

IMPLEMENTASI INDEX MYSQL DAN SERVER-SIDE DATATABLES UNTUK OPTIMALISASI PEMROSESAN DATA SISTEM MBKM FIK BERBASIS CODEIGNITER4

IMPLEMENTATION OF MYSQL INDEX AND SERVER-SIDE DATATABLES FOR DATA PROCESSING OPTIMIZATION OF CODEIGNITER4-BASED MBKM FIK SYSTEM

Renjiro Azhar Pramono^{*1}, Filmada Ocky Saputra², Gustina Alfa Trisnapradika³

^{*}Email: 111202012428@mhs.dinus.ac.id

^{1, 2, 3} Program Studi S1 Teknik Informatika, Universitas Dian Nuswantoro, Semarang, Indonesia

Abstrak

Penelitian ini dilatar belakangi untuk mengatasi permasalahan yang dihadapi oleh Sistem Merdeka Belajar-Kampus Merdeka (MBKM) di Fakultas Ilmu Komputer Universitas Dian Nuswantoro (FIK UDINUS). Kendala utama yang dihadapi oleh sistem ini adalah pemrosesan data pengguna yang sangat banyak, yang berdampak negatif pada efisiensi dan performa sistem. Penelitian ini bertujuan untuk mengoptimalkan pemrosesan data pengguna dalam sistem MBKM FIK UDINUS dengan menerapkan *Index* pada basis data *MySQL* dan *Server-side DataTables*. Dengan demikian, penelitian ini bertujuan untuk meningkatkan efisiensi dan performa sistem. Penelitian ini menggunakan metode pengembangan *Extreme Programming (XP)*. Hasil penelitian menunjukkan peningkatan yang signifikan dalam efisiensi pemrosesan data pengguna sistem MBKM FIK UDINUS, adanya peningkatan rata-rata sebesar 25,86% dalam performa sistem, serta peningkatan efisiensi kueri rata-rata sebesar 94,06%. Terdapat peningkatan yang mencolok dalam performa sistem, terutama dalam pengambilan data dari basis data dan waktu respon sistem. Berdasarkan temuan penelitian, dapat disimpulkan bahwa penerapan *Index MySQL* dan *Server-side DataTables* efektif dalam meningkatkan performa sistem MBKM FIK UDINUS. Hasil positif ini berkontribusi pada pengalaman pengguna yang lebih baik dan efisiensi sistem. Rekomendasi dari penelitian ini dapat diterapkan dalam lingkungan sejenis untuk meningkatkan efisiensi dan performa sistem serupa.

Kata kunci: *DataTables, Index, MBKM, Sistem Informasi, Server.*

Abstract

This research has been launched to address the problems faced by the Merdeka Learning-Campus (MBKM) at the Faculty of Computer Science at Dian Nuswantoro University. (FIK UDINUS). The main obstacle faced by this system is the excessive processing of user data, which has a negative impact on the efficiency and performance of the system. This research aims to optimize the processing of user data in the FIK UDINUS MBKM system by applying *Index* on *MySQL* database and *Server-side DataTables*. Thus, this research aims to improve the efficiency and performance of the system. The study uses the *Extreme Programming (XP)* development method. The results showed a significant improvement in the data processing efficiency of users of the FIK UDINUS MBKM system, an average improvement of 25.86% in system performance, as well as an average increase in query efficiency by 94.06%. Based on the findings of the research, it can be concluded that the application of *Index MySQL* and *Server-side DataTables* is effective in improving the performance of the FIK UDINUS MBKM system. These positive results contribute to better user experience and system efficiency. Recommendations from this research can be applied in similar environments to improve the efficiency and performance of similar systems.

Keywords: *DataTables, Index, MBKM, Information System, Server*

I. PENDAHULUAN

Pada jaman ini, teknologi tidak akan dapat dipisahkan kaitannya dengan kehidupan sehari-hari manusia. Pengembangan teknologi pendidikan merupakan hal yang sangat menentukan keberhasilan suatu bangsa agar dapat bersaing di era globalisasi [1], [2]. Perkembangan teknologi dalam dunia pendidikan memiliki makna yang sangat penting bagi perkembangan mahasiswa, karena mahasiswa merupakan generasi muda calon pemimpin bangsa di masa depan [3], [4]. Di era digital saat ini, perkembangan teknologi telah banyak berpengaruh dalam dunia Pendidikan [1], [5], [6], [7], [8]. Teknologi komputer sangat mendukung pengolahan data untuk menciptakan sistem informasi yang memberikan hasil kerja yang maksimal [9]. Oleh karena itu instansi juga dituntut untuk terus berkembang mengikuti zaman.

Sistem Merdeka Belajar-Kampus Merdeka (MBKM) Fakultas Ilmu Komputer Universitas Dian Nuswantoro (FIK UDINUS) adalah sistem yang membantu dalam proses pengajuan surat dan berkas yang menjadi syarat mengikuti program Kampus Merdeka dari Kemdikbudristek RI, yang tertuang dalam kebijakan Merdeka Belajar-Kampus Merdeka sesuai dengan Permendikbud Nomor 3 Tahun 2020 tentang Standar Nasional Pendidikan Tinggi [10]. Sistem yang baik akan berpengaruh terhadap segala informasi yang akan dihasilkan [11]. Sistem MBKM FIK UDINUS menghadapi sejumlah tantangan dalam pemrosesan data pengguna yang banyak. Deska [12] dalam penelitiannya melakukan peningkatan performa sistem hanya dilakukan pada sisi basis data saja dengan teknologi *Indexing* kemudian ditingkatkan lagi dengan adanya penerapan *Server-Side DataTables* pada penelitian ini. Hal yang membedakan penelitian ini dengan penelitian sebelumnya adalah implementasi teknik *Index* yang dikombinasikan dengan *Server-side DataTables* untuk optimalisasi performa dan efisiensi sistem dengan metode pengembangan *Extreme Programming*.

Tujuan utama dari penelitian ini adalah untuk meningkatkan performa dan efisiensi pemrosesan data pengguna dalam sistem MBKM FIK UDINUS melalui penerapan teknologi *Index* pada basis data *MySQL* dan *Server-side DataTables* serta untuk memberikan dasar ilmiah yang kuat bagi pengembangan sistem serupa di institusi pendidikan tinggi lain yang mungkin menghadapi tantangan serupa. Dengan demikian, penelitian ini diharapkan dapat memberikan kontribusi pada pengembangan

sistem yang efisien dan efektif dalam menangani data dalam jumlah besar dan kompleks [13] di UDINUS, serta memperkaya pemahaman kita tentang pengelolaan sistem serupa dalam konteks pendidikan tinggi.

II. TINJAUAN PUSTAKA

A. Index

Index dapat diartikan sebagai struktur data fisik yang terpisah yang memungkinkan kueri dapat mengakses satu atau lebih baris data dengan cepat [12]. *Index* dapat digunakan untuk efisiensi sumber daya pada basis data. Untuk menggunakan *Index* pada basis data *MySQL* diperlukan kueri *ALTER* untuk melakukan modifikasi pada tabel di basis data.

B. MySQL

MySQL adalah basis data relasional yang memungkinkan penyimpanan data terstruktur, seperti data proyek, data sensor, data dari database lain, dll [14]. *MySQL* dikembangkan oleh *Oracle* dan dirilis perdana pada tahun 1995. *MySQL* merupakan basis data gratis dengan lisensi *General Public License (GPL)*.

C. Server-side

Server Side merupakan teknologi *webpage* yang menerapkan jenis pemrograman *web* yang di proses di *server* [15]. Ini mencakup proses pengolahan, manipulasi, dan pengiriman data sebelum tampil di peramban pengguna.

D. DataTables

DataTables adalah *plug-in* untuk *jQuery Javascript library* untuk interaksi pada tabel *HTML* [16]. *DataTables* memungkinkan pengguna mengelola dan menampilkan data pada tabel secara interaktif dengan fitur-fiturnya antara lain *search*, *pagination*, dan lain sebagainya dengan teknologi *AJAX*.

E. Apache JMeter

Apache JMeter adalah perangkat lunak *open source*, 100% aplikasi *Java* murni dirancang untuk memuat tes perilaku fungsional dan mengukur kinerja sistem [17]. *JMeter* digunakan untuk mensimulasikan beban berat pada *server*, objek, atau kelompok jaringan atau untuk menganalisis kinerja keseluruhan aplikasi *web* di bawah banyak jenis beban. *JMeter* dapat menganalisis kemampuan untuk memuat dan melakukan tes dari banyak aplikasi yang berbeda seperti basis data, *middleware* yang berorientasi pesan, dan protokol internet [18].

F. dbForge Studio

dbForge Studio adalah *tools* yang juga bisa digunakan untuk menghasilkan visualisasi *database SQL* [19]. Ini adalah *GUI MySQL* yang lengkap yang membantu membuat dan mengeksekusi kueri, mengembangkan dan memecahkan rutinitas yang disimpan, mengotomatisasi manajemen objek basis data, membandingkan dan menyinkronkan basis data, menganalisis data tabel, dan banyak lagi.

G. Load Testing

Load testing adalah teknik *performance testing* yang mana respon sistem diukur dalam berbagai *load condition* [20]. *Load testing* diperlukan untuk membuat simulasi akses aplikasi *web/website* secara simultan. Cara ini lebih baik dibandingkan dengan harus mengundang sekian belas, atau puluh orang sekaligus untuk mengakses sebuah *website* [20].

H. Backend

Backend adalah bagian untuk administrator agar bisa mengelola sebuah data perusahaan [21]. *Backend* merupakan bagian dari aplikasi yang bertanggung jawab untuk menyediakan kebutuhan yang tak terlihat oleh pengguna (tidak berinteraksi langsung dengan pengguna), seperti bagaimana data disimpan, diolah, serta ditransaksikan secara aman. Itu semua bertujuan untuk mendukung aplikasi *frontend* bekerja sesuai dengan fungsinya.

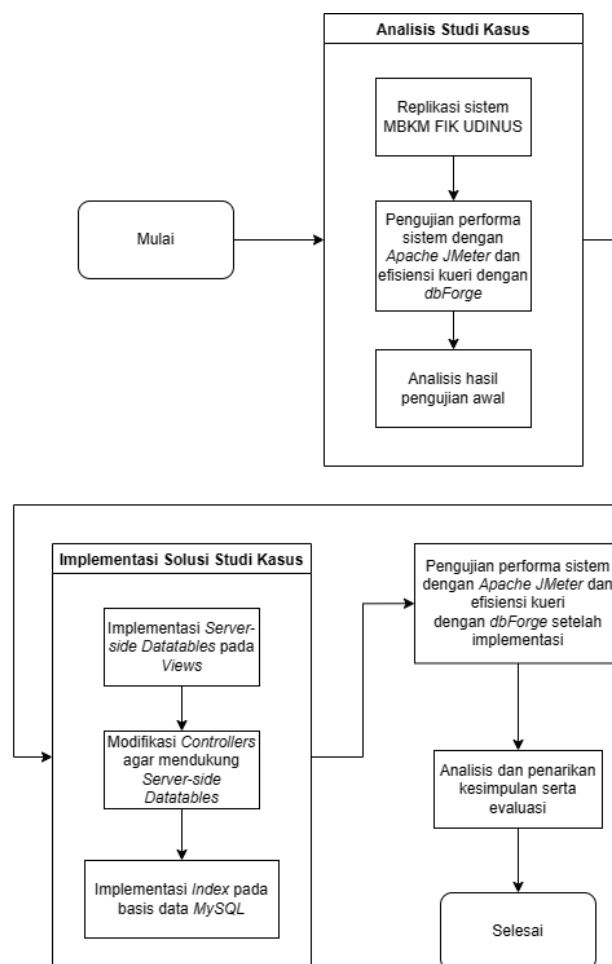
III. METODE

Penelitian ini dibagi menjadi beberapa tahap. Tahap yang dilakukan untuk menyelesaikan permasalahan penelitian antara lain pengumpulan data dengan pengujian sistem, analisis, tahapan implementasi solusi, dan pengujian sistem informasi [22]. Data tersebut ditindaklanjuti untuk kemudian di analisis. Pengumpulan data yang dilakukan bersifat kuantitatif karena data yang terkumpul dapat diukur dan dianalisis secara statistik [23], [24]. Penulis kemudian melakukan perancangan pengembangan sistem yang berlandaskan penelitian sebelumnya. Penelitian ini dilaksanakan di Fakultas Ilmu Komputer, Universitas Dian Nuswantoro, Semarang, Provinsi Jawa Tengah, selama semester Genap bulan Agustus 2023.

Dalam penelitian ini, penulis memilih metodologi pengembangan perangkat lunak *Extreme Programming (XP)*, metode ini digunakan untuk dapat secara tepat mengatasi permasalahan [25]. Salah satu keuntungan utama menggunakan *XP*

adalah fleksibilitasnya dalam menghadapi perubahan kebutuhan [26]. Dalam pengembangan sistem, kebutuhan seringkali berubah seiring waktu. Dengan *XP*, tim pengembang dapat secara cepat menyesuaikan diri dengan perubahan tersebut.

Skema pengujian pertama yang digunakan adalah *Performance Testing* pada bagian sistem dimana ditetapkan skenario dengan jumlah pengguna simultan yang tinggi untuk mengukur respons sistem dan waktu tanggap pada tingkat beban tinggi untuk kemudian di analisis apakah sistem dapat menangani beban tinggi tanpa mengalami penurunan kinerja yang signifikan. Kemudian skema pengujian kedua adalah *Efficiency Testing* pada basis data dimana kueri yang dijalankan oleh basis data diukur waktu respons dan efisiensi sumber daya yang digunakan ketika dieksekusi.



Gambar 1. Kerangka Penelitian

Dari kerangka penelitian pada Gambar 1 di atas, penulis kemudian melakukan penelitian kali ini dengan rincian proses kerja seperti berikut :

A. Replikasi sistem MBKM FIK UDINUS

Langkah awal dalam penelitian ini adalah mereplikasi sistem MBKM FIK UDINUS. Dengan menciptakan salinan sistem yang representatif, penulis dapat memeriksa dan mengidentifikasi permasalahan yang ada dengan lebih baik. Penulis melakukan replikasi dengan mengkloning kode sumber sistem, instalasi *package*, lalu memigrasi basis data yang telah diatur di dalam sumber kode.

B. Pengujian performa sistem dengan Apache JMeter dan efisiensi kueri dengan dbForge

Setelah berhasil mereplikasi sistem, penulis melakukan uji coba terhadap performa sistem dan efisiensi basis data tahap awal sebelum diimplementasikannya *Server-side DataTables* dan *Index* pada basis data *MySQL* dengan menggunakan *Apache JMeter* dan *dbForge Studio*. Tujuan dari penelitian ini adalah untuk mengetahui performa [27]. Alasan penulis menggunakan aplikasi tersebut karena *JMeter* merupakan aplikasi *load testing* yang gratis serta memiliki banyak fitur. Salah satu fitur yang menarik dari aplikasi tersebut adalah dapat membuat laporan pengetesan baik berbentuk tabel maupun grafik [28]. Sedangkan *dbForge Studio* dipilih oleh penulis karena memiliki *GUI* yang mudah dipahami serta fitur yang lengkap.

C. Analisis hasil pengujian awal

Analisis data dari pengujian tersebut mengungkap sejumlah masalah yang perlu diperhatikan dengan cermat. Pertama-tama, salah satu masalah utama yang terdeteksi adalah keterlambatan dalam sistem. Keterlambatan ini mengacu pada waktu yang diperlukan sistem untuk merespons permintaan pengguna. Hal ini sangat penting dalam pengalaman pengguna karena pengguna mengharapkan sistem berjalan dengan responsif dan cepat. Dengan adanya keterlambatan ini, pengguna mungkin merasa frustrasi dan tidak puas dengan kinerja sistem.

Masalah lain yang muncul adalah penggunaan sumber daya yang boros. menggunakan sumber daya pada basis data dengan tidak efisien. Penggunaan sumber daya yang tidak efisien dapat berdampak pada kinerja keseluruhan sistem dan dapat mengakibatkan penurunan responsivitas. Selain itu, penggunaan sumber daya yang boros juga dapat meningkatkan biaya operasional sistem.

D. Implementasi Server-side DataTables pada Views

Langkah selanjutnya adalah mengimplementasikan *Server-side DataTables* pada tampilan (*Views*) sistem yang dapat dilihat pada Gambar 2. Ini melibatkan modifikasi pada kode basis sistem untuk mendukung penggunaan *DataTables*. Implementasi ini akan mengubah cara sistem berinteraksi dengan data pengguna.

```
$(document).ready(function () {
    // Inisialisasi DataTables
    window.table = $('#table').DataTable({
        responsive: true,
        stateSave: true,
        processing: true,
        serverSide: true,
        ajax: {
            url: "<?="
base_url('route_ke_controller'); ?>", // Endpoint ke
controller
            type: "GET",
        },
        columns: [
            // Kolom disesuaikan dengan response yang
            // dikirimkan dari controller
            { data: "Nilai" },
            // Jika ada kolom aksi pada tabel
            {
                data: 'id',
                render: function (data, type, row,
meta) {
                    return `
                    <a href="<?="
base_url('route_ke_controller'); ?>/${data}"
class="btn btn-primary btn-sm">Edit</a>
                    <a href="<?="
base_url('route_ke_controller'); ?>/${data}"
onclick="return confirm('Apakah Anda yakin ingin
menghapus ${row.column}')" class="btn btn-danger btn-
sm">Hapus</a>
                    `;
                }
            },
        ],
        // Pengurutan data
        order: [
            [0, 'asc']
        ],
    });
});
```

Gambar 2. Contoh implementasi *Server-side DataTables*

E. Modifikasi Controllers agar mendukung Server-side DataTables

Selanjutnya, penulis melakukan modifikasi pada bagian *backend* sistem, terutama pada bagian *Controllers*. Arsitektur perangkat lunak yang dipilih dalam pengembangan sistem ini adalah *Model View Controller* [29]. Pengaturan yang tepat di sisi *Controllers* akan memungkinkan sistem untuk berfungsi dengan baik. Yang sebelumnya sintaks menggunakan pendekatan *client-side* kita ubah supaya mendukung *server-side* dengan menambahkan seluruh sintaks berikut pada bagian *controllers* di sisi *backend* sistem pada *framework CodeIgniter4*.

```

public function getData(){
    $request = service('request');
    $model = new Model();

    $columns = [
        'kolom_di_basis_data',
    ];

    $select = implode(' ', $columns);

    $builder = $model->select($select);
    $totalRecords = $model->countAllResults();

    $draw = $request->getVar('draw');
    $start = $request->getVar('start');
    $length = $request->getVar('length');
    $order = $request->getVar('order');
    $search = $request->getVar('search')

    ['value'];

    $recordsFiltered = $builder->countAllResults();

    foreach ($order as $orderColumn) {
        $builder->orderBy($columns[$orderColumn['column']],
        $orderColumn['dir']);
    }

    if (!empty($search)) {
        $builder->groupStart();
        $builder->orLike('kolom_di_basis_data',
        $search)->orLike('kolom_di_basis_data', $search);
        $builder->groupEnd();
    }

    $builder->limit($length, $start);
    $data = $builder->get()->getResult();

    $output = [
        'draw' => $draw,
        'recordsTotal' => $totalRecords,
        'recordsFiltered' => $recordsFiltered,
        'data' => $data,
    ];

    return $this->response->setJSON($output);
}

```

Gambar 3. Contoh implementasi *Server-side DataTables* pada *Controllers*

Modifikasi dilakukan pada sumber kode yang sudah ada pada sistem sebelumnya supaya dapat dihubungkan dengan sisi klien/*client side*. Modifikasi yang dilakukan dapat terlihat pada contoh kode pada Gambar 3. Disitu terdapat beberapa perbedaan antara lain, *Controllers* yang telah dimodifikasi memiliki logika untuk melakukan fitur pencarian data dari sisi klien, data yang dikirimkan, terdapat beberapa parameter tambahan, dan dapat mengatur kolom apa saja yang ingin diambil datanya dari basis data. Dan proses pengolahan data dilakukan oleh *backend* sebelum dikirimkan ke sisi klien yang menyebabkan sistem dapat berjalan lebih cepat. *Backend* sendiri

berada di bagian *server*, sehingga pemrosesan data dibebankan pada *server*.

F. Implementasi Index pada basis data MySQL

Dalam tahap ini, penulis melakukan penerapan *Index* pada basis data *MySQL*. *MySQL* adalah sistem manajemen basis data relasional *open source* yang *real-time* dengan langkah-langkah keamanan untuk mengakses, memanipulasi dan menyimpan secara lokal pada basis data [23]. Ini melibatkan eksekusi beberapa kueri yang bertujuan untuk memodifikasi basis data. *Index* yang diterapkan akan memungkinkan sistem untuk mencari data dengan lebih efisien.

```

ALTER TABLE `nama_tabel`
ADD KEY `index_key_name` (`nama_kolom_pada_tabel`)

```

Gambar 4. Contoh kueri untuk memodifikasi tabel pada basis data

Pada Gambar 4 memperlihatkan contoh kueri yang digunakan untuk melakukan modifikasi pada tabel di basis data. Pemilihan kolom pada tabel basis data yang dimodifikasi dengan *index* adalah kolom dengan data yang cenderung unik atau sering digunakan dalam pencarian, jika kasusnya adalah data mahasiswa maka nama, NIM, atau *email*.

G. Pengujian performa sistem dengan Apache JMeter dan efisiensi kueri dengan dbForge setelah implementasi

Tahapan akhir dari penelitian ini adalah uji performa sistem dan efisiensi basis data dalam pemrosesan data ke sisi pengguna setelah diimplementasikannya *Index* pada basis data *MySQL* dan *Server-side DataTables*. Pengujian perangkat lunak diperlukan untuk menunjukkan cacat dan kesalahan yang dibuat selama fase pengembangan [30]. Pengujian tahap akhir ini akan memberikan gambaran tentang sejauh mana perbaikan dan perubahan yang diterapkan telah mempengaruhi performa dan efisiensi sistem.

IV. HASIL DAN PEMBAHASAN

Pengujian performa dan efisiensi tahap awal melibatkan simulasi sebanyak 100 pengguna yang mengakses salah satu laman pada sistem secara bersamaan dan proses simulasi diulang sebanyak 5 kali. Lalu untuk pengujian efisiensi kueri dengan memasukkan kueri ke *dbForge Studio*. Tujuannya adalah untuk mengidentifikasi potensi masalah kinerja dan penundaan dalam situasi beban tinggi yang hasilnya disajikan pada Tabel 1 hingga Tabel 6.

Tabel 1. Hasil uji performa jisistem sebelum implementasi

HTTP Request & Query	Waktu Respons (milidetik)		
	Minimum	Maksimum	Rata-rata
HTTP Request			
Fetch	944	34441	29820.17
Search	1121	35862	30512.44
Page load	1060	69572	61560.63
Query			
SELECT	55	16418	2578.61
WHERE LIKE	13	11053	1564.77

Tabel 2. Hasil uji efisiensi sistem sebelum implementasi

Profile	Nilai
Rows checked	1015 rows
InnoDB buffer pool read request (Global)	1078 reads
Query cost	107,49 %
Summed query duration	6,348 ms

Kemudian pada pengujian performa dan efisiensi tahap akhir penulis menggunakan spesifikasi yang sama dengan tahap awal namun bedanya terletak pada sintaks sistem dan basis data yang sudah dimodifikasi dengan hasil berikut :

Tabel 3. Hasil uji performa sistem setelah implementasi

HTTP Request & Query	Waktu Respons (milidetik)		
	Minimum	Maksimum	Rata-rata
HTTP Request			
Fetch	1092	33128	9327.52
Search	799	34393	29865.18
Page load	767	40557	35401.62
Query			
SELECT	59	13008	2750.74
WHERE LIKE	1	7952	1210.41

Tabel 4. Hasil uji efisiensi sistem setelah implementasi

Profile	Nilai
Rows checked	3 rows
InnoDB buffer pool read request (Global)	18 reads
Query cost	3,82 %
Summed query duration	1,158 ms

Dari kedua hasil pengujian di atas, dapat disajikan dalam bentuk perbandingan antara hasil uji performa

sistem serta efisiensi kueri sebelum dan sesudah implementasi. Berikut adalah hasil perbandingannya.

Tabel 5. Perbandingan hasil uji performa sistem sebelum dan setelah implementasi

HTTP Request & Query	Perbandingan rata-rata (milidetik)		
	Sebelum	Sesudah	Perbandingan (%)
HTTP Request			
Fetch	29820.17	9327.52	68,72 ▲
Search	30512.44	29865.18	2,12 ▲
Page load	61560.63	35401.62	42,49 ▲
Query			
SELECT	2578.61	2750.74	-6,68 ▼
WHERE LIKE	1564.77	1210.41	22,65 ▲
Rata-rata perbandingan performa		25,86 ▲	

Hasil uji performa sistem menunjukkan peningkatan yang signifikan dalam sebagian besar aspek performa. *HTTP Request Fetch*, yang mengukur waktu untuk tabel mengambil data dari basis data, mengalami peningkatan sebesar 68,72%, mengindikasikan pemrosesan data yang lebih cepat setelah penerapan *Index MySQL* dan *Server-side Data Tables*. Demikian juga, *Page load*, yang mencatat waktu yang dibutuhkan untuk memuat laman, mengalami peningkatan sebesar 42,49%, menciptakan pengalaman pengguna yang lebih responsif.

Tabel 6. Perbandingan hasil uji efisiensi kueri sebelum dan setelah implementasi

Profile	Perbandingan rata-rata (milidetik)		
	Sebelum	Sesudah	Perbandingan (%)
Rows checked	1015 rows	3 rows	99,70 ▲
InnoDB buffer pool read request	1078 reads	18 reads	98,33 ▲
Query cost	107,49 %	3,82 %	96,45 ▲
Summed query duration	6,348 ms	1,158 ms	81,76 ▲

Rata-rata perbandingan efisiensi	94,06 ▲
----------------------------------	---------

Peningkatan yang mencolok juga terlihat dalam uji efisiensi kueri basis data. *Rows checked* mengalami peningkatan 99,70%, menunjukkan bahwa proses verifikasi data menjadi lebih efisien. Hasil lainnya seperti *Innodb buffer pool read request* yang menunjukkan jumlah permintaan baca logis dari memori yang telah dilakukan *InnoDB* dan *Query cost* menunjukkan peningkatan 98,33% dan 96,45% masing-masing, yang mencerminkan efisiensi tinggi dalam mengambil data dari basis data.

Hasil yang positif dari uji performa sistem dan efisiensi kueri basis data menggambarkan keberhasilan penerapan *Index MySQL* dan *Server-side DataTables* dalam meningkatkan performa dan efisiensi sistem MBKM berbasis *CodeIgniter4*, meskipun terdapat penurunan performa sebesar -6,68% untuk *Query SELECT* karena *Index* tidak perlu digunakan pada kueri yang sederhana. Penurunan yang signifikan dalam waktu yang dibutuhkan sistem untuk mengambil data dari basis data (*HTTP Request Fetch*) dan waktu muat halaman (*Page load*) mencerminkan optimasi yang kuat dalam pemrosesan data pengguna.

Lebih lanjut, efisiensi kueri basis data yang tinggi, seperti terlihat dalam peningkatan *Rows checked*, *Innodb buffer pool read request*, dan *Query cost*, menunjukkan bahwa penggunaan *Index* pada basis data *MySQL* telah menghasilkan hasil yang signifikan dalam mengurangi beban sistem. Efisiensi ini memungkinkan sistem untuk beroperasi lebih cepat dan dengan *overhead* yang lebih rendah, menghasilkan waktu respon yang lebih baik dan penggunaan sumber daya yang lebih efisien.

V. PENUTUP

A. Kesimpulan

Secara keseluruhan, hasil penelitian ini menyimpulkan bahwa penerapan *Index MySQL* dan *Server-side DataTables* secara signifikan meningkatkan performa sistem MBKM berbasis *CodeIgniter4*. Peningkatan yang substansial terlihat dalam waktu pemrosesan data pengguna dan pengambilan data dari basis data. Hasil kuantitatif menunjukkan adanya peningkatan rata-rata sebesar 25,86% dalam performa sistem, serta peningkatan efisiensi kueri rata-rata sebesar 94,06%.

Hasil ini secara langsung berkontribusi pada pengalaman pengguna yang lebih baik dan efisiensi

sistem. Efisiensi kueri basis data yang tinggi, dengan peningkatan sebesar 99,70% dalam *Rows checked*, 98,33% dalam *Innodb buffer pool read request*, dan 96,45% dalam *Query cost*, menunjukkan bahwa penggunaan *Index* pada basis data *MySQL* memberikan manfaat yang signifikan dalam mengurangi beban sistem. Dengan peningkatan performa dan efisiensi ini, sistem MBKM FIK UDINUS telah mengoptimalkan operasionalnya, memberikan dukungan bagi perkembangan pendidikan yang lebih baik.

B. Saran

Berdasarkan penelitian yang telah dilakukan terdapat beberapa saran dari hasil penelitian ini. Masih perlu dilakukannya pengujian lebih lanjut pada jenis-jenis kueri yang lain dan uji coba lanjutan untuk mengetahui sejauh mana *server* dapat menangani permintaan pengguna dalam waktu tertentu.

Kemudian mengingat penurunan performa sebesar -6,68% pada *Query SELECT* yang menggunakan *Index*, disarankan untuk memeriksa kembali penggunaan *Index* pada kueri yang kompleks. Pengoptimalan indeks dapat dilakukan dengan lebih selektif, sehingga hanya kueri-kueri yang membutuhkan indeks yang digunakan. Hal ini dapat membantu menjaga performa sistem dalam situasi kueri sederhana.

DAFTAR PUSTAKA

- [1] M. Maddukelleng, J. Jihan, H. Gunawan, H. Murcahyanto, dan W. Pasaribu, "Hybrid Learning Innovation: Challenges for Developing Teachers Skills in Indonesia," *Al Qalam: Jurnal Ilmiah Keagamaan dan Kemasyarakatan*, vol. 17, no. 2, hlm. 100, Mar 2023, doi: 10.35931/aq.v17i2.1959.
- [2] W. Wahyudi, J. Jumadi, dan D. A. Nurhidayah, "Implementasi Instrumen Kemandirian Belajar Mahasiswa untuk Meningkatkan Prestasi Belajar di Masa Pandemi COVID-19," *Jurnal Cendekia : Jurnal Pendidikan Matematika*, vol. 6, no. 1, hlm. 925–932, Feb 2022, doi: 10.31004/cendekia.v6i1.1299.
- [3] K. D. P. Meke, R. B. Astro, dan M. H. Daud, "Dampak Kebijakan Merdeka Belajar Kampus Merdeka (MBKM) pada Perguruan Tinggi Swasta di Indonesia," *EDUKATIF : JURNAL ILMU PENDIDIKAN*, vol. 4, no. 1,

- hlm. 675–685, Des 2021, doi: 10.31004/edukatif.v4i1.1940.
- [4] A. Renz dan R. Hilbig, “Prerequisites for artificial intelligence in further education: identification of drivers, barriers, and business models of educational technology companies,” *International Journal of Educational Technology in Higher Education*, vol. 17, no. 1, hlm. 14, Des 2020, doi: 10.1186/s41239-020-00193-3.
- [5] E. S. Karpov dan E. G. Karpova, “Digitalization of Higher Education in the Context of Globalization,” *KnE Social Sciences*, hlm. 9–19, Feb 2022, doi: 10.18502/kss.v7i3.10270.
- [6] M. A. Mohamed Hashim, I. Tlemsani, dan R. Matthews, “Higher education strategy in digital transformation,” *Educ Inf Technol (Dordr)*, vol. 27, no. 3, hlm. 3171–3195, Apr 2022, doi: 10.1007/s10639-021-10739-1.
- [7] M. Mohzana, M. Fahrurrozi, dan H. Murcahyanto, “Pengaruh Penggunaan E-Learning pada Mahasiswa,” *Journal of Education and Instruction (JOEAI)*, vol. 4, no. 1, hlm. 233–242, Jun 2021, doi: 10.31539/joeai.v4i1.2087.
- [8] A. Syaputra dan E. Hasanah, “Learning Strategies In The Digital Era,” *International Journal of Educational Management and Innovation*, vol. 3, no. 1, hlm. 74–83, Jan 2022, doi: 10.12928/ijemi.v3i1.5507.
- [9] E. Nurelasari, “Perancangan Sistem Informasi Akademik Pada Sekolah Menengah Pertama Berbasis Web,” *Komputika : Jurnal Sistem Komputer*, vol. 9, no. 1, hlm. 67–73, Apr 2020, doi: 10.34010/komputika.v9i1.2243.
- [10] E. Simatupang dan I. Yuhertiana, “Merdeka Belajar Kampus Merdeka terhadap Perubahan Paradigma Pembelajaran pada Pendidikan Tinggi: Sebuah Tinjauan Literatur,” *Jurnal Bisnis, Manajemen, dan Ekonomi*, vol. 2, no. 2, hlm. 30–38, Apr 2021, doi: 10.47747/jbme.v2i2.230.
- [11] A. Muntohar, “Edumatic: Jurnal Pendidikan Informatika Sistem Informasi Data Klien Berbasis Java Pada Kantor Notaris dan PPAT,” vol. 4, no. 2, 2020, doi: 10.29408/edumatic.v4i2.2515.
- [12] A. Deska, I. Akbar, dan P. Studi Magister Ilmu Komputer, “Tuning Database Pada Sistem Penerimaan Mahasiswa Baru Menggunakan Optimasi Query dan Indexing Tuning Database on New Student Admission System Using Query and Indexing Optimization.”
- [13] R. P. Aryanto, A. Nilogiri, dan A. E. Wardoyo, “Optimasi Pengurutan Data Bilangan dengan Menggabungkan Algoritma Selection Sort Hybrid dan Bucket Sort,” *Edumatic: Jurnal Pendidikan Informatika*, vol. 7, no. 1, hlm. 39–48, Jun 2023, doi: 10.29408/edumatic.v7i1.12358.
- [14] A. Cravero, A. Bustamante, M. Negrier, dan P. Galeas, “Agricultural Big Data Architectures in the Context of Climate Change: A Systematic Literature Review,” *Sustainability*, vol. 14, no. 13, hlm. 7855, Jun 2022, doi: 10.3390/su14137855.
- [15] C. Purnama Yanti, N. K. N. N. Pande, dan I. M. S. Sandhiyasa, “IMPLEMENTASI SERVER SIDE UNTUK MANAJEMEN AUDIT MUTU INTERNAL STMIK STIKOM INDONESIA,” *Jusikom : Jurnal Sistem Komputer Musirawas*, vol. 5, no. 1, hlm. 1–12, Mei 2020, doi: 10.32767/jusikom.v5i1.753.
- [16] R. Somya dan B. Beny, “Pemanfaatan Plugin DataTables untuk Sistem Informasi di Unit Indostamping PT Pura Barutama,” *Jurnal Informatika*, vol. 6, no. 1, hlm. 84–90, Apr 2019, doi: 10.31311/ji.v6i1.5286.
- [17] D. I. Permatasari, “Pengujian Aplikasi menggunakan metode Load Testing dengan Apache JMeter pada Sistem Informasi Pertanian,” *Jurnal Sistem dan Teknologi Informasi (JUSTIN)*, vol. 8, no. 1, hlm. 135, Jan 2020, doi: 10.26418/justin.v8i1.34452.
- [18] R. Samli dan Z. Orman, “İleri Mühendislik Çalışmaları ve Teknolojileri Dergisi.”
- [19] H. I. Robbani dan R. P. Wibowo, “Rancang Bangun Aplikasi Visualisasi Database SQL Server dengan Dynamic Management View Berbasis Graph Neo4j untuk Memetakan Relasi Implisit pada Database,” *Jurnal Teknik ITS*, vol. 7, no. 1, Apr 2018, doi: 10.12962/j23373539.v7i1.29299.
- [20] D. I. Permatasari, “Pengujian Aplikasi menggunakan metode Load Testing dengan Apache JMeter pada Sistem Informasi

- Pertanian,” *Jurnal Sistem dan Teknologi Informasi (JUSTIN)*, vol. 8, no. 1, hlm. 135, Jan 2020, doi: 10.26418/justin.v8i1.34452.
- [21] I. Jauharul Makhnun, “LITERATURE REVIEW PENERAPAN DATA VISUALIZATION PADA PERUSAHAAN,” *Jurnal Impresi Indonesia (JII)*, vol. 2, no. 2, 2023, doi: 10.58344/jii.v2i2.2156.
- [22] L. Afuan, N. Nofiyati, dan N. Umayah, “Rancang Bangun Sistem Informasi Bank Sampah di Desa Paguyangan,” *Edumatic: Jurnal Pendidikan Informatika*, vol. 5, no. 1, hlm. 21–30, Jun 2021, doi: 10.29408/edumatic.v5i1.3171.
- [23] R. Bhagat, “An MQTT based IoT-RFID Attendance System using NodeMCU Firmware: A Review,” *International Research Journal of Engineering and Technology*, 2020, [Daring]. Tersedia pada: www.irjet.net
- [24] N. Azizah, H. G. R. Kumbara, P. Krishnacahya, dan L. M. Tjahjono, “Sistem Pendukung Metode Pembelajaran Self Paced Learning bagi Mahasiswa di dalam Kelas berbasis Web,” *Edumatic: Jurnal Pendidikan Informatika*, vol. 7, no. 1, hlm. 20–29, Jun 2023, doi: 10.29408/edumatic.v7i1.8317.
- [25] F. O. Saputra, K. Ingsih, E. Kartikadarma, W. Isthika, L. Johary, dan M. B. Sakti, “Pengelolaan Sampah Rumah Tangga Berbasis Aplikasi Pada Seluruh Bank Sampah Di Kecamatan Semarang Barat,” *Jurnal Pengabdian Multidisiplin*, vol. 3, no. 2, Okt 2023, doi: 10.51214/japamul.v3i2.632.
- [26] W. M. Sari, A. Amran, dan H. O. Lingga Wijaya, “PENERAPAN E-COMMERCE MENGGUNAKAN METODE EXTREME PROGRAMMING PADA UMKM KABUPATEN MURATARA,” *Jusikom : Jurnal Sistem Komputer Musirawas*, vol. 5, no. 2, hlm. 136–144, Nov 2020, doi: 10.32767/jusikom.v5i2.1095.
- [27] A. Dwi Sapto dan H. Pandu Rumakso, “Uji Coba Performa Bentuk Airfoil Menggunakan Software Qblade Terhadap Turbin Angin Tipe Sumbu Horizontal,” *Jurnal Teknik Mesin*, vol. 10, no. 1, hlm. 1, Mar 2021, doi: 10.22441/jtm.v10i1.10212.
- [28] R. Abbas, Z. Sultan, dan S. N. Bhatti, “Comparative analysis of automated load testing tools: Apache JMeter, Microsoft Visual Studio (TFS), LoadRunner, Siege,” dalam *2017 International Conference on Communication Technologies (ComTech)*, IEEE, Apr 2017, hlm. 39–44. doi: 10.1109/COMTECH.2017.8065747.
- [29] I. Y. Lim dkk., “Sistem Informasi Feeder Data Kependudukan berbasis Mobile,” *Edumatic: Jurnal Pendidikan Informatika*, vol. 7, no. 1, hlm. 30–38, Jun 2023, doi: 10.29408/edumatic.v7i1.12306.
- [30] M. Huda, M. A. Yaqin, R. F. Kurniawan, dan M. W. Fitra Choiri, “Penerapan Systematic Literature Review untuk Survei Strategi Pengujian Perangkat Lunak,” *ILKOMNIKA: Journal of Computer Science and Applied Informatics*, vol. 4, no. 1, hlm. 116–133, Apr 2022, doi: 10.28926/ilkomnika.v4i1.255.