



IMPLEMENTASI ALGORITMA BUBBLE SORT DALAM PENGURUTAN DATA TRANSAKSI PENJUALAN MOTOR SECOND (STUDI KASUS DUTA MOTOR BINJAI)

Liza Audia Zahra¹, Dinda Aulia Putri Br Sembiring², Ninda Clarica³, Safrizal⁴

^{1,2,3} Program Studi Sistem Informasi, Fakultas Ilmu Komputer, STMIK Kaputama Binjai

⁴ Program Studi Sistem Informasi, Fakultas Ekonomi Sains dan Teknologi, Universitas Muhammadiyah
Asahan

E-mail: lizaaudia12@gmail.com¹, dindaauliaputribrsbr@gmail.com², nindaclarica14@icloud.com³,
rizalsyl75@gmail.com⁴

ABSTRACT

Data sorting is a fundamental process that plays a crucial role in sales information systems, especially for used motorcycle businesses that analyze transactions on a daily basis. This study discusses the implementation of the Bubble Sort algorithm on a sales transaction dataset of a used motorcycle business, aiming to understand how the algorithm works in sorting data based on the gross income column and to analyze its time efficiency and the number of comparisons performed. The dataset used was obtained from used motorcycle sales transactions containing more than 1,000 records, with the implementation carried out using HTML, CSS, and PHP, and the first six data entries taken as research samples. The results of the study show that the Bubble Sort algorithm successfully sorted the transaction data in ascending order with a total of 15 comparison steps, consistent with the time complexity of $O(n^2)$. Although the algorithm is relatively simple, its implementation proves effective as both a learning method and a foundational approach for data sorting systems in used motorcycle sales businesses.

Keywords: Algorithm, Bubble Sort, Data Sorting, Sales System, Used Motorcycle Business.

ABSTRAK

Pengurutan data merupakan proses dasar yang sangat penting dalam sistem informasi penjualan, terutama bagi usaha penjualan motor second yang melakukan analisis transaksi setiap harinya. Penelitian ini membahas implementasi algoritma Bubble Sort pada dataset transaksi penjualan usaha motor second dengan tujuan untuk mengetahui bagaimana algoritma tersebut bekerja dalam mengurutkan data berdasarkan kolom gross income serta menganalisis efisiensi waktu dan jumlah perbandingan yang terjadi. Dataset yang digunakan diambil dari data transaksi penjualan motor bekas yang berisi lebih dari 1000 transaksi, dengan implementasi menggunakan bahasa HTML, CSS, dan PHP serta pengambilan 6 data pertama sebagai sampel penelitian. Hasil penelitian menunjukkan bahwa algoritma Bubble Sort berhasil

Article History

Received: Oktober 2025

Reviewed: Oktober 2025

Published: Oktober 2025

Plagiarism Checker No
235

Prefix DOI :

[10.8734/Kohesi.v1i2.365](https://doi.org/10.8734/Kohesi.v1i2.365)

Copyright : Author

Publish by : Kohesi



This work is licensed
under a [Creative Commons Attribution-NonCommercial 4.0 International License](https://creativecommons.org/licenses/by-nc/4.0/)



mengurutkan data transaksi secara ascending dengan total 15 langkah perbandingan, sesuai dengan kompleksitas waktu $O(n^2)$. Meskipun algoritma ini tergolong sederhana, penerapannya terbukti efektif sebagai metode pembelajaran sekaligus dasar dalam sistem pengurutan data pada usaha penjualan motor second.

Kata kunci: *Algoritma, Bubble Sort, Pengurutan Data, Sistem Penjualan, Usaha Motor Second.*

1. PENDAHULUAN

Kemajuan teknologi informasi telah membawa perubahan signifikan dalam cara pelaku usaha mengelola data dan mengambil keputusan bisnis. Digitalisasi sistem informasi penjualan menjadi faktor utama dalam meningkatkan efisiensi dan efektivitas operasional, termasuk pada usaha penjualan motor second. Penerapan sistem digital memungkinkan proses analisis dan pelaporan dilakukan secara otomatis, akurat, dan cepat, sehingga membantu pengusaha dalam mengambil keputusan strategis secara tepat waktu (Maulidina & Nafiati, 2023).

Salah satu aspek penting dalam sistem pengelolaan data penjualan adalah proses pengurutan (sorting), yang berfungsi menampilkan laporan transaksi secara terstruktur berdasarkan nilai, tanggal, atau kategori tertentu. Tanpa sistem pengurutan yang baik, penyusunan laporan penjualan dapat menjadi lambat dan berisiko menimbulkan kesalahan (Barokah et al., 2025). Dalam praktiknya, banyak usaha kecil masih mengelola data secara manual, sehingga dibutuhkan metode pengurutan yang sederhana, efisien, dan mudah diimplementasikan (Arista & Nugroho, 2023).

Algoritma Bubble Sort menjadi salah satu solusi dasar yang dapat digunakan untuk mengatasi masalah pengurutan data tersebut. Meskipun memiliki kompleksitas waktu $O(n^2)$, algoritma ini mudah dipahami dan sesuai untuk dataset kecil hingga menengah (Nirwan et al., 2024). Beberapa penelitian, seperti Mahrozi dan Faisal (2023) serta Panggabean et al. (2023), menunjukkan efektivitas Bubble Sort dalam pengurutan data berbasis web dengan hasil yang stabil dan efisien. Pada penelitian ini, algoritma Bubble Sort diterapkan pada dataset transaksi penjualan motor second untuk mengurutkan data berdasarkan nilai gross income secara ascending menggunakan bahasa pemrograman HTML, CSS, dan PHP (Safrianti & Ezwarsyah, 2024).

2. METODE PENELITIAN

Penelitian ini digolongkan sebagai penelitian terapan (applied research). Tujuan utamanya adalah mengembangkan dan mengimplementasikan solusi praktis yang dapat



mengatasi tantangan spesifik dalam pengelolaan dan pengurutan data transaksi penjualan motor bekas di Duta Motor Binjai.

Pendekatan yang diadopsi dalam studi ini adalah rekayasa perangkat lunak (software engineering). Kerangka kerja ini digunakan untuk memastikan pengembangan sistem berjalan secara terstruktur, yang meliputi serangkaian fase krusial: analisis kebutuhan data Duta Motor Binjai, perancangan modul pengurutan, implementasi kode Algoritma Bubble Sort, dan pengujian sistem secara menyeluruh. Dengan demikian, penelitian ini secara eksplisit berfokus pada penerapan teknis Algoritma Bubble Sort untuk fungsi pengurutan data transaksi.

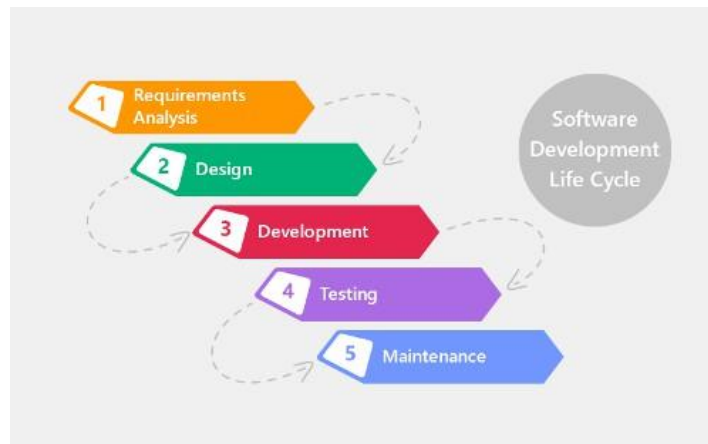
Untuk mengevaluasi keberhasilan implementasi, digunakan pula pendekatan kuantitatif deskriptif. Pendekatan ini melibatkan pengukuran performa algoritma secara empiris, khususnya dengan mencatat efisiensi waktu eksekusi (kecepatan komputasi) dan memverifikasi tingkat akurasi (keberhasilan pengurutan) Algoritma Bubble Sort terhadap dataset transaksi nyata. Selanjutnya, hasil pengukuran kuantitatif tersebut dideskripsikan secara naratif untuk memberikan gambaran yang komprehensif mengenai peningkatan atau perubahan kinerja sistem setelah adopsi metode Bubble Sort.

2.1 Kerangka Kerja Pengembangan Sistem

Untuk kerangka kerja pengembangan sistem, penelitian ini mengadopsi Model *Waterfall* (Air Terjun). Pemilihan model ini didasarkan pada sifat proyek yang memiliki persyaratan dan definisi solusi yang jelas dan stabil sejak awal, yaitu implementasi Algoritma Bubble Sort. Model *Waterfall* menawarkan pendekatan yang sekuensial dan sistematis, di mana setiap tahapan harus diselesaikan secara tuntas sebelum berlanjut ke tahap berikutnya. Fase-fase utama yang akan diterapkan meliputi Analisis Kebutuhan (mengidentifikasi data dan kriteria pengurutan Duta Motor Binjai), Perancangan Sistem (merancang struktur *database* dan *interface*), Implementasi (mengembangkan kode Bubble Sort), dan Pengujian Sistem (memvalidasi fungsionalitas dan mengukur efisiensi waktu eksekusi). Pendekatan linier ini memastikan proses pengembangan berjalan terstruktur dan terdokumentasi dengan baik, sesuai kebutuhan penelitian akademis.

2.2 Model Pengembangan Sistem

Dalam penelitian ini, model yang digunakan adalah Model Waterfall. Model Waterfall, yang diperkenalkan oleh Winston W. Royce (1970), adalah metodologi pengembangan perangkat lunak yang sistematis dan berurutan, di mana setiap fase harus diselesaikan sepenuhnya sebelum beralih ke fase berikutnya.

Gambar 1. Siklus Waterfall

1. Requirements Analysis (Analisis Kebutuhan)

Tahap ini diawali dengan studi lapangan melalui observasi langsung terhadap alur kerja dan pengelolaan data transaksi penjualan motor bekas di Duta Motor Binjai. Hasil analisis menunjukkan bahwa organisasi kesulitan dalam mengelola data transaksi historis, terutama saat diperlukan pengurutan data (misalnya, mencari 10 motor dengan harga jual tertinggi, atau mengurutkan berdasarkan tanggal penjualan terbaru). Kondisi ini menyebabkan inefisiensi waktu saat pembuatan laporan atau pengambilan keputusan strategis. Oleh karena itu, penelitian ini bertujuan untuk mengembangkan modul pengurutan data transaksi yang dapat berjalan secara otomatis, cepat, dan terstruktur.

2. System Design (Perancangan Sistem)

Fase ini melibatkan perencanaan arsitektur logis dan fisik sistem. Perancangan difokuskan pada tiga elemen utama: struktur data, basis data, dan algoritma inti. Basis data dirancang menggunakan MySQL yang dikelola melalui lingkungan XAMPP. Antarmuka sistem dikembangkan menggunakan bahasa PHP untuk demonstrasi berbasis web. Fitur krusial yang dirancang adalah modul pengurutan data transaksi yang mengintegrasikan Algoritma Bubble Sort sebagai mekanisme pengurutan utamanya.

3. Development / Implementation (Pengembangan / Implementasi)

Tahap ini adalah realisasi dari cetak biru desain menjadi perangkat lunak berbasis web yang fungsional. Sistem dikembangkan untuk menerima dataset transaksi penjualan dari Duta Motor Binjai, kemudian pengguna dapat memilih kriteria pengurutan (misalnya, harga, tanggal). Proses pengurutan menggunakan prinsip Bubble Sort, di mana data akan dibandingkan dan ditukar secara berpasangan berulang kali hingga seluruh dataset terurut secara sempurna.

4. Testing (Pengujian)

Tujuan utama pengujian adalah untuk memvalidasi fungsionalitas dan mengukur kinerja Algoritma Bubble Sort pada dataset nyata. Pengujian dilakukan menggunakan data sampel dari total keseluruhan transaksi motor bekas yang tersedia.



Pengujian meliputi:

Pengujian Fungsionalitas: Memastikan data terurut secara akurat 100% sesuai kriteria yang dipilih (menaik atau menurun).

Pengujian Kinerja: Mengukur waktu komputasi (eksekusi) yang dibutuhkan Bubble Sort untuk mengurutkan dataset dengan volume data yang berbeda-beda.

Hasil pengujian ini kemudian dianalisis untuk menyimpulkan efisiensi waktu dan kinerja algoritma Bubble Sort dalam konteks sistem berskala menengah Duta Motor Binjai.

5. *Maintenance* (Pemeliharaan)

Setelah sistem berhasil melalui tahap pengujian dan diimplementasikan di Duta Motor Binjai, fase pemeliharaan dilakukan. Aktivitas ini mencakup pemantauan stabilitas operasional sistem, penanganan bug yang mungkin muncul pasca-implementasi, serta penyesuaian fungsionalitas jika terjadi perubahan pada format data transaksi atau kebutuhan bisnis koperasi di masa depan.

2.3 Data dan Sumber Data

Data yang digunakan pada penelitian ini diperoleh dari dua sumber data, yaitu dari data primer dan data sekunder

Data Primer: Diperoleh langsung dari pihak Duta Motor Binjai melalui wawancara dan observasi. Ini mencakup proses bisnis penjualan, kebutuhan pengurutan data, dan format data transaksi saat ini.

Data Sekunder: Berupa data historis transaksi penjualan motor bekas dari Duta Motor Binjai (misalnya, selama 3-6 bulan terakhir), yang meliputi:

Table 1. Atribut Data

Model	Tahun	Harga	Transmisi	Odometer	Jenis
-------	-------	-------	-----------	----------	-------

Untuk keperluan pengujian algoritma, pada penelitian ini digunakan 6 data penjualan pertama sebagai *sampel dataset*, data seperti yang ditunjukkan pada Tabel 2.

Table 2. Data Motor

Model	Tahun	Harga	Transmisi	Odometer	Jenis
Supra X 125	2017	7999	Manual	17307	Bebek
Vario 125	2016	14499	Automatic	25233	Skuter
Vario 125	2016	11399	Automatic	37877	Skuter
Revo	2016	6499	Manual	23789	Bebek
PCX	2015	10199	Automatic	33177	Skuter
Revo	2015	6149	Manual	41650	Bebek

Dataset yang didapatkan dari data penjualan Sepeda Motor Second akan digunakan untuk menguji tingkat efisiensi dan akurasi dari algoritma *Bubble Sort* dalam pengurutan data berdasarkan **Harga Sepeda Motor**



2.4 Cara Kerja Algoritma *Bubble Sort*

Penerapan algoritma Bubble Sort dalam penelitian ini digunakan untuk mengurutkan data penjualan motor bekas berdasarkan kriteria tertentu (seperti harga atau tahun) dengan dua arah pengurutan, yaitu ascending (menaik) dan descending (menurun).

Adapun langkah-langkah proses kerja algoritma adalah sebagai berikut:

1. Inisialisasi Data Motor

Sistem menyediakan kumpulan data motor dalam bentuk array yang berisi informasi seperti nama motor, tahun, dan harga.

2. Menentukan Pasangan Data untuk Dibandingkan

Algoritma memulai proses dari elemen pertama dan membandingkannya dengan elemen di sebelahnya.

3. Perbandingan dan Pertukaran Data (Swap)

Jika data pertama lebih besar dari data kedua (untuk ascending), maka posisi keduanya ditukar. Untuk descending, pertukaran dilakukan jika data pertama lebih kecil dari data kedua.

4. Iterasi Proses Pengurutan

Setelah satu putaran selesai, elemen terkecil (ascending) atau terbesar (descending) akan berada di posisi akhir array. Proses dilanjutkan untuk sisa data hingga semua elemen terurut.

5. Pemeriksaan Kondisi Berhenti

Proses berhenti ketika dalam satu kali iterasi tidak terjadi pertukaran data, yang berarti seluruh elemen telah terurut sempurna.

6. Menampilkan Hasil Pengurutan

Setelah semua langkah selesai, sistem menampilkan hasil pengurutan sesuai kriteria dan arah urutan yang dipilih oleh pengguna.

2.5 Rumus Proses *Bubble Sort*

Secara matematis, proses pengurutan data dengan algoritma Bubble Sort dapat dinyatakan dalam bentuk rumus berikut:

$$\begin{aligned} & \text{for } i = 0 \text{ to } n - 1: \\ & \quad \text{for } j = 0 \text{ to } n - i - 2: \\ & \quad \quad \text{if } X[j] > X[j + 1]: \text{ then swap } X[j] \text{ and } X[j + 1] \end{aligned}$$

Keterangan :

$X[j]$ = elemen ke- j dari data



n = jumlah total data

swap = proses pertukaran posisi dua elemen jika urutan tidak sesuai

Rumus tersebut menunjukkan bahwa pada setiap iterasi, elemen dengan nilai terbesar akan berpindah ke posisi akhir array (untuk ascending), sedangkan untuk descending, elemen terkecil yang akan berpindah ke posisi akhir.

2.5 Contoh Data dan Proses Pengurutan

Diberikan contoh data awal sebagai berikut:

[7999.0,14499.0,11399.0,6499.0,10199.0,6149.0]

Langkah-langkah proses pengurutan secara ascending:

1. Bandingkan $X[1]$ dengan $X[2]$ dan susun sehingga $X[1] < X[2]$
2. Bandingkan $X[2]$ dengan $X[3]$ dan susun sehingga $X[2] < X[3]$.
3. Bandingkan $X[n - 1]$ dengan $X[n]$ dan susun sehingga $X[n - 1] < X[n]$
4. Setelah satu iterasi, elemen terbesar berada di posisi akhir.
5. Ulangi langkah di atas hingga seluruh data terurut.

Berikut Data Awal

Nama Motor	Tahun	Harga(Rp)
Supra X 125	2017	Rp. 7.999.000
Vario 125	2016	Rp. 14.499.000
Vario 125	2016	Rp. 11.399.000
Revo	2016	Rp. 6.499.000
PCX	2015	Rp. 10.199.000
Revo	2015	Rp. 6.149.000

Setelah seluruh proses selesai, array akan tersusun secara **ascending** sesuai urutan nilai harga motor dari yang terendah hingga tertinggi.

3. HASIL DAN PEMBAHASAN

Hasil dari penelitian ini berupa modul sistem informasi interaktif berbasis web yang dirancang khusus untuk mengimplementasikan dan menganalisis kinerja Algoritma Bubble Sort dalam pengurutan data transaksi motor bekas di Duta Motor Binjai.

Sistem ini bertujuan untuk membantu pengguna, seperti staf administrasi atau manajer, dalam menyusun dan menampilkan data inventaris motor berdasarkan kriteria yang diinginkan dengan cepat.

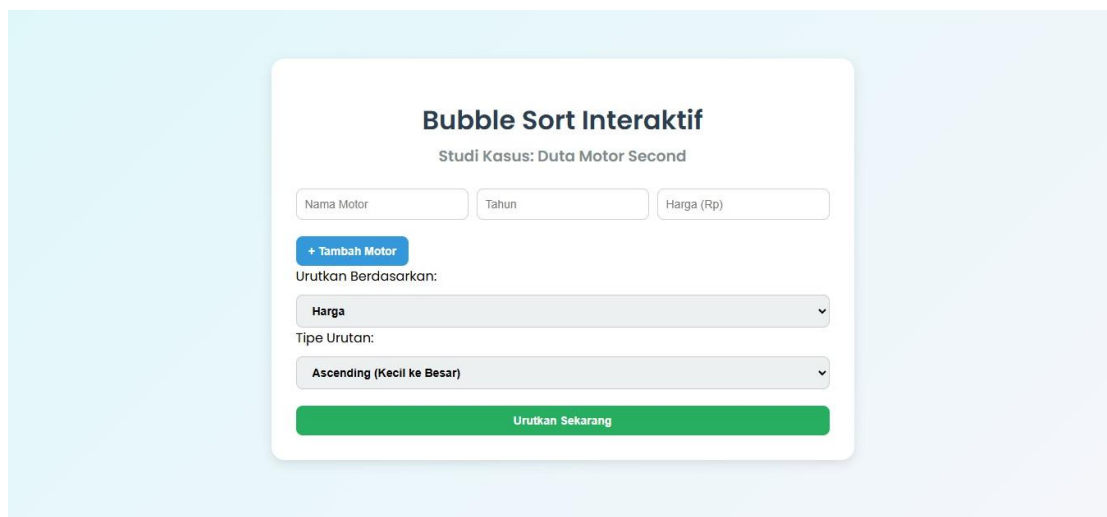
Sistem interaktif ini dikembangkan dengan kerangka kerja berbasis web, menggunakan bahasa pemrograman PHP dan lingkungan lokal XAMPP. Data inventaris motor bekas, yang meliputi Nama Motor, Tahun, dan Harga (Rp), disimpan dan dikelola menggunakan MySQL sebagai basis data utama.

Fitur utama dari sistem ini berpusat pada modul pengurutan. Pengguna dapat:

1. Memasukkan Data: Menambahkan data motor baru (Nama Motor, Tahun, Harga) melalui kolom input yang tersedia.
2. Memilih Kriteria Pengurutan: Data dapat diurutkan berdasarkan salah satu kolom (misalnya, Harga, Tahun, atau Nama Motor).
3. Menentukan Tipe Urutan: Pengguna dapat memilih tipe urutan Ascending (Kecil ke Besar) atau Descending (Besar ke Kecil).

1. Halaman Utama dari Sistem

Gambar 2. Tampilan Utama



Setelah kriteria pengurutan ditetapkan, penekanan tombol "Urutkan Sekarang" memicu eksekusi Algoritma Bubble Sort pada backend sistem. Proses ini dilakukan dengan prinsip perbandingan dan pertukaran elemen yang berdekatan secara berulang kali hingga semua data tersusun. Hasil pengujian fungsional yang ditampilkan pada laman terpisah menunjukkan keberhasilan algoritma. Dalam kasus uji pengurutan berdasarkan Harga secara Ascending, data motor telah tersusun rapi dari harga terendah (mulai dari Revo seharga Rp 6.149) hingga harga tertinggi (Vario 125 seharga Rp 14.499). Validasi fungsional ini membuktikan bahwa Algoritma Bubble Sort telah diimplementasikan secara akurat dan mampu menyajikan data yang terstruktur sesuai dengan kebutuhan Duta Motor Binjai.

2. Halaman Inputan dan Pengurutan dari Sistem

Gambar 3. Tampilan Pengurutan Data Berdasarkan *Ascending*



Bubble Sort Interaktif
Studi Kasus: Duta Motor Second

Supra X 125	2017	7999	
Vario 125	2016	14499	✖
Vario 125	2016	11399	✖
Revo	2016	6499	✖
PCX	2015	10199	✖
Revo	2015	6149	✖

+ Tambah Motor

Urutkan Berdasarkan: **Harga**

Tipe Urutan: **Ascending (Kecil ke Besar)**

Setelah pengguna input isi dari sistem dan kriteria pengurutan ditetapkan, terdapat tombol "Urutkan Sekarang" memicu eksekusi Algoritma Bubble Sort pada backend sistem. Proses ini dilakukan dengan prinsip perbandingan dan pertukaran elemen yang berdekatan secara berulang kali hingga semua data tersusun. Hasil pengujian fungsional yang ditampilkan pada laman terpisah menunjukkan keberhasilan algoritma. Dalam kasus uji pengurutan berdasarkan Harga secara Ascending, data motor telah tersusun rapi dari harga terendah (mulai dari Revo seharga Rp 6.149) hingga harga tertinggi (Vario 125 seharga Rp 14.499). Validasi fungsional ini membuktikan bahwa Algoritma Bubble Sort telah diimplementasikan secara akurat dan mampu menyajikan data yang terstruktur sesuai dengan kebutuhan Duta Motor Binjai.

3. Halaman Hasil dari Pengurutan *Bubble Sort*

Gambar 4. Tampilan Hasil Dari Pengurutan *Bubble Sort*



Hasil Pengurutan Bubble Sort

Kolom Urut: harga
Tipe Urutan: Ascending (Kecil ke Besar)

NAMA MOTOR	TAHUN	HARGA (Rp)
Revo	2015	6.149
Revo	2016	6.499
Supra X 125	2017	7.999
PCX	2015	10.199
Vario 125	2016	11.399
Vario 125	2016	14.499

Total Perbandingan: 15
 Total Pertukaran: 11
 Kompleksitas Waktu: $O(n^2)$

Kembali



Salah satu nilai tambah dari sistem ini adalah kemampuannya menyajikan metrik kinerja kuantitatif yang diperoleh langsung dari proses eksekusi *Bubble Sort*. Berdasarkan data sampel yang diuji, sistem mencatat dan menampilkan angka-angka kunci: Total Perbandingan: 15 dan Total Pertukaran: 11. Hasil ini memberikan data empiris yang valid untuk mengukur biaya operasional algoritma. Sistem juga mengidentifikasi bahwa Algoritma Bubble Sort memiliki Kompleksitas Waktu: $O(n^2)$. Angka perbandingan dan pertukaran ini menjadi data primer untuk menganalisis efisiensi algoritma, yang akan didiskusikan lebih lanjut untuk menyimpulkan kinerja Bubble Sort, terutama ketika diterapkan pada volume data transaksi yang lebih besar di Duta Motor Binjai.

3.1 Implementasi Program Pengurutan Data

Untuk menguji kinerja algoritma *Bubble Sort* dalam sistem *Interaktif Duta Motor Binjai*, dilakukan percobaan pengurutan menggunakan data motor sebagaimana ditunjukkan pada Table 3 berikut.

Table 3. Sample Data Motor Yang akan diuji

Nama Motor	Tahun	Harga(Rp)
Supra X 125	2017	Rp. 7.999.000
Vario 125	2016	Rp. 14.499.000
Vario 125	2016	Rp. 11.399.000
Revo	2016	Rp. 6.499.000
PCX	2015	Rp. 10.199.000
Revo	2015	Rp. 6.149.000

Sebagai contoh, akan dilakukan pengurutan terhadap barang Sepeda Motor berdasarkan “**Harga(Rp)**” menggunakan algoritma *Bubble Sort*

1. Rumus Proses *Bubble Sort*

Secara matematis, proses pengurutan data dengan algoritma Bubble Sort dapat dinyatakan dalam bentuk rumus berikut:

$$\begin{aligned}
 & \text{for } i = 0 \text{ to } n - 1: \\
 & \quad \text{for } j = 0 \text{ to } n - i - 2 : \\
 & \quad \quad \text{if } X[j] > X[j + 1]: \text{ then swap } X[j] \text{ and } X[j + 1]
 \end{aligned}$$

Keterangan :

$X[j]$ = elemen ke-j dari data

n = jumlah total data

swap = proses pertukaran posisi dua elemen jika urutan tidak sesuai

Rumus tersebut menunjukkan bahwa pada setiap iterasi, elemen dengan nilai terbesar akan berpindah ke posisi akhir array (untuk ascending), sedangkan untuk descending, elemen terkecil yang akan berpindah ke posisi akhir.



2. Proses Pengurutan Data Berdasarkan *Ascending*

Langkah-langkah proses pengurutan menggunakan Algoritma *Bubble Sort* terdapat pada table 4 sebagai berikut :

Table 4. Langkah-Langkah Pengurutan Data

[7999.0,14499.0,11399.0,6499.0,10199.0,6149.0]

Fase 1:

7.999.000	14.499.000	11.399.000	6.499.000	10.199.000	6.149.000
14.499.000 > 11.399.000 ditukar					
7.999.000	11.399.000	14.499.000	6.499.000	10.199.000	6.149.000
7.999.000	11.399.000	14.499.000	6.499.000	10.199.000	6.149.000
14.499.000 > 6.499.000 ditukar					
7.999.000	11.399.000	6.499.000	14.499.000	10.199.000	6.149.000
14.499.000 > 10.199.000 ditukar					
7.999.000	11.399.000	6.499.000	10.199.000	14.499.000	6.149.000
14.499.000 > 6.149.000 ditukar					

Fase 2:

7.999.000	11.399.000	6.499.000	10.199.000	6.149.000	14.499.000
11.399.000 > 6.499.000 ditukar sedangkan 14.499.000 tetap					
7.999.000	6.499.000	11.399.000	10.199.000	6.149.000	14.499.000
11.399.000 > 10.199.000 ditukar					
7.999.000	6.499.000	10.199.000	11.399.000	6.149.000	14.499.000
11.399.000 > 6.149.000 ditukar					

**Fase 3:**

7.999.000	6.499.000	10.199.000	6.149.000	11.399.000	14.499.000
-----------	-----------	------------	-----------	------------	------------

10.199.000 > 6.149.000 ditukar sedangkan 11.399.000 tetap

7.999.000	6.499.000	6.149.000	10.199.000	11.399.000	14.499.000
-----------	-----------	-----------	------------	------------	------------

6.499.000 > 6.149.000 ditukar

7.999.000	6.149.000	6.499.000	10.199.000	11.399.000	14.499.000
-----------	-----------	-----------	------------	------------	------------

6.149.000 > 6.499.000 ditukar

Fase 4:

7.999.000	6.149.000	6.499.000	10.199.000	11.399.000	14.499.000
-----------	-----------	-----------	------------	------------	------------

14.499.000 > 10.199.000 ditukar

7.999.000	6.149.000	6.499.000	10.199.000	11.399.000	14.499.000
-----------	-----------	-----------	------------	------------	------------

7.999.000 > 6.149.000 ditukar

Fase 5:

6.149.000	7.999.000	6.499.000	10.199.000	11.399.000	14.499.000
-----------	-----------	-----------	------------	------------	------------

7.999.000 > 6.499.000 ditukar sedangkan 14.499.000

6.149.000	6.499.000	7.499.000	10.199.000	11.399.000	14.499.000
-----------	-----------	-----------	------------	------------	------------

3. Hasil Pengurutan**Table 5. Hasil Pengurutan Data Motor dengan Algoritma *Bubble Sort***

Iterasi	0	1	2	3	4	5
Data Awal	7.999.000	14.499.000	11.399.000	6.499.000	10.199.000	6.149.000
Perbandingan 1	7.999.000	11.399.000	14.499.000	6.499.000	10.199.000	6.149.000
Perbandingan 2	7.999.000	11.399.000	14.499.000	6.499.000	10.199.000	6.149.000
Perbandingan 3	7.999.000	11.399.000	6.499.000	14.499.000	10.199.000	6.149.000
Perbandingan 4	7.999.000	11.399.000	6.499.000	14.499.000	10.199.000	6.149.000
Perbandingan 5	7.999.000	6.499.000	11.399.000	14.499.000	10.199.000	6.149.000
Perbandingan 6	7.999.000	6.499.000	11.399.000	14.499.000	10.199.000	6.149.000
Perbandingan 7	7.999.000	6.499.000	11.399.000	10.199.000	14.199.000	6.149.000
Perbandingan 8	7.999.000	6.499.000	11.399.000	10.199.000	6.149.000	14.499.000



Perbandingan 9	7.999.000	6.499.000	11.399.000	6.149.000	10.199.000	14.499.000
Perbandingan 10	7.999.000	6.499.000	6.149.000	11.399.000	10.199.000	14.499.000
Perbandingan 11	7.999.000	6.499.000	6.149.000	10.199.000	11.399.000	14.499.000
Perbandingan 12	7.999.000	6.149.000	6.499.000	10.199.000	11.399.000	14.499.000
Perbandingan 13	6.149.000	7.999.000	6.499.000	10.199.000	11.399.000	14.499.000
Perbandingan 14	6.149.000	6.499.000	7.999.000	10.199.000	11.399.000	14.499.000
Perbandingan 15	6.149.000	6.499.000	7.999.000	10.199.000	11.399.000	14.499.000

4. Analisis Proses Pengurutan *Bubble Sort*

Untuk menganalisis waktu yang dibutuhkan oleh Algoritma Bubble Sort dalam mengurutkan seluruh dataset, fokus utama adalah pada total jumlah perbandingan yang dilakukan algoritma. Jumlah perbandingan ini menjadi indikator langsung dari beban komputasi.

Total waktu pengurutan ($T_{\text{Pengurutan}}$) dapat dihitung menggunakan rumus berikut:

$$T_{\text{pengurutan}} = C_{\text{total}} \times t$$

Keterangan :

C_{total} = Jumlah total perbandingan yang dilakukan sampai seluruh data terurut sempurna.

t = Waktu rata-rata yang dibutuhkan untuk setiap perbandingan(detik).

$T_{\text{pengurutan}}$ = Total Waktu yang dibutuhkan sistem untuk menyelesaikan proses pengurutan(detik).

Berdasarkan hasil pengujian pada data sampel Duta Motor Binjai (terdiri dari 6 data motor), diperoleh nilai:

C_{total} = 15 kali (Sesuai dengan total perbandingan dari Iterasi 1 hingga 5)

t (waktu rata-rata setiap perbandingan) diasumsikan 0.02 detik

Maka, waktu total pengurutan adalah :

$$T_{\text{pengurutan}} = 15 \times 0,02 = 0,30 \text{ detik}$$

Kesimpulan Analisis Waktu

Hasil perhitungan sebesar 0,30 detik menunjukkan bahwa Algoritma Bubble Sort mampu menyelesaikan proses pengurutan seluruh data transaksi (6 elemen) dengan sangat cepat pada skala data kecil. Nilai waktu respon ini dapat digunakan sebagai dasar untuk membandingkan kinerja sistem secara teoretis dengan hasil pengujian rata-rata sistem (yang mungkin menunjukkan hasil sedikit berbeda karena faktor overhead sistem). Kecepatan ini membuktikan efektivitas Bubble Sort untuk data Duta Motor Binja yang volumenya relatif menengah ke bawah.

KESIMPULAN

Penelitian ini berhasil mengimplementasikan Algoritma Bubble Sort ke dalam sebuah Sistem Interaktif Pengurutan Data Transaksi Motor Bekas Berbasis Web menggunakan teknologi PHP, HTML, dan CSS, yang terbukti akurat dalam menyajikan data terurut (Ascending/Descending) sesuai kriteria (Nama Motor, Tahun, atau Harga). Keberhasilan



fungsi ini didukung oleh analisis kuantitatif pada enam data sampel yang menunjukkan total 15 Perbandingan dan 11 Pertukaran, memvalidasi bahwa algoritma bekerja sesuai dengan teori Kompleksitas Waktu $O(n^2)$. Meskipun efisiensi Bubble Sort rendah untuk dataset besar, kemudahan implementasi dan stabilitasnya menjadikannya relevan untuk sistem berskala kecil hingga menengah seperti transaksi harian di Duta Motor Binjai.

Selain itu, sistem yang dikembangkan juga berkontribusi sebagai media pembelajaran berkat fitur visualisasi proses pengurutan yang interaktif. Dengan demikian, implementasi ini tidak hanya memenuhi kebutuhan pengurutan data transaksi secara praktis, tetapi juga memberikan dasar analisis yang kuat, meskipun disarankan untuk mempertimbangkan algoritma yang lebih efisien (seperti Quick Sort atau Merge Sort) untuk pengembangan sistem skala besar di masa mendatang.

DAFTAR PUSTAKA

- Arbansyah, A., Ilham, M. F. N., Suryawan, S. H., & Wirayuda, P. (2024). Application of Bubble Sort Optimization in New Student Admission Selection Using Brute Force Algorithm. *TEPIAN*, 5(2), 79-86.
<https://doi.org/10.51967/tepihan.v5i2.3055>
- Barokah, M. R. S., Saputra, T., & Sutabri, T. (2025). Perbandingan Kinerja Algoritma Bubble Sort dan Insertion Sort dalam Pengurutan Data Penjualan UMKM. *Jurnal Manajemen Informatika & Teknologi*, 5(1), 184-195.
<https://journal.stiestekom.ac.id/index.php/mifortekh/article/view/756>
- Mahrozi, N., & Faisal, M. (2023). Analisis Perbandingan Kecepatan Algoritma Selection Sort dan Bubble Sort. *Scientica: Jurnal Ilmiah Sains dan Teknologi*, 1(2), 89-98.
<https://doi.org/10.572349/scientica.v1i2.209>
- Nirwan, S., Maulani, M. R., & Mulki, R. M. (2024). Optimasi Pesanan Pelanggan pada Aplikasi Penjualan Online Busana Muslim melalui Penerapan Algoritma Bubble Sort. *Jurnal Teknik Informatika*, 16(1), 8-16.
<https://ejurnal.ulbi.ac.id/index.php/informatika/article/view/3442>
- Panggabean, A. B., Ramadhan, R. R., Perina, I., Toro, Y. L., & Syahputra, A. (2023). Implementasi Algoritma Bubble Sort pada Sistem Pelayanan Perpustakaan Menggunakan Laravel. *Sudo Jurnal Teknik Informatika*, 2(1), 19-27.
<https://doi.org/10.56211/sudo.v2i1.185>
- Reina, R. & Gautama, J. B. (2013). *Perbandingan Bubble Sort dengan Insertion Sort pada Bahasa Pemrograman C dan Fortran*. *ComTech: Computer, Mathematics and Engineering Applications*, 4(2), 1106-1115. DOI:10.21512/comtech.v4i2.2553.
<https://doi.org/10.21512/comtech.v4i2.2553>
- Rihartanto R., Susanto A., Rizal A. (2018). *Performance of Parallel Computing in Bubble Sort Algorithm*. *Indonesian Journal of Electrical Engineering and Computer Science (IJECS)*, 7(3), 861-866. DOI:10.11591/ijeecs.v7.i3.pp861-866
<http://doi.org/10.11591/ijeecs.v7.i3.pp861-866>



- Safrianti, A., & Ezwarsyah. (2024). Optimalisasi Pengurusan Data dengan Algoritma Sorting: Studi Kasus Pengelolaan Stok Barang. *Jurnal Sains dan Teknologi 4.0*, 1(2), 29-36.
<https://jurnal.poltekapp.ac.id/index.php/jst40/article/view/152>
- Wulandari, S. (2025). Pengembangan Sistem Penjualan Interaktif Berbasis Algoritma Sorting. *Jurnal Rekayasa Komputer*, 6(2), 102-112.
<https://jurnal.umt.ac.id/index.php/jrk/article/view/3268>
- Yusuf, A. (2024). *Analisis Algoritma Bubble Sort Ascending/Descending dan Implementasinya Menggunakan Bahasa Pemrograman Python*. *JISCO : Journal of Information System and Computing*, 2(2),
<https://doi.org/10.30631/jisco.v2i2.102>