STORAGE – Jurnal Ilmiah Teknik dan Ilmu Komputer, Vol. 4 No. 2, Mei 2025, 82 - 91
e-ISSN: 2828-5344 | DOI: 10.55123

# TINJAUAN LITERATUR DENGAN PENDEKATAN *SYSTEMATIC LITERATURE REVIEW* UNTUK OPTIMASI KUERI DALAM BASIS DATA

**[*1)]Ega Budiman, [2)]Muhammad Fadli, [3)]David Kurniawan, [4)]Erliyan Redy Susanto**

[1,2,3,4)] Fakultas Teknik dan Ilmu Komputer, Magister Ilmu Komputer, Universitas Teknokrat Indonesia
[2)] Jurusan Ekonomi dan Bisnis, Pengelolaan Perhotelan, Politeknik Negeri Lampung
[1)] ega_budiman@teknokrat.ac.id, [2)] muhammadfadliofficial@polinela.ac.id,
[4)]davidkurniawan@teknokrat.ac.id, [5)]erliyan.redy@teknokrat.ac.id

**INFO ARTIKEL**

**Riwayat Artikel:**
Diterima : 7 Mei 2025
Disetujui : 27 Mei 2025

**Kata Kunci:**
Optimasi Kueri, Basis Data MySQL, SLR

**ABSTRAK**

Efisiensi basis data MySQL sangat bergantung pada teknik optimasi kueri yang diterapkan, terutama dalam skenario dengan dataset besar dan kompleksitas kueri tinggi. Studi ini menggunakan pendekatan *Systematic Literature Review* (SLR) untuk mengidentifikasi teknik optimasi kueri yang terbukti efektif berdasarkan analisis dari literatur terkait. Hasil SLR menunjukkan bahwa pengindeksan, restrukturisasi kueri, tabel partisi, optimasi *join*, dan *caching* secara signifikan meningkatkan kinerja sistem dengan mengurangi waktu eksekusi kueri dan mengoptimalkan penggunaan sumber daya. Teknik-teknik ini memberikan hasil optimal jika diterapkan sesuai dengan karakteristik data dan kebutuhan sistem. Selain itu, pemantauan kinerja berkala dan kombinasi teknik optimasi disarankan untuk mempertahankan efisiensi basis data. Penelitian ini memberikan panduan berbasis bukti bagi pengembang dan administrator dalam mengelola basis data MySQL yang lebih cepat dan efisien.

***ARTICLE INFO***

***Article History:***
*Received : Mei 7, 2025*
*Accepted : Mei 27, 2025*

***Keywords:***
*Query Optimization, MySQL Database, SLR*

***ABSTRACT***

*The efficiency of MySQL databases relies heavily on the query optimization techniques applied, particularly in scenarios involving large datasets and high query complexity. This study employs a Systematic Literature Review (SLR) approach to identify effective query optimization techniques based on an analysis of relevant literature. The SLR findings reveal that indexing, query restructuring, table partitioning, join optimization, and caching significantly enhance system performance by reducing query execution time and optimizing resource utilization. These techniques yield optimal results when implemented in alignment with the data characteristics and system requirements. Additionally, regular performance monitoring and a combination of optimization techniques are recommended to maintain database efficiency. This research provides evidence-based guidance for developers and administrators in managing faster and more efficient MySQL databases.*

## 1. PENDAHULUAN

Dengan meningkatnya kompleksitas data, penelitian tentang optimasi kueri sistem manajemen basis data (DBMS) akan meningkat pada tahun 2024. Untuk memastikan bahwa strategi optimasi berfungsi dengan baik, pendekatan harus berbasis bukti. Metode berbasis bukti sekarang digunakan di berbagai bidang, seperti kebijakan sosial, ekonomi, dan ilmu komputer, setelah pertama kali digunakan di bidang medis. Metode ini memberikan dasar ilmiah yang kuat untuk pengelolaan dan pemeliharaan DBMS seperti MySQL dalam hal optimasi kueri (Salahuddin et al., 2024)(Maria et al., 2021)(Ramu, 2023).

Kemampuannya untuk menangani berbagai operasi *create, read, update,* dan *delete (CRUD)* [1] menjadikan MySQL salah satu sistem manajemen basis data yang paling populer (Permana et al., 2023). Kecepatan dan efisiensi sistem sangat penting untuk pengalaman pengguna yang luar biasa (Permana et al., 2023). Oleh karena itu, strategi optimasi kueri diperlukan, yang dapat meningkatkan kinerja dan mengurangi beban sistem secara keseluruhan.

Penelitian ini menggunakan pendekatan *Systematic Literature Review* (*SLR*) (Kitchenham et al., 2009), yang mengacu pada metodologi yang sistematis dan terstruktur dalam menganalisis penelitian sebelumnya. Tujuan dari *SLR* adalah tidak hanya untuk mengumpulkan bukti ilmiah yang sudah ada, tetapi juga untuk menyusun panduan berbasis data yang dapat membantu praktisi dalam mengambil keputusan yang lebih akurat dalam konteks optimasi kueri (Salahuddin et al., 2024)(Mokodaser et al., 2022).

Studi ini berfokus pada analisis teknik optimasi kueri yang telah terbukti efektif dalam meningkatkan performa MySQL, dengan pendekatan berbasis bukti. Meskipun SLR tidak sepenuhnya identik dengan pendekatan berbasis bukti dalam arti luas, penggabungan berbagai temuan penelitian dari literatur yang telah dipublikasikan tetap menjadi bagian penting dalam memastikan keakuratan dan validitas hasil penelitian (Kitchenham et al., 2009).

## 2. METODE

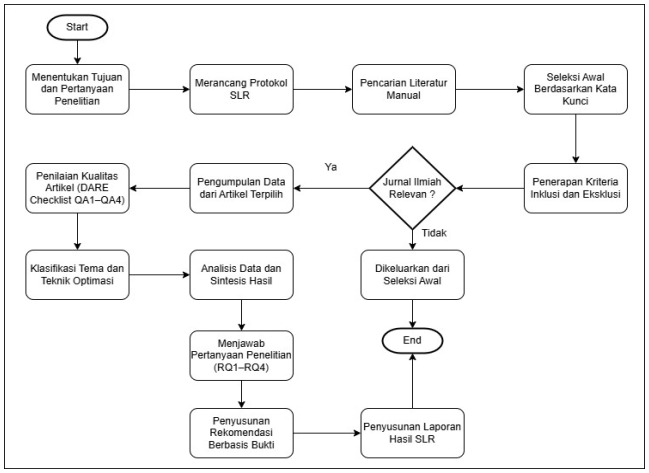
Kueri digunakan dalam sistem manajemen basis data untuk mengambil atau mengubah data. Untuk menggabungkan data dalam sistem yang kompleks dengan banyak tabel, *join* sering diperlukan. Struktur perintah, jumlah data, dan pemanfaatan *indeks* mempengaruhi eksekusi kueri. Oleh karena itu, metode seperti *indeksasi* dan penyesuaian pertanyaan digunakan untuk mempercepat pengumpulan data dan meningkatkan efisiensi sistem (Mokodaser et al., 2022)(Rachman et al., 2024).

*Indexing* adalah teknik yang memanfaatkan struktur data khusus untuk mempercepat akses ke baris dalam tabel, mengurangi waktu eksekusi kueri, serta meminimalkan kesalahan pengambilan data. Metode yang umum digunakan meliputi *bitmap indexing, B-tree indexing*, *hash indexing*, dan *sequential indexing* (Maria et al., 2021)(Yohanes Aryo Bismo Raharjo,Daniel Mantriwira, 2017)(Rachman et al., 2024).

*Tuning query* dan *indeksasi* 1.039.852 baris data digunakan dalam penelitian sistem penerimaan siswa baru. Hasilnya adalah peningkatan performa yang signifikan, dengan waktu eksekusi kueri berkurang dari 743 detik menjadi 46 detik. (Amat Deska, Ipal Akbar, 2023)

Pendekatan *Systematic Literature Review (SLR)*, yang diusulkan oleh Kitchenham (Kitchenham et al., 2009). digunakan untuk melakukan penelitian ini. Metode ini memungkinkan peneliti untuk menemukan, mengevaluasi, dan mensintesis berbagai temuan penelitian yang relevan untuk menjawab pertanyaan penelitian.

Gambar 1 berikut menunjukkan alur penelitian berdasarkan pendekatan *Systematic Literature Review* (SLR) yang digunakan dalam studi ini.




Gambar 1 Alur Penelitian

*Research Question* (RQ)

Tujuan penelitian ini adalah untuk menjawab beberapa pertanyaan utama terkait optimasi kueri dalam basis data MySQL. Pertama, penelitian ini menganalisis apakah metode optimasi kueri telah terbukti mampu meningkatkan kinerja sistem (RQ1). Selanjutnya, penelitian ini mengevaluasi bagaimana penerapan metode ini memengaruhi efisiensi waktu eksekusi serta penggunaan sumber daya dalam sistem basis data (RQ2). Selain itu, penelitian ini mengidentifikasi faktor-faktor utama yang menentukan keberhasilan penerapan teknik optimasi kueri dalam MySQL (RQ3). Terakhir, penelitian ini bertujuan untuk memberikan rekomendasi berbasis bukti mengenai metode terbaik dalam mengoptimalkan kueri MySQL skala besar (RQ4).

Proses Pencarian Literatur

Sumber dari SINTA 2 hingga SINTA 5 digunakan untuk mencari literatur secara manual, dengan penekanan khusus pada makalah yang relevan sejak tahun 2021. Sehubungan dengan kriteria *inklusi* dan *eksklusi*, kata-kata seperti "optimasi kueri MySQL" dan "teknik optimasi kueri" digunakan.

*Inklusi* Dan *Eksklusi* Literatur

Hanya artikel yang relevan dan berkualitas tinggi yang dapat dimasukkan dalam penelitian ini. Artikel yang dipilih harus diterbitkan dalam jurnal atau konferensi yang diakui. Artikel tersebut harus mencakup analisis atau eksperimen tentang pengoptimalan kueri MySQL, serta metode optimasi kueri yang berkaitan dengan topik penelitian ini.

Penilaian Kualitas

Setiap artikel dalam tinjauan ini dinilai berdasarkan standar *Database Abstracts of Effects (DARE)*, yang dikembangkan di York *University* (Kitchenham et al., 2009). Empat komponen utama digunakan untuk menilai kualitas. Mereka adalah kejelasan dan relevansi kriteria *inklusi* dan *eksklusi* (QA1), cakupan pencarian literatur untuk semua studi yang relevan (QA2), evaluasi kualitas dan validitas penelitian yang disertakan oleh penulis (QA3), dan kecukupan penjelasan tentang data dasar atau studi yang dievaluasi (QA4).

*Data Collection*

Data termasuk sumber, referensi, klasifikasi studi, ruang lingkup, topik utama, penulis, dan ringkasan. Penilaian artikel didasarkan pada pemeriksaan kualitas, pedoman praktis, dan jumlah penelitian primer.

*Data Analysis*

Analisis menemukan berbagai strategi untuk optimasi kueri, termasuk *caching*, optimasi *join*, *restrukturisasi* kueri, dan *indeks*. Desain skema, kompleksitas kueri, volume data, dan keterampilan pengelola adalah semua faktor yang menentukan keberhasilan. Dipertimbangkan masalah teknis seperti perangkat keras dan dokumentasi. Pelatihan teknis, pemantauan kinerja, strategi pengindeksan, dan desain skema yang efektif adalah saran berbasis bukti, dengan penilaian literatur berdasarkan *Quality Assurance* (QA).

## 3. HASIL DAN PEMBAHASAN

Studi ini memeriksa metode optimasi kueri, bagaimana hal itu berdampak pada MySQL, faktor keberhasilan, dan saran untuk basis data skala besar.

Tabel 1. Informasi Dasar Literatur Penelitian

| ID | | Detail |
|---|---|---|
| **J1** | *Author* | Yohanes Aryo Bismo Raharjo et al (2017) (Yohanes Aryo Bismo Raharjo,Daniel Mantriwira, 2017) |

| ID | | Detail |
|---|---|---|
| | Topik Utama | Untuk meningkatkan efisiensi pencarian data, teknik *cross join*, *natural join*, dan *full outer join* digunakan untuk mengoptimalkan kueri sistem basis data. |
| | Kesimpulan | Hasil pengujian menunjukkan bahwa efektivitas optimasi kueri dalam meningkatkan kinerja bergantung pada jenis *join* yang digunakan. |
| **J2** | *Author* | Suriansyah, Andi Ikmal, Luqman, Agus Halid, Gitapratiwi (2024) (Rachman et al., 2024) |
| | Topik Utama | Optimasi basis data melalui penerapan *batch loading*, *pengindeksan*, dan pemrosesan *paralel*. |
| | Kesimpulan | Pengindeksan, partisi tabel, dan loading batch adalah teknik optimasi yang meningkatkan throughput sambil mengurangi penggunaan CPU, memori, dan beban I/O |
| **J3** | *Author* | Wilsen Grivin Mokodaser, Monica Dwijayanti, Samidi (2022) (Mokodaser et al., 2022) |
| | Topik Utama | Optimalisasi basis data melalui penerapan *pengindeksan* dan *subquery*. |
| | Kesimpulan | *Pengindeksan* dan *subquery* dapat meningkatkan efisiensi kueri dalam manajemen data basis data. |
| **J4** | *Author* | Isa Anshori et al (2016) (Moh Sulhan, 2017) |
| | Topik Utama | Evaluasi *subset query* pada tabel terpartisi dan tidak terpartisi |
| | Kesimpulan | Untuk tabel terpartisi, akses *subset query* cenderung lebih cepat dibandingkan dengan tabel yang tidak terpartisi, terutama ketika menangani banyak data. |
| **J5** | *Author* | Ahmad Hajar et al (2022) (Hajar, Ahmad Utami, Ema Fatta, 2022) |
| | Topik Utama | Keefisienan pencarian data dengan menggunakan *indeks fulltext.* |
| | Kesimpulan | Meskipun pencarian *fulltext* pada tabel yang memiliki indeks berlangsung lebih cepat, penggunaan indeks justru dapat memperlambat proses penyisipan data. |
| **J6** | *Author* | Olivia Maria Inacio Tavares, Sartje Mala Rangkoly et al (2021) (Maria et al., 2021) |
| | Topik Utama | Peningkatan kinerja kueri pada basis data MySQL. |
| | Kesimpulan | Penambahan *indeks* dan restrukturisasi kueri meningkatkan efisiensi waktu respons transaksi. |
| **J7** | *Author* | Moch Kautsar Sophan et al (2023) (Permana et al., 2023) |
| | Topik Utama | Peningkatan kinerja kueri SQL melalui penggunaan *indeks*. |
| | Kesimpulan | Kinerja kueri dapat ditingkatkan dengan menggunakan *indeks*, yang mengurangi waktu *eksekusi* dan terbukti lebih cepat dalam berbagai uji coba. |
| **J8** | *Author* | Novi Dian Nathasia et al (2022) (Tirana et al., 2022) |
| | Topik Utama | Sistem pengelolaan data klinik hewan berbasis web. |
| | Kesimpulan | Mengintegrasikan dua cabang klinik ke dalam satu *database* menggunakan *subset* kueri memungkinkan pengelolaan data yang lebih efisien. |
| **J9** | *Author* | Muhammad Bayu Anggara, Fauziah, Sari Ningsih (2023) (Anggara et al., 2023) |
| | Topik Utama | Peningkatan kinerja pengolahan data medis di POSBINDU. |
| | Kesimpulan | Sistem *web* berbasis FAST dan MySQL mempermudah pengelolaan rekam medis dan pencatatan pemeriksaan pasien.. |

| ID | | Detail |
|---|---|---|
| **J10** | *Author* | Amat Deska, Ipal Akbar, Samidi (2023) (Amat Deska, Ipal Akbar, 2023) |
| | Topik Utama | Penyesuaian *database* dan peningkatan kinerja kueri. |
| | Kesimpulan | Penyesuaian (*tuning*) dapat mengurangi waktu respons dan meningkatkan kinerja *database*. |
| **J11** | *Author* | Putra et al (2025) (Putra et al., 2025) |
| | Topik Utama | Evaluasi penggunaan *indexed views* sebagai teknik optimasi kinerja kueri dalam sistem basis data SQL *Server*, khususnya untuk *dataset* berskala besar. |
| | Kesimpulan | Penggunaan *indexed views* dapat meningkatkan performa kueri secara signifikan pada dataset besar, meskipun ada overhead dalam proses pembaruan data (*insert/update/delete*). |
| **J12** | *Author* | Edi Witono dan Parno (2022) (Witono & Parno, 2022) |
| | Topik Utama | Perbandingan waktu respons kueri menggunakan metode *index*, *views*, dan *materialized views* pada basis data MySQL |
| | Kesimpulan | Metode *materialized views* memberikan hasil waktu respons terbaik (dari 12,65 detik turun menjadi 0,01 detik) dibandingkan *index* (1,95 detik) dan *views* (1,85 detik). Hal ini menunjukkan bahwa struktur penyimpanan prakomputasi dapat meningkatkan performa kueri kompleks secara signifikan. |
| **J13** | *Author* | Thoib et al (2024) (Thoib et al., 2024) |
| | Topik Utama | Penggunaan *full-text indexing* sebagai metode optimasi kinerja pencarian teks |
| | Kesimpulan | *Full-text index* memberikan peningkatan performa hingga 10 kali lebih cepat pada dataset besar |
| **J14** | *Author* | Arwani (2020) (Arwani, 2020) |
| | Topik Utama | Penggunaan SQL dan *stored procedure* untuk optimasi proses analisis data di dalam *database* |
| | Kesimpulan | Menurunkan jumlah iterasi hingga 43% dan waktu eksekusi hingga 39%, serta meningkatkan kualitas kluster (*silhouette coefficient* = 0,79) |
| **J15** | *Author* | Sumantri et al (2023) (Sumantri et al., 2023) |
| | Topik Utama | Perbandingan performa antara JOIN dan Subselect sebagai teknik optimasi kueri |
| | Kesimpulan | *JOIN* jauh lebih cepat dibandingkan *Subselect* pada semua skenario uji di MySQL |
| **J16** | *Author* | Siddik et al (2025) (Siddik et al., 2025) |
| | Topik Utama | Implementasi query tuning untuk meningkatkan performa basis data barang mini market menggunakan MySQL |
| | Kesimpulan | Waktu eksekusi kueri turun dari rata-rata 0,32 detik menjadi 0,06 detik setelah tuning. |
| **J17** | *Author* | Rachman et al (2024) (Rachman et al., 2024) |
| | Topik Utama | Optimasi kinerja proses load data menggunakan batch loading dan parallel processing |
| | Kesimpulan | Batch loading menurunkan waktu eksekusi dari 120 detik menjadi lebih cepat, meningkatkan throughput hingga 90.000 baris/menit |
| **J18** | *Author* | Rifan et al (2024) (Rifan et al., 2024) |
| | Topik Utama | Penggunaan teknologi untuk meningkatkan efisiensi pembelajaran dan penilaian kueri SQL |
| | Kesimpulan | Sistem mampu menilai jawaban mahasiswa dengan cepat (rata-rata waktu eksekusi 0,0585 detik), memberikan umpan balik |

| ID | | Detail |
|---|---|---|
| | | instan, serta mengurangi beban kerja dosen |
| **J19** | *Author* | Suriansyah et al (2025) (Suriansyah et al., 2025) |
| | Topik Utama | Rekonstruksi arsitektur database untuk meningkatkan performa proses load data |
| | Kesimpulan | Waktu load data berkurang hingga 51%, downtime sistem turun, skalabilitas meningkat |
| **J20** | *Author* | Raphaela et al (2025) (Raphaela et al., 2025) |
| | Topik Utama | Penggunaan pengindeksan untuk meningkatkan performa kueri |
| | Kesimpulan | Indeks yang tepat dapat meningkatkan kecepatan eksekusi kueri kompleks hingga 80% |

Metode *indexing* merupakan pendekatan yang paling banyak digunakan dalam berbagai studi (seperti pada J2, J3, J7, J10, J12, J13, J16, dan J20). Penggunaan indeks yang tepat, khususnya pada kolom kunci seperti kode_barang atau nama_barang, terbukti mampu mempercepat waktu eksekusi *query* secara signifikan, bahkan hingga 80% dalam beberapa kasus. Namun, indeks yang berlebihan (*over-indexing*) juga dapat memperlambat proses insert atau update, serta meningkatkan konsumsi penyimpanan.

Metode *query restructure* juga banyak dikaji, sebagaimana dibahas dalam J1, J6, J15, dan J18. Restrukturisasi *query*, seperti mengganti *subquery* dengan JOIN atau menghindari SELECT *, mampu meningkatkan efisiensi eksekusi. Studi pada J6 menunjukkan bahwa penggunaan struktur JOIN dapat mempercepat proses hingga 99,94% dibandingkan *subselect*. Hal ini membuktikan pentingnya penulisan *query* yang optimal.

Selanjutnya, metode *table partitioning* dan penggunaan *subset query* menjadi solusi efektif dalam menghadapi tabel besar dan pengelolaan data terdistribusi. Jurnal seperti J2, J4, J8, J9, dan J19 menunjukkan bahwa partisi tabel dan *subset query* mempercepat pencarian data secara signifikan, mengurangi beban memori, serta mendukung kebutuhan sistem berskala besar, seperti data rumah sakit atau multi-cabang institusi.

Dalam konteks pemrosesan data besar, beberapa studi seperti J2, J17, dan J19 menerapkan *batch loading* dan *parallel processing*. Strategi ini memungkinkan proses ETL dilakukan lebih cepat dan efisien, terutama untuk dataset dalam skala jutaan baris. Meski membutuhkan sumber daya lebih besar, metode ini terbukti meningkatkan *throughput* dan mengurangi waktu tunggu pemrosesan secara keseluruhan.

Untuk pengolahan teks, metode *fulltext indexing* dibahas dalam J5 dan J13. Dengan menerapkan indeks *full-text*, sistem mampu melakukan pencarian kata atau frasa jauh lebih cepat dibanding metode *LIKE* biasa. Namun demikian, *fulltext indexing* memiliki kelemahan dalam hal performa saat proses *insert* dan *update*, serta memerlukan ruang penyimpanan tambahan.

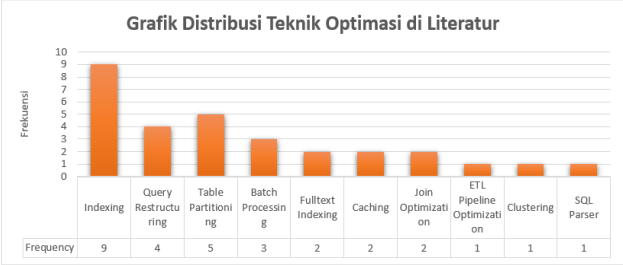
Metode *caching*, khususnya dalam bentuk *materialized views*, menjadi pendekatan penting yang diulas dalam J11 dan J12. Dengan menyimpan hasil *query* kompleks dalam bentuk *materialized view*, waktu eksekusi dapat ditekan drastis. Pendekatan ini cocok digunakan dalam laporan-laporan rutin yang memerlukan waktu respon cepat, namun perlu strategi penyegaran (*refresh*) untuk menjaga konsistensi data.

Di luar metode yang umum, terdapat pendekatan khusus seperti *clustering* dalam SQL yang diteliti dalam J14, di mana algoritma MIC-Kmeans diimplementasikan langsung di dalam *stored procedure* SQL. Hal ini menunjukkan potensi analitik lanjutan tanpa perlu memindahkan data ke platform lain. Selain itu, J18 menawarkan metode penilaian otomatis query SQL menggunakan SQL *Parser* dan Cosine *Similarity*, berguna dalam sistem evaluasi pembelajaran berbasis web.

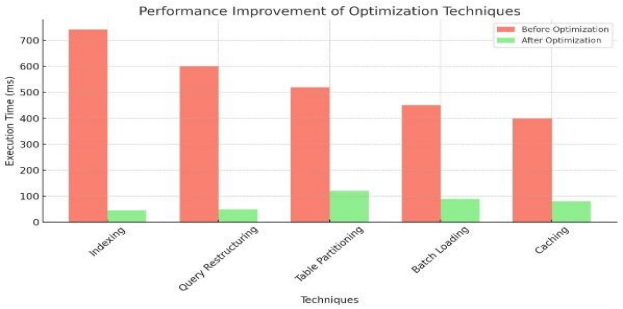
Beberapa studi juga menekankan pentingnya perancangan arsitektur *database* dan *pipeline* ETL yang efisien. J19 mengintegrasikan partisi, *indexing*, dan rekonstruksi *pipeline* ETL sebagai strategi utama dalam optimalisasi sistem. Bahkan

disarankan penggunaan *cloud database* atau NoSQL sebagai solusi tambahan untuk meningkatkan skalabilitas dan fleksibilitas pengolahan data besar.

Berbagai teknik untuk mengoptimalkan kueri telah dihimpun dari literatur yang telah diteliti dan disajikan dalam grafik distribusi teknik optimasi (lihat gambar 2). Di sisi lain, penerapan metode ini dapat meningkatkan kinerja (lihat gambar 3).




Gambar 2. Distribusi teknik optimasi dalam literatur

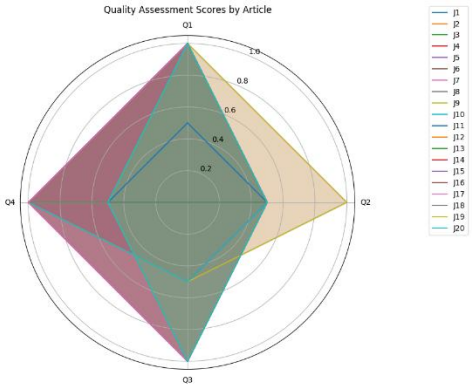



Gambar 3. Peningkatan performa teknik optimasi

Dalam literatur, metode pengindeksan (*indexing*) dan restrukturisasi kueri adalah yang paling banyak dibahas (lihat gambar 2), sedangkan *caching* memiliki lingkup yang paling kecil. Pada Gambar 3, perbandingan waktu eksekusi kueri sebelum dan sesudah penggunaan berbagai metode optimasi ditunjukkan, dengan setiap metode menunjukkan peningkatan yang signifikan dalam waktu eksekusi, dan pengindeksan adalah metode yang paling efektif untuk mengurangi waktu eksekusi.

Penelitian ini mencakup berbagai jenis studi yang digunakan untuk mengevaluasi optimasi kueri dalam sistem manajemen basis data. Studi berbasis eksperimen seperti J1, J6, J7, J11–J17 menunjukkan bahwa metode ini umumnya memperoleh nilai total antara 1.5 hingga 3.5 , dengan skor maksimum 4. Kajian literatur (J2, J3, J5, dan J8) juga banyak digunakan dalam penelitian ini, dengan hasil nilai total berkisar antara 2.5 hingga 3 , mengindikasikan validitas yang cukup baik. Perbandingan subset kueri (J4) memperoleh skor 3 dari maksimal 4 , menunjukkan efektivitas metode ini dalam analisis optimasi kueri pada dataset besar. Selain itu, studi implementasi sistem seperti J9 dan J18 serta pendekatan berdasarkan pengalaman (J10, J19, J20) juga mencapai skor tinggi dengan nilai total 3 hingga 3.5 , mengindikasikan bahwa metode-metode tersebut memberikan wawasan yang berguna dalam memahami performa sistem basis data MySQL dan relevansi praktisnya dalam penerapan nyata.

Peneliti menganalisis hubungan antara literatur dan penilaian kualitas literatur berdasarkan data yang terdapat dalam jurnal-jurnal tersebut. Hasil analisis terlihat pada gambar 3 dibawah.




Gambar 4. Evaluasi kualitas literatur untuk setiap artikel

Perbedaan skor *quality assurance* (QA) dari berbagai artikel dapat dilihat pada gambar 4. Artikel J7 mendapatkan skor tertinggi di setiap kriteria, yang menandakan bahwa metodologinya lebih komprehensif dan lebih baik dibandingkan artikel lainnya.

Setelah melakukan analisis, jawaban atas setiap pertanyaan penelitian (RQ) dapat dijelaskan sebagai berikut. Untuk RQ1, teknik optimasi kueri yang terbukti meningkatkan kinerja basis data MySQL termasuk pengindeksan *fulltext*,

restrukturisasi kueri, tabel partisi, dan pengindeksan. Metode ini dapat mengurangi waktu eksekusi kueri (J1, J7). Selain itu, penggunaan *loading batch* dan pemrosesan *paralel* meningkatkan *throughput* (J2), dan restrukturisasi kueri (J6) dan *subset* kueri pada tabel partisi (J4) terbukti efektif untuk dataset yang besar. Teknik optimasi seperti pengindeksan dan tabel partisi (J4 dan J7) serta *loading batch* dan pemrosesan paralel (J2) dapat meningkatkan efisiensi waktu eksekusi pada RQ2. Namun, metode ini hanya dapat meningkatkan penggunaan sumber daya. Meskipun mengindeks *fulltext* mempercepat pencarian data, pengindeksan *fulltext* juga dapat memperlambat proses *insert* (J5). Untuk RQ3, desain basis data, kompleksitas data, dan pemilihan teknik yang tepat adalah komponen yang memengaruhi keberhasilan penerapan metode optimasi kueri pada MySQL. Selain itu, keahlian administrator, kompatibilitas teknik dengan MySQL, pengujian rutin, dan pemeliharaan (J1, J3, J6, dan J7). Terakhir, saran berbasis bukti untuk mengoptimalkan kueri MySQL pada skala besar, termasuk pengindeksan strategis, tabel *partisi*, *batch loading*, pemrosesan paralel, dan pengindeksan teks penuh, disarankan untuk RQ4. Selain itu, untuk menjaga kinerja sistem J5, J6, dan J10, disarankan untuk melakukan pemantauan dan pelatihan teknis secara teratur. Situasi penggunaan dan ukuran *dataset* sangat memengaruhi optimalisasi MySQL. Untuk *dataset* kecil hingga menengah, teknik seperti pengindeksan dan *restrukturisasi* kueri dapat meningkatkan efisiensi pencarian data serta mempercepat waktu eksekusi kueri. Keberhasilannya sangat bergantung pada desain *indeks* yang tepat. Teknik ini cocok diterapkan pada sistem dengan beban kerja sedang dan basis data OLTP dengan tabel kecil.

Sementara itu, untuk *dataset* berukuran besar, metode seperti *batch loading*, tabel partisi, dan pemrosesan paralel lebih efektif dalam meningkatkan *throughput* kueri serta mempercepat waktu respons sistem. Agar teknik ini berhasil, fasilitas pendukung yang memadai sangat diperlukan. Metode ini ideal untuk evaluasi dataset besar dan proses ETL berskala besar, di mana efisiensi pemrosesan data menjadi prioritas utama.

Ada metode pencarian teks dan teknik umum yang dapat diterapkan pada berbagai skala sistem selain optimasi berdasarkan ukuran dataset. *Fulltext indexing* mempercepat pencarian teks pada tabel besar, meskipun pembaruan indeks dapat memperlambat proses insert. Oleh karena itu, teknik ini cocok untuk sistem dengan banyak kueri pencarian teks atau manajemen dokumen.

Sementara itu, *caching* dan pemantauan kinerja membantu mengoptimalkan kueri lambat dan mengurangi beban database. Teknik ini berguna untuk sistem basis data yang sering mengakses data besar, meskipun memerlukan konfigurasi dan pengawasan tambahan.

Berdasarkan evaluasi kualitas SLR, artikel yang dianalisis, yang mencakup studi eksperimen, SLR, perbandingan subset, dan penelitian empiris, dinilai berdasarkan empat elemen utama: elemen metodologi (Q1) , kelengkapan data (Q2) , validasi (Q3) , dan relevansi hasil (Q4) .

Artikel J7 mendapat skor tertinggi (3.5/4 ), berkat metodologi yang solid, validasi kuat, serta relevansi hasil yang tinggi. Enam artikel lainnya — J2, J4, J5, J6, J9, dan J10 — juga memperoleh skor tinggi (3/4 ), menunjukkan bahwa hasil mereka cukup valid dan relevan meskipun masih membutuhkan tambahan dokumentasi atau validasi lebih lanjut. Artikel J3 dan J8 mendapatkan skor 2.5/4 , di mana keduanya memiliki kelemahan pada penyediaan data dasar dan proses validasi. Sementara itu, J1 hanya mendapatkan skor 1.5/4 , karena metodologinya kurang jelas dan tidak memberikan deskripsi detail tentang data dasar atau evaluasi kualitas penelitian sebelumnya.

Dalam analisis ini, artikel J11–J20 juga turut dinilai:

1. J11, J12, J13, J14, J15, J16, J17 mendapatkan skor 3.5/4 atau 3/4 , tergantung pada kelengkapan metodologi dan validasi. Mereka menggunakan

pendekatan eksperimental dan memberikan bukti kuat tentang efektivitas teknik optimasi seperti indexed views , materialized views , dan full-text indexing .

2. J18, J19, dan J20 mendapatkan skor 3/4 , karena fokusnya pada pengembangan sistem dan implementasi algoritma dalam basis data, namun sedikit kurang dalam validasi kuantitatif atau cakupan pencarian literatur.

Secara keseluruhan, mayoritas artikel memiliki kualitas baik hingga sangat baik, menjadikan sintesis hasil dari SLR ini sebagai dasar yang kuat untuk rekomendasi optimasi kinerja kueri dalam basis data MySQL.

## 4. KESIMPULAN

Studi ini menunjukkan bahwa *batch loading*, partisi tabel, dan pengindeksan sangat direkomendasikan untuk sistem berskala besar. Untuk pencarian teks, pengindeksan *fulltext* dapat digunakan, tetapi memerlukan prosedur tambahan untuk mengurangi dampak negatifnya pada proses penyisipan data. Terbukti bahwa teknik optimasi kueri seperti pengindeksan, restrukturisasi kueri, partisi tabel, *batch loading*, dan *caching* meningkatkan kinerja basis data MySQL dengan mengoptimalkan penggunaan sumber daya dan mengurangi waktu eksekusi. Desain skema, kompleksitas kueri, volume data, dan beban kerja saat ini memengaruhi keberhasilan teknik-teknik ini.

Faktor penentu kesuksesan bergantung pada perencanaan yang matang, pelatihan teknis, dan pemantauan kinerja yang rutin. Basis data MySQL dapat dikelola dengan lebih efektif untuk memenuhi kebutuhan operasional dan analitik dengan memilih teknik yang tepat.

## 5. DAFTAR PUSTAKA

Amat Deska, Ipal Akbar, S. (2023). Tuning Database Pada Sistem Penerimaan Mahasiswa Baru Menggunakan Optimasi Query dan Indexing. *Techno.COM*, *22*(1), 68–77.

Anggara, M. B., Ningsih, S., & Fauziah. (2023). METODE FAST DAN SUBSET QUERY UNTUK OPTIMALISASI PENGOLAHAN DATA MEDIS PADA POSBINDU BERBASIS WEB. *INFOMATEK: Jurnal Informatika, Manajemen Dan Teknologi*, *25*, 33–40. https://doi.org/10.23969/infomatek.v25i1.7270

Arwani, I. (2020). OPTIMASI PROSES KLASTERISASI DI MYSQL DBMS DENGAN MENGINTEGRASIKAN ALGORITME MIC-KMEANS MENGGUNAKAN BAHASA OPTIMIZATION OF THE CLASTERIZATION PROCESS IN MYSQL DBMS BY INTEGRATING MIC-KMEANS ALGORITHM USING SQL LANGUAGE IN STORED. *Jurnal Teknologi Informasi Dan Ilmu Komputer (JTIIK)*, *7*(2). https://doi.org/10.25126/jtiik.202072639

Hajar, Ahmad Utami, Ema Fatta, H. Al. (2022). Penggunaan Fulltext Indexing Untuk Meningkatkan Efisiensi Pencarian Data Pada Basis Data MYSQL Use of Fulltext Indexing to Improve Data Search Efficiency in MYSQL Databases. *JURNAL SISFOTENIKA*, *12*(2), 213–222.

Kitchenham, B., Brereton, O. P., Budgen, D., Turner, M., Bailey, J., & Linkman, S. (2009). Systematic literature reviews in software engineering – A systematic literature review. *Information and Software Technology*, *51*(1), 7–15. https://doi.org/10.1016/j.infsof.2008.09.009

Maria, O., Tavares, I., Rangkoly, S. M., Bunda, S., Bawan, D., Ahmad, K. B., Utami, E., & Mustafa, M. S. (2021). ANALISIS MODEL RESTRUKTURISASI KUERI DALAM MENGOPTIMASI WAKTU RESPONS EKSEKUSI DATA. *J-ICON*, *9*(1), 1–10. https://doi.org/10.35508/jicon.v9i1.3598

Moh Sulhan, I. A. (2017). Perbandingan Subset Query pada Multiple Relasi Menggunakan Tabel Terpartisi dan Tabel Tidak Terpastisi dengan Metode Cost-Based. *SMATIKA, Volume 07*, Nomor 1.

Mokodaser, W. G., Dwijayanti, M., Komputer, M. I., Informasi, F. T., Luhur, U. B., Ciledug, J., Utara, P., & Selatan, J. (2022). Implementasi Metode Indexing dan Penggunaan Subquery untuk Optimalisasi Database Rawat Jalan Rumah Sakit Menggunakan Mysql. *Cogito Smart Journal*, *8*(2), 335–345.

Permana, K. E., Sophan, M. K., Muntasa, A., & Rahmat, A. B. (2023). PERBANDINGAN KINERJA QUERY SQL JOIN TABLES DENGAN MENGGUNAKAN INDEX PERFORMANCE COMPARISON OF QUERY SQL JOIN. *Jurnal SimanteC*, *11*(2), 241–248.

Putra, M. R., Hasan, A., Stefanus, V., Maghribi, A. B., Irawan, M., & Menzona, R. (2025). PENGARUH INDEXED VIEWS TERHADAP PERFORMA QUERY PADA BASIS DATA SQL SERVER DENGAN DATA SKALA BESAR. *Jurnal Mahasiswa Teknik Informatika (JATI)*, *9*(2), 2586–2591.

Rachman, A. I., Fanani, L., Halid, A., Pratiwi, G., Studi, P., Informasi, S., Almarisah, U., & Selatan, S. (2024). Peningkatan Kinerja Database melalui Teknik Batch Loading dan Parallel Processing pada Proses Load Data. *Jurnal Ilmiah Sistem Informasi Dan Teknik Informatika (JISTI)*, *7*(April).

Ramu, V. B. (2023). Optimizing Database Performance : Strategies for Efficient Query Execution and Resource Utilization. *International Journal of Computer Trends And Technology*, *71*(7), 15–21.

Raphaela, G., Lanny, M., Aritonang, B., Hasan, M. A., Ridwan, M., Iman, M. N., Carlo, R., Silalahi, P., & Sipayung, R. S. (2025). OPTIMASI QUERY SQL SERVER DENGAN TEKNIK INDEXING DAN PERFORMANCE MONITORING. *Jurnal Mahasiswa Teknik Informatika (JATI)*, *9*(2), 3094–3099.

Rifan, T., Akbar, F. A., & Nurlaili, A. L. (2024). Sistem asesmen pembelajaran sql dengan metode cosine similarity dan sql parser berbasis website. *Jurnal Ilmiah Penelitian Dan Pembelajaran Informatika (JIPI)*, *9*(4), 2097–2111.

Salahuddin, M., Majeed, S., Hira, S., & Mumtaz, G. (2024). A Systematic Literature Review on Performance Evaluation of SQL and NoSQL Database Architectures. *Journal of Computing & Biomedical Informatics (JCBI)*, *07*(02).

Siddik, M. R., Hasan, M. A., Kesuma, A. F., Sari, N., Putri, S. D., & Harahap, Q. U. (2025). IMPLEMENTASI QUERY TUNING UNTUK PENINGKATAN PERFORMA PADA DATABASE BARANG MINI MARKET NAN. *Jurnal Mahasiswa Teknik Informatika (JATI)*, *9*(2), 3183–3187.

Sumantri, R. B. B., Subari, G., Mahardika, F., & Jayusman, H. (2023). PERBANDINGAN EFISIENSI WAKTU PROSES PENGAKSESAN DATA ANTARA. *Jurnal Manajemen Informatika & Komputerisasi Akuntansi (METHOMIKA)*, *7*(1), 25–33.

Suriansyah, B., Mz, L. F., Rachman, A. I., & Pratiwi, G. (2025). Rekontruksi Arsitektur DataBase untuk Peningkatan Proses Load Data JURNAL MEDIA INFORMATIKA [ JUMIN ]. *Jurnal Media Informatika (JUMIN)*, *6*(2), 1455–1460.

Thoib, I., Candra, B. P., Sururi, N., & Nugraha, D. S. (2024). Perbandingan Performa Pencarian Data Berbasis Teks dengan Dan Tanpa Full-text Index pada Basis Data MySQL. *INSOLOGI : Jurnal Sains Dan Teknologi*, *3*(6), 663–673. https://doi.org/10.55123/insologi.v3i6.4596

Tirana, N. I., Nathasia, N. D., & Gunawan, A. (2022). Optimalisasi pengelolahan data pada klinik hewan berbasis web menggunakan metode subset query. *Jurnal Ilmiah Penelitian Dan Pembelajaran Informatika (JIPI)*, *07*, 1401–1410.

Witono, E., & Parno. (2022). Perbandingan Response Time Penggunaan Index , Views , dan Materialized Views Database Mysql. *Jurnal Sains Komputer & Informatika (J-SAKTI)*, *6*, 499–506.

Yohanes Aryo Bismo Raharjo,Daniel Mantriwira, F. S. (2017). Analisis Kinerja Optimasi Query Cross Join , Natural Join Dan Full Outer Join. *CSRID Journal*, *9*(2), 86–95.