



Penerapan Algoritma Bubble Sort Dalam Aplikasi Mobile Penentuan Nilai Prestasi Siswa

Cindy Dinda Resiana¹, Yunan Fauzi Wijaya², Ucuk Darusalam^{3*}

^{1,2,3}Sistem Informasi, Teknologi Komunikasi dan Informatika, Universitas Nasional, Indonesia

Email: ucuk.darusalam@civitas.unas.ac.id

Abstract

In high school, the aim is to improve the effectiveness and time efficiency of the student performance assessment process. The solution obtained is to use the bubble sort algorithm, which will be optimized to increase the performance of the bubble sort algorithm and generate more efficient results. In the implementation of the bubble sort algorithm for this research, the sorting is done in descending order, starting from the smallest value. Consequently, the first position with the highest value is obtained as "94" for Nurlita Hasibuan, and the lowest value is "56" for Hendro Mahardika. In this study, the optimization of the algorithm resulted in a time complexity of $O(n \log(n))$, which performed efficiently. The system testing in this research utilized the SUS method, and the final result obtained a SUS score of "83.95", which corresponds to a grade A with an Excellent rating.

Keywords: Bubble Sort Algorithm, Time Complexity, Mobile Application, Online Academic, System Usability Scale (SUS)

Abstrak

Pada sekolah menengah atas yang menjadi tujuan, proses penilaian kinerja siswa yang masih belum efektif dan memakan waktu. Solusi yang didapat yaitu menggunakan algoritma bubble sort yang nanti nya akan dioptimalisasikan dengan cara meningkatkan performa algoritma bubble sort agar waktu yang dihasilkan lebih efisien. Pada penerapan algoritma bubble sort untu penelitian ini mengurutkan dari nilai terkecil (descending), maka diperoleh hasil untuk posisi pertama dengan nilai tertinggi "94" yaitu Nurlita Hasibuan, dan nilai terendah "56" yaitu Hendro Mahardika. Dalam penelitian ini optimalisasi algoritma yang dihasilkan dengan menggunakan kompleksitas waktu yaitu $O(n \log(n))$ dengan efesinsi dengan baik. Proses pengujian sistem di studi ini menggunakan metode SUS dan diperoleh hasil akhir dalam skor SUS pada pengujian aplikasi akademik online, yaitu "83,95" mendapat grade A dengan penilaian Excellent.

Kata kunci: Algoritma Bubble Sort, Kompleksitas Waktu, Aplikasi Mobile, Akademik Online, SUS (System Usability Scale)

1. PENDAHULUAN

Teknologi smartphone termaksud perangkat mobile Android dapat dijadikan sebagai landasan dalam pengembangan aplikasi karena smartphone lebih mudah digunakan, lebih fleksibel, mudah digenggam dan dibawa kemana saja, apalagi saat ini masyarakat lebih sering menggunakannya di perangkat mobile untuk mencari berbagai macam hal informasi, meliputi informasi tentang kegiatan belajar mengajar serta informasi tentang sekolah dan proses pembelajaran di sekolah [1]. Cepatnya perkembangan teknologi informasi sudah membuat dunia ke era baru dari yang tidak terbayangkan sebelumnya. Komputer, yang jadi alat pengolah data, memegang peran penting dalam mengolah dan menyajikan beragam informasi karena jumlah data yang besar [2].

Pada penelitian ini menggunakan algoritma bubble sort, yang memiliki keunggulan yaitu mekanisme yang sederhana sehingga mudah dipahami dan diimplementasikan dalam program aplikasi, kinerjanya juga cukup stabil,

algoritma bubble sort dapat dijalankan tanpa memerlukan penambahan memori dalam komputer [3].

Permasalahan yang terjadi pada sekolah ini karena keterbatasan dalam melakukan penilaian secara manual, sering kali terjadi kesalahan perhitungan sebab jumlah penilaian yang harus dilakukan dan jumlah siswa yang dinilai cukup besar. Dalam pengolahan data, tingkat ketelitian yang sangat penting, sebab kesalahan pada pengolahan data dapat mengakibatkan informasi yang dihasilkan menjadi tidak akurat. Pada intinya, kunci permasalahan ini terletak pada data awal, yaitu nilai siswa sebab data ini merupakan sumber informasi dasar yang harus dikelola dengan baik untuk menghasilkan nilai akhir siswa yang relevan.

Di sekolah menengah atas yang menjadi tujuan, proses penilaian kinerja siswa yang masih belum efektif dan memakan waktu. Solusi yang didapat yaitu penerapan menggunakan algoritma bubble sort dalam aplikasi akademik online dan mengoptimalkan dengan cara meningkatkan performa algoritma bubble sort agar waktu yang dihasilkan lebih efisien. Dalam pengembangan sistem informasi akademik online ini, digunakan framework Flutter yang menggunakan bahasa pemrograman Dart.

Tujuan melakukan penelitian ini menggunakan algoritma bubble sort adalah meningkatkan efisiensi serta meminimalkan kesalahan dalam penentuan nilai prestasi siswa. Dengan menggunakan algoritma Bubble Sort, proses pengurutan data nilai siswa menjadi lebih efisien, dan meningkatkan performa algoritma bubble sort agar waktu yang dihasilkan lebih efisien.

Pengujian sistem penelitian ini memakai metode System Usability Scale (SUS) sudah dipakai serta diuji bertahun-tahun dan terbukti jadi metode yang terpercaya guna mengevaluasi kegunaan (usability) suatu sistem sesuai standar industri. Lewat metode ini, penulis melakukan evaluasi terhadap aplikasi akademik online [4].

2. METODOLOGI PENELITIAN

2.1. Algoritma Bubble Sort

Bubble Sort ialah suatu algoritma pengurutan data yang mengoperasikan lewat membandingkan tiap elemen dengan elemen lainnya. Bila elemen saat ini lebih besar atau lebih kecil dari elemen lainnya, maka posisi (indeks) keduanya akan ditukar. Jika tidak, posisi (indeks) elemen tetap tidak berubah. Dalam algoritma ini, metode yang digunakan adalah membandingkan setiap elemen secara berurutan, dan kemudian melakukan pertukaran jika terdapat elemen yang urutannya belum sesuai. Metode ini dilakukan berulang kali oleh algoritma ini sampai tidak ada lagi data yang belum terurut [5].

2.2. Metode Pengembangan

Studi ini memakai metode Extreme Programming (XP) pada pengembangan perangkat lunak model agile. XP tujuannya guna tingkatkan kualitas perangkat lunak serta responsibilitas pada perubahan dan kebutuhan pelanggan. Fokusnya ialah tingkatkan produktivitas pada pengembangan perangkat lunak [6].

Extreme Programming



Gambar 1. Extreme Programming

Tahapan extreme programming melibatkan beberapa fase, yaitu:

1. Planning

Pada tahap planning ini penulis menetapkan target dan tujuan, perencanaan dimulai dengan keputusan dalam menganalisa rencana pengembangan, kebutuhan apa saja yang perlu di rancangs seperti kebutuhan untuk membuat akademik online menggunakan algoritma bubble sort.

2. Design

Pada langkah ini penulis merancang sistem Unified Modeling Language (UML) seperti flowchart, use case diagram, activity diagram, dan Entity Relationships Diagram (ERD) untuk mengetahui apa saja yang diperlukan guna membuat sistem informasi ini.

3. Coding

Langkah ini adalah tahap dimana kode dibuat dengan menerapkan beberapa praktik extreme programming seperti standar pengkodean, pemrograman berpasangan, integrasi berkelanjutan, dan kepemilikan kode bersama.

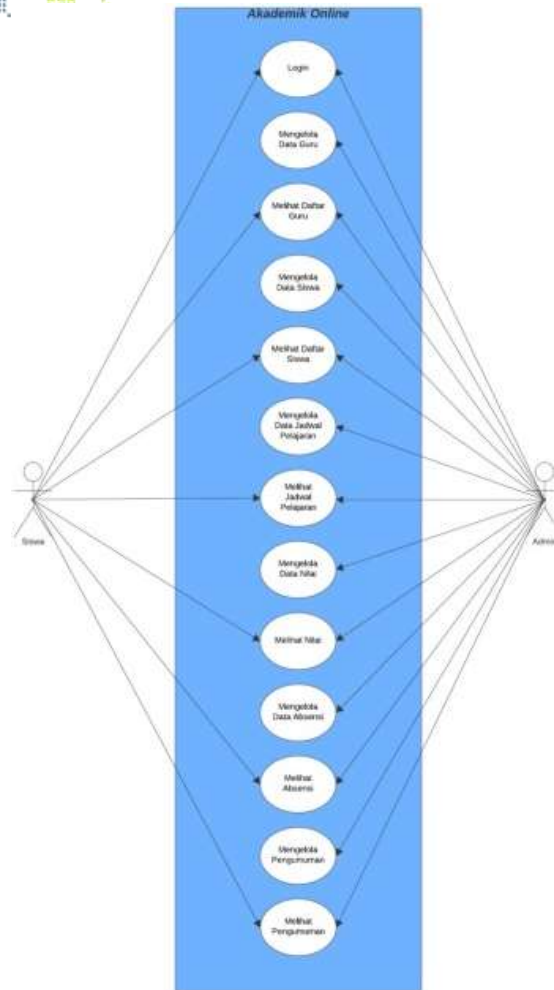
4. Testing

Pada langkah akhir ini penulis melakukan pengujian terhadap sistem informasi akademik online ini dengan menggunakan metode pengujian SUS (System Usability Scale) dengan tujuan supaya program bisa beroperasi sesuai yang diharapkan.

2.3. Desain Penelitian

a) Use Case

Activity diagram merupakan komponen UML yang memodelkan proses yang terjadi dalam sistem [7]. Aktivitas itu bisa berbentuk rangkaian menu atau proses bisnis yang ada pada sistem. Pada Gambar 2.2 terdapat 2 aktor yaitu siswa dan admin. Pada aktor siswa hanya bisa melihat tampilan aplikasi akademik online, sedangkan pada aktor admin dapat merubah data, menambahkan data, dan menghapus data. Adapun operasional activity diagram sistem aplikasi akademik ini yakni:

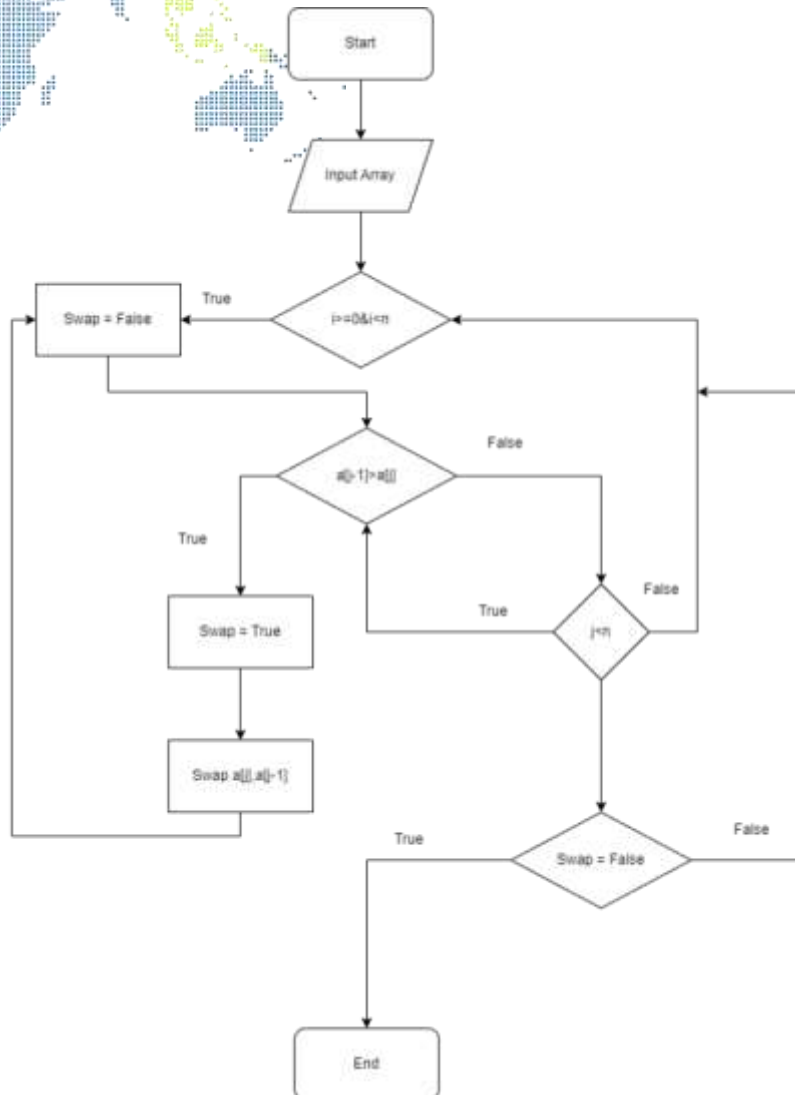


Gambar 2. Use Case

b) Flowchart

Diagram alir, atau juga dikenal sebagai flowchart, ialah diagram alir yang secara logis mengarahkan aliran suatu program system [8]. Pada penelitian ini alur proses sistem yang dilakukan oleh administrator sekolah digambarkan menggunakan flowchart.

Pada Gambar 3 merupakan langkah-langkah dalam alur kerja algoritma bubble sort, yaitu memasukkan nilai array kemudian disini tidak melakukan proses pertukaran karena masih initial state (keadaan awal), selanjutnya jika $a(j-1) > a(j)$ itu benar maka akan melakukan pertukaran menjadi $a(j), a(j-1)$ begitu terus melakukan pertukaran nilai hingga menjadi terurut. Apabila $a(j-1) > a(j)$ salah maka nilai j akan melakukan proses pertukaran dengan nilai yang selanjutnya, jika sudah tidak ada proses pertukaran nilai dan sudah berurutan maka proses akan berakhir. Berikut ini flowchart untuk alur proses algoritma bubble sort adalah sebagai berikut:

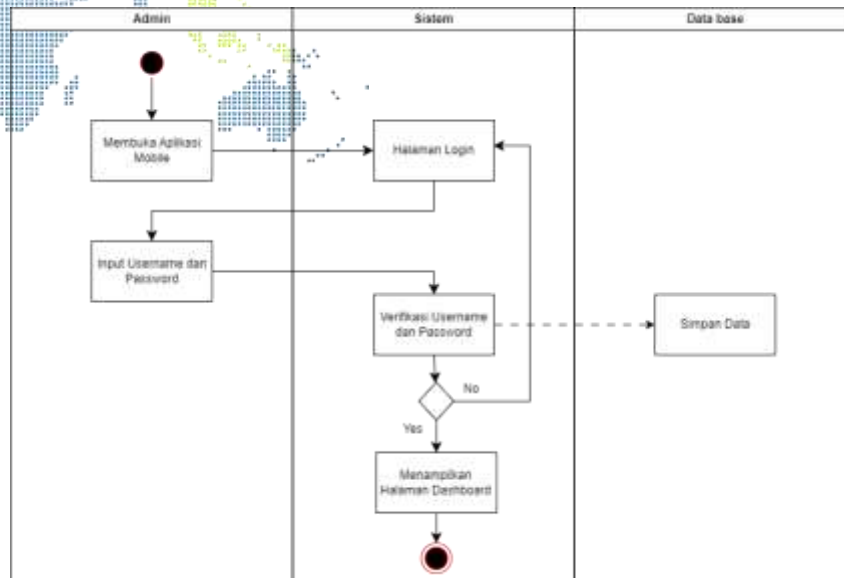


Gambar 3. Flowchart Algoritma Bubble Sort

c) Activity Diagram

Activity Diagram merupakan komponen UML yang memodelkan proses yang terjadi dalam sistem. Aktivitas itu bisa berbentuk rangkaian menu yang ada pada sistem [9].

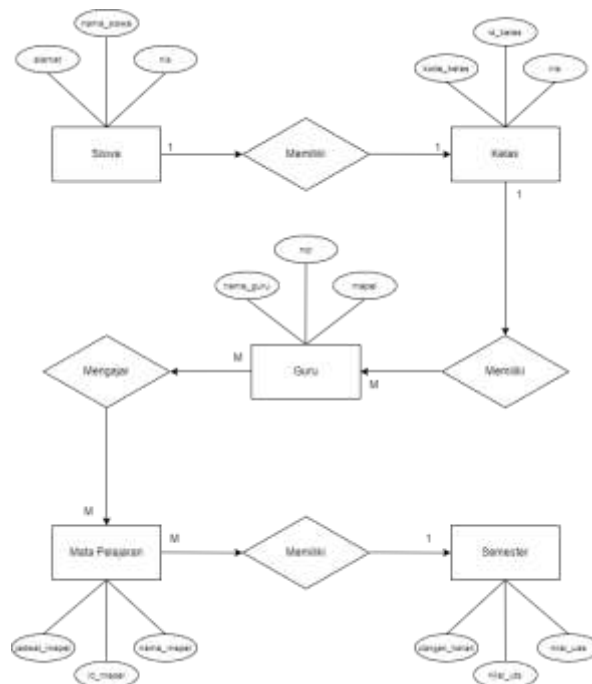
Pada Gambar 4 merupakan aktivitas diagram dalam melakukan login, yaitu admin membuka aplikasi mobile maka sistem akan menampilkan halaman login, setelah itu admin memasukkan username dan password maka sistem akan menyimpan data di database dan melakukan pengecekan apabila username atau password salah maka sistem akan kembali ke halaman login sedangkan apabila benar maka sistem akan menampilkan halaman dashboard. Model pengoperasian sistem aplikasi akademik online ini adalah sebagai berikut:



Gambar 4. Activity Diagram Sistem Akademik Online

d) Entity Relationship Diagram (ERD)

ERD ialah model guna membuat database, supaya lebih mudah guna Gambarkan data dengan hubungan pada suatu bentuk desain [10]. Lewat adanya ERD, sistem database yang terbentuk bisa diGambarkan jadi lebih terstruktur. Pada Gambar 5 menjelaskan setiap satu siswa memiliki satu kelas (one to one), satu kelas memiliki banyak guru (one to many), banyak guru mengajar banyak mata pelajaran (many to many), dan banyak mata pelajaran memiliki satu semester (many to one). Berikut ini ERD dari sistem aplikasi akademik online yaitu:



Gambar 5. ERD Sistem Akademik Online

3. HASIL DAN PEMBAHASAN

3.1. Desain Sistem Interface

a. Halaman Login

Jika user yang sudah memiliki akun terdaftar untuk masuk ke dalam sistem atau layanan yang ditawarkan. Untuk memverifikasi identitas user dengan memasukkan kredensial login yang valid, seperti alamat email dan kata sandi, user dapat mengakses fitur-fitur yang terbatas hanya bagi user yang terotentikasi.



Gambar 6. Halaman Login

b. Halaman Home Page

Pada tampilan halaman home page ini terdapat tampilan admin dan user. Pada tampilan admin terdapat fitur daftar guru, info lomba, daftar nilai, absensi, jadwal pelajaran, dan daftar siswa. Sedangkan pada tampilan user berisikan beberapa fitur, seperti info lomba, daftar guru, daftar nilai, absensi, dan jadwal pelajaran. Tampilan homepage ini didesain semenarik mungkin dan simple sehingga memudahkan user dalam menggunakannya.



Gambar 7. Halaman Home Page

c. Halaman Daftar Guru

Pada halaman daftar guru yang berisikan informasi singkat mengenai daftar guru yakni nama guru, nip, dan mata pelajaran yang diajar. Tampilan admin dapat merubah serta menghapus daftar guru.

**Gambar 8.** Daftar Guru**d. Halaman Info Lomba**

Pada tampilan halaman info lomba ini guru memberikan informasi mengenai lomba dan cara mendaftar lomba yang dapat diikuti oleh siswa, baik lomba antar sekolah atau lomba yang diadakan oleh pemerintah.

**Gambar 9.** Info Lomba**e. Halaman Daftar Nilai**

Pada tampilan halaman daftar nilai terdapat history nilai siswa yang memuat semester, mata pelajaran, ulangan harian, uts, dan uas. Untuk tampilan admin dapat menambah data mata pelajaran yang ingin diinput nilai nya

Daftar Nilai

Tambah Data

No	Mata Pelajaran	Ulangan Harian	UTS	UAS
1	Fisika	80	75	80
2	Kimia	65	75	70

Gambar 10. Daftar Nilai**3.2. Implementasi Sistem**

3.2.1. Cara Kerja Algoritma Bubble Sort

Berikut ini merupakan sample data nilai siswa sebelum menggunakan pengurutan algoritma bubble sort sebagai berikut:



No	Nama Siswa	Kelas	Rata-rata	Action
1	Nisya Zahwa	X - IPA 1	80	
2	Jidan Arya	X - IPA 1	77	
3	Daffa Patibuana	X - IPA 1	83	
4	Divania Fadli	X - IPA 1	80	
5	Hendro Mahardika	X - IPA 1	56	
6	Fahmi Fahreza	X - IPA 1	57	
7	Hidayat Syamsudin	X - IPA 1	64	
8	Nurlita Hasibuan	X - IPA 1	94	
9	Maya Astuti	X - IPA 1	90	
10	Rudi Alamasyah	X - IPA 1	64	

Gambar 11. Sample Data

Pada Gambar 11 menampilkan sample data dari siswa yang berdasarkan nilai ulangan harian, uts, dan uas. Kemudian dikalkulasi hingga mendapatkan nilai rata-rata setiap siswa.

```
void bubbleSort() async{
  List<int> list = await dbHelper.getAverageValues();

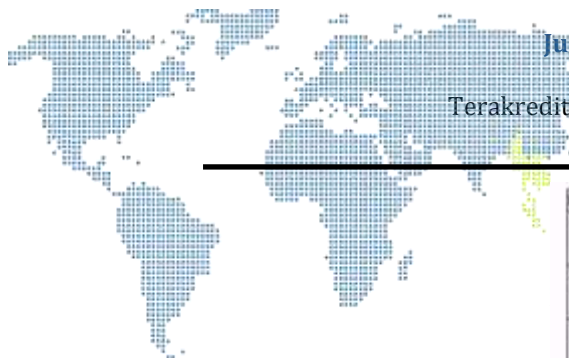
  for (var i in list) {
    for (var j = 0, j = 1; j < list.length; i++, j++) {
      if (list[i] > list[j]) {
        int temp = list[i];

        list[i] = list[j];
        list[j] = temp;
      }
    }
  }

  final sortedData = await dbHelper.getSortedData(list);
  setState(() {
    _dataNilai = sortedData;
  });
}
```

Gambar 12. Source Code Algoritma Bubble Sort

Pada Gambar 12 menerangkan proses untuk melakukan pengurutan terhadap nilai akhir siswa. Dengan pertukaran nilai dari terbesar ke nilai terkecil (descending).



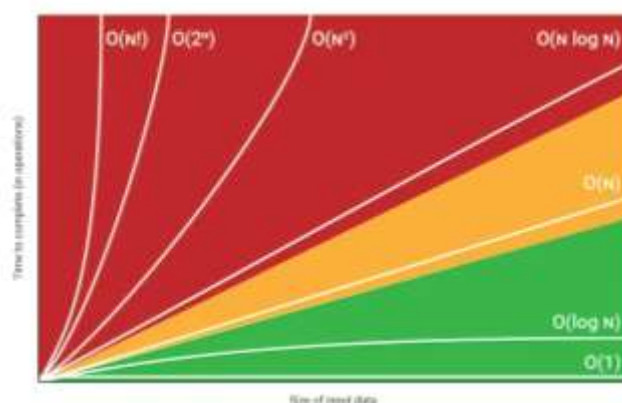
No	Nama Siswa	Kelas	Rata - rata	Action
1	Nurlita Hasibuan	X - IPA 1	94	
2	Maya Astuti	X - IPA 1	90	
3	Daffa Patibuana	X - IPA 1	83	
4	Nisya Zaiwa	X - IPA 1	80	
5	Divania Fadli	X - IPA 1	80	
6	Jidan Arya	X - IPA 1	77	
7	Hisyam Syamsudin	X - IPA 1	64	
8	Rudi Alamasyah	X - IPA 1	64	
9	Fahmi Fahreza	X - IPA 1	57	
10	Hendro Mahardika	X - IPA 1	56	

Gambar 13. Pengurutan Nilai

Pada Gambar 13 didapatkan hasil pengurutan nilai siswa, yaitu sudah diurutkan nilai akhir prestasi siswa. Untuk diurutkan pertama dengan nilai tertinggi '94' yaitu Nurlita Hasibuan, dan nilai terendah '56' yaitu Hendro Mahardika.

3.2.2. Optimalisasi Bubble Sort

Dalam ilmu komputer, kompleksitas waktu sebuah algoritma adalah ukuran seberapa lama waktu yang diperlukan algoritma guna menyelesaikan tugasnya [11]. Dalam konteks ini, menghitung notasi Big-O dari sebagian besar algoritma dalam bidang ilmu komputer memiliki kompleksitas-k sebagai berikut:



Gambar 14. Visualizations of asymptotic time complexity

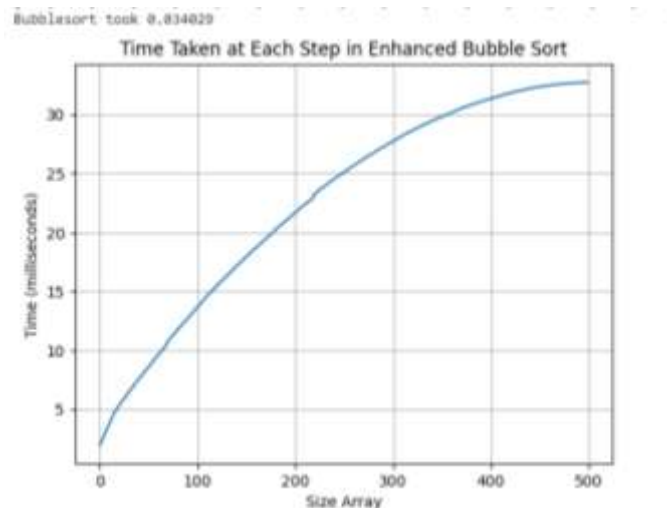
Source: <https://pub.towardsai.net/big-o-notation-what-is-it-69cfd9d5f6b8>

Tabel 1. Classification of time complexities [12]

No.	Complexity	Name	Efficiency
1.	$O(1)$	Constant	Excellent
2.	$O(\log(n))$	Logarithmic	Excellent
3.	$O(n)$	Linear	Great

No.	Complexity	Name	Efficiency
4.	$O(n \log(n))$	$n \log(n)$	Favourable
5.	$O(n^2)$	Quadratic	Slow
6.	$O(n^3)$	Cubik	Slow
7.	$O(a^n)$	Exponential	Extreme
8.	$O(n!)$	Factorial	Terrible

Untuk dapat mengoptimalkan maka dibutuhkan EBS (Enhanced Bubble Sort), dengan mengurangi jumlah perbandingan sebanyak 2, kompleksitas waktu menjadi $O(n \log(n))$, yang jauh lebih baik daripada kompleksitas waktu (n^2) pada bubble sort. Perbedaan antara bubble sort yang sudah dioptimalisasi dengan yang belum mungkin tidak terlihat jelas ketika ukuran array inputnya kecil. Namun, ketika ukuran array menjadi besar maka perbedaan kinerja antara EBS dan BS menjadi nyata, EBS menjadi lebih unggul dalam kecepatan. Dibawah ini hasil algoritma bubble sort yang sudah dioptimalisasi.



Gambar 15. Optimalisasi Bubble Sorts

3.3. Pengujian Sistem

Dalam penelitian ini, digunakan metode evaluasi yang dikenal sebagai System Usability Scale (SUS). Metode ini adalah sebuah kuesioner yang dapat digunakan sebagai alat pengujian untuk mengukur tingkat kegunaan (usability) suatu sistem computer menurut persepsi pengguna [13]. Pada pengujian sistem ini menggunakan metode System Usability Scale atau SUS. Dalam pengujian ini menggunakan sample data 50 responden dengan cara memperoleh skala skor 1 sampai 5 pada setiap 10 pertanyaan kuesioner yang diberikan kepada responden. Pengujian ini menjadi tolak ukur seberapa baik, mudah dalam penggunaan fitur dan desain aplikasi mobile akademik online. Sehingga hasil akhir yang didapatkan dalam skor SUS pada pengujian aplikasi akademik online, yaitu "83,95" mendapat grade A dengan penilaian Excellent.

4. SIMPULAN

Dapat disimpulkan bahwa pada penerapan algoritma bubble sort pada penelitian ini mengurutkan dari nilai terbesar ke nilai terkecil (descending), maka diperoleh hasil posisi pertama dengan nilai tertinggi '94' yaitu Nurlita Hasibuan, dan nilai terendah '56' yaitu Hendro Mahardika. Di studi ini bisa memberi kontribusi pada pengembangan aplikasi mobile untuk melakukan pengurutan dengan menentukan nilai prestasi siswa lewat memakai algoritma bubble sort yang sudah dioptimalisasikan sehingga hasil yang didapatkan dalam kompleksitas waktu, yaitu $O(n \log(n))$ lebih cepat dibandingkan dengan penelitian terdahulu. Hasil pengujian sistem dengan metode SUS mendapat skor SUS "83,95" pada nilai A dengan penilaian excellent.

DAFTAR PUSTAKA

- [1] F. Nuraeni, R. Setiawan, W. Nurhakim, And M. S. Mubarak, "Sistem Informasi Akademik Berbasis Mobile Apps Sebagai Media Informasi Akademik Online," *Jurnal Algoritma*, Vol. 18, No. 2, Pp. 358–366, Jan. 2022, Doi: 10.33364/Algoritma/V.18-2.951.
- [2] M. Hakiki, "Perancangan Sistem Informasi Manajemen Berbasis Sekolah Sma Negeri 1 Muara Bungo," Vol. 6, No. 1, 2021.
- [3] P. S. Maria And E. Susianti, "Implementasi Algoritma Bubble Sort Untuk Ekstraksi Kode Biner Pada Intel Hex File," Vol. 08, No. 04, 2023.
- [4] M. A. Kosim, S. R. Aji, And M. Darwis, "Pengujian Usability Aplikasi Pedulilindungi Dengan Metode System Usability Scale (Sus)," *Sistek*, Vol. 4, No. 2, Aug. 2022, Doi: 10.31326/Sistek.V4i2.1326.
- [5] D. R. Poetra, "Performa Algoritma Bubble Sort Dan Quick Sort Pada Framework Flutter Dan Dart Sdk(Studi Kasus Aplikasi E-Commerce)," *Jatissi*, Vol. 9, No. 2, Pp. 806–816, Jun. 2022, Doi: 10.35957/Jatissi.V9i2.1886.
- [6] L. Ariyanti, M. N. D. Satria, And D. Alita, "Sistem Informasi Akademik Dan Administrasi Dengan Metode Extreme Programming Pada Lembaga Kursus Dan Pelatihan," *Jtsi*, Vol. 1, No. 1, Pp. 90–96, Jun. 2020, Doi: 10.33365/Jtsi.V1i1.214.
- [7] S. Wahyudi, "Pengembangan Sistem Informasi Klinik Berbasis Web," Vol. 06, No. 01, 2020.
- [8] S. Syamsiah, "Perancangan Flowchart Dan Pseudocode Pembelajaran Mengenal Angka Dengan Animasi Untuk Anak Paud Rambutan," *String*, Vol. 4, No. 1, P. 86, Aug. 2019, Doi: 10.30998/String.V4i1.3623.
- [9] T. Wulandari, "Rancang Bangun Sistem Pemesanan Wedding Organizer Menggunakan Metode Rad Di Shofia Ahmad Wedding," Vol. 11, 2022.
- [10] R. Sunantoro And D. Anubhakti, "Analisa Dan Rancangan E-Commerce Pada," *Jurnal Idealis*, Vol. 2, 2019.
- [11] R. R. Basir, "Analisis Kompleksitas Ruang Dan Waktu Terhadap Laju Pertumbuhan Algoritma Heap Sort, Insertion Sort Dan Merge Dengan Pemrograman Java," *String*, Vol. 5, No. 2, P. 109, Dec. 2020, Doi: 10.30998/String.V5i2.6250.
- [12] R. Adin, "Analysis Of Time Complexity In Sorting Algorithms".
- [13] C. Damayanti, A. Triayudi, And I. D. Sholihati, "Analisis Ui/Ux Untuk Perancangan Website Apotek Dengan Metode Human Centered Design Dan System Usability Scale," *Mib*, Vol. 6, No. 1, P. 551, Jan. 2022, Doi: 10.30865/Mib.V6i1.3526.