatasi masalah kerusakan data, karena data yang direplikasi tetap tersedia pada server lain. Oleh karena itu, replikasi database dapat menjadi solusi efektif dalam mengoptimalkan kinerja WordPress, meningkatkan keandalan sistem, dan mempercepat pengelolaan data pada server.

DAFTAR PUSTAKA

- Abdurrahman, A., Nushair, M. A., & Yahya, K. (2022). Penerapan Replikasi Basis Data Terdistribusi Menggunakan Metode Synchronous Pada Hotel Favehotel Makassar. *Nusantara Hasana Journal*, 1(12), 46–53.
- Ademarbun, A. C. L. M., Sianturi, U. S., Rizki, D., Pakpahan, Y. R., Nasution, H. S., & Kurniawan, H. (2024). Pengembangan Website Top Up Game Pengembangan Website Top Up Game Online Dengan Menggunakan CMS Wordpress: Pengembangan Website Top Up Game Online Dengan Menggunakan CMS Wordpress. *Jurnal Komputer Teknologi Informasi Dan Sistem Informasi (JUKTISI)*, 3(1), 689–696.
- Anggoro, D. A., Kom, S., Supriyanti, W., Kom, S., Kom, M., Putri, D. A. P., & Kom, S. (2021). *Konsep Dasar Sistem Basis Data dengan MySQL*. Muhammadiyah University Press.
- Aqsal, M., & Sutrisno, J. (2022). Analisa dan Implementasi E-Commerce Berbasis Content Management System (CMS) Wordpress Pada Bengkel Madmen Motogarage. *Prosiding Seminar Nasional Mahasiswa Fakultas Teknologi Informasi (SENAFTI)*, 1(1), 1831–1839.
- Heryanto, A., & Hartati, Y. (2020). Backup Database Dengan Multi Master Replikasi Pada Kluster Server. *Jurnal Ilmiah Ilmu Komputer Fakultas Ilmu Komputer Universitas Al*



- Asyariah Mandar*, 6(1), 7–12.
- Irfannudin, F. A., & Hamdani, A. U. (2020). Implementasi Sistem E-Commerce Pada Guna Motor Berbasis Content Management System (Cms) Wordpress. *IDEALIS: InDonEsiA Journal Information System*, 3(1), 42–48.
- Iskandar, A. P. S., Setiawan, H., Judijanto, L., Mahendra, G. S., Ardi, M., Putri, N. A. R., Handika, I. P. S., Ratih, R., Mandowen, S. A., Wazaumi, D. D., & others. (2024). *Teknologi Big Data: Pengantar Dan Penerapan Teknologi Big Data Di Berbagai Bidang*. PT. Green Pustaka Indonesia.
- Jaelani, W. L. (2021). Implementasi Replikasi Basis Data Dan Model Discretionary Acces Control Untuk Keamanan Database Studi Kasus Smk Plus Pratama Adi Banjaran. *Scientia Regendi*, 2(2), 104–115.
- Khaliq, A., Arianti, C., Simanjuntak, C. A., & Harahap, D. A. P. (2023). Perancangan Website Profil Program Studi Menggunakan Content Management System Wordpress. *Jurnal Nasional Teknologi Komputer*, 3(3), 196–201.
- Muliawan, D. (2023). PENERAPAN REPLIKASI SYNCHRONOUS PADA APLIKASI DATA KRS ONLINE PADA UNIVERSITAS JABAL GHAFUR MENGGUNAKAN POSTGRESQL. *Future Academia: The Journal of Multidisciplinary Research on Scientific and Advanced*, 1(1), 31–45.
- Nurhayati, S. T., & Nasution, M. I. P. (2023). Database Management System Pada Perusahaan. *Jurnal Akuntansi Keuangan Dan Bisnis*, 1(2), 62–64.
- Purba, R. A., Sudarso, A., Silitonga, H. P., Sisca, S., Supitriyani, S., Yusmanizar, Y., Nainggolan, L. E., Sudirman, A., Widyastuti, R. D., Novita, A. D., & others. (2020). *Aplikasi teknologi informasi: teori dan implementasi*. Yayasan Kita Menulis.
- Ramadhan, R. F., & Mukhaiyar, R. (2020). Penggunaan Database Mysql dengan Interface PhpMyAdmin sebagai Pengontrolan Smarhome Berbasis Raspberry Pi. *JTEIN: Jurnal Teknik Elektro Indonesia*, 1(2), 129–134.
- Ramadhani, G. T. A., Steyer, M. R. R., Maulidan, M. H., & Setiawan, A. (2024). Analisis Kerentanan WordPress dengan WPScan dan Teknik Mitigasi. *Journal of Internet and Software Engineering*, 1(4), 15.



- Siregar, M., Kusumawati, N., Susanto, A., & others. (2021). Sistem Informasi Company Profile Pt. Sinar Nusantara Sakti Menggunakan Wordpress. *Jurnal Sistem Informasi Dan Sistem Komputer*, 6(1), 46–57.
- Siregar, U. K., Sitakar, T. A., Haramain, S., Lubis, Z. N. S., Nadhirah, U., Yahfizham, Y., & others. (2024). Pengembangan database Management system menggunakan My SQL. *Jurnal Sains, Teknologi \& Komputer*, 1(1), 8–12.
- Surentu, Y. Z., Warouw, D. M. D., & Rembang, M. (2020). Pentingnya website sebagai media informasi destinasi wisata di dinas kebudayaan dan pariwisata kabupaten minahasa. *Acta Diurna Komunikasi*, 2(4).
- Syahputri, K., & Nasution, M. I. P. (2023). Peran Database Dalam Sistem Informasi Manajemen. *Jurnal Akuntansi Keuangan Dan Bisnis*, 1(2), 54–58.
- Widia, D. M., & Asriningtias, S. R. (2021). *Cara Cepat dan Praktis Membangun Web Dinamis dengan PHP dan MySQL*. Universitas Brawijaya Press.
- Wijoyo, H., Cahyono, Y., Ariyanto, A., & Wongso, F. (2020). Digital economy dan pemasaran era new normal. *Insan Cendekia Mandiri*.