



***Benchmarking Kualitas Performa Java dan Kotlin di Android  
Runtime Menggunakan Metode Sorting***

**SKRIPSI**

**Diajukan Sebagai Syarat untuk Menyelesaikan  
Pendidikan Program Strata-1 pada  
Program Studi Teknik Informatika**

**Oleh:**

**Muhammad Rizki Saputra**

**2019.11.0024**

**PROGRAM STUDI TEKNIK INFORMATIKA  
FAKULTAS ILMU KOMPUTER DAN SAINS  
UNIVERSITAS INDO GLOBAL MANDIRI  
2024**

***Benchmarking Kualitas Performa Java dan Kotlin di Android  
Runtime Menggunakan Metode Sorting***



**SKRIPSI**

**Diajukan Sebagai Syarat untuk Menyelesaikan  
Pendidikan Program Strata-1 pada  
Program Studi Teknik Informatika**

**Oleh:**

**Muhammad Rizki Saputra**

**2019.11.0024**

**PROGRAM STUDI TEKNIK INFORMATIKA  
FAKULTAS ILMU KOMPUTER DAN SAINS  
UNIVERSITAS INDO GLOBAL MANDIRI**

**2024**

**LEMBAR PENGESAHAN SKRIPSI**

***Benchmarking Kualitas Performa Java dan Kotlin di Android  
Runtime Menggunakan Metode Sorting***

Oleh

**Muhammad Rizki Saputra**

**NPM : 2019.11.0024**


**Palembang, 16 Juli 2024**

**Pembimbing I**



**Rudi Heriansyah, S.T., M.Eng., Ph.D  
NIK : 2022.01.0315**

**Pembimbing II**



**Dwi Asa Verano, M.Kom  
NIK:2000.01.0022**

**Mengetahui,**

**Dekan Fakultas Ilmu Komputer dan Sains**

FAKULTAS ILKOM & SAINS



**Rudi Heriansyah, S.T., M.Eng., Ph.D  
NIK:2022.01.0315**

## LEMBAR PERSETUJUAN DEWAN PENGUJI

Pada hari Kamis tanggal 4 Juli 2024 telah dilaksanakan ujian sidang skripsi :

Nama : Muhammat Rizki Saputra

NPM : 2019.11.0024

Judul : Benchmarking Kualitas Performa Java dan Kotlin di Android  
Runtime Menggunakan Metode Sorting

Oleh Prodi Teknik Informatika Fakultas Ilmu Komputer dan Sains Universitas Indonesia  
Global Mandiri Palembang

Palembang, 16 Juli 2024

Penguji 1,



Dr. Gasim, S.Kom., M.Si

NIK: 2023.01.0340

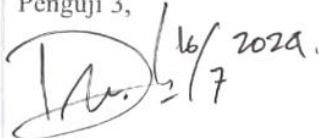
Penguji 2,



Indah Permatasari, M.Kom

NIK: 2021.01.0290

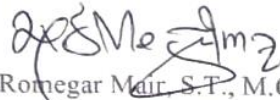
Penguji 3,



Dwi Asa Verano, M.Kom

NIK: 2000.01.0022

Menyetujui,  
Ka. Prodi Teknik Informatika



Zaid Romegar Mair, S.T., M.Cs

NIK: 2021.01.0307



PERSETUJUAN UJIAN SKRIPSI  
FAKULTAS ILMU KOMPUTER DAN SAINS  
FM-PM-10.3/12-02/R0

Program Studi : Teknik Informatika

Nama : Muhammad Rizki Saputra

NPM : 2019110024

Judul : *Benchmarking* Kualitas Performa *Java* dan *Kotlin* di *Android Runtime* Menggunakan Metode *Sorting*

Skripsi telah disetujui untuk dipertahankan di hadapan Tim Penguji Skripsi.

Persetujuan

No	Nama	Tanda Tangan	Tanggal Persetujuan
1.	Rudi Heriansyah, S.T., M.Eng., Ph.D		21/6/24
2.	Dwi Asa Verano, M.Kom		20/6 2024



**SURAT KETERANGAN SIAP SIDANG SKRIPSI  
PROGRAM STUDI TEKNIK INFORMATIKA (SI)  
FASILKOM DAN SAINS UNIVERSITAS INDO GLOBAL MANDIRI**

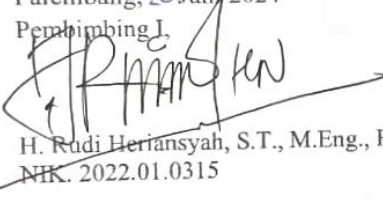
Kami yang bertanda tangan dibawah ini, menerangkan bahwa :

Nama : Muhammat Rizki Saputra  
NPM : 2019110024  
Judul : *Benchmarking* Kualitas Performa *Java* dan *Kotlin* di *Android Runtime*  
Menggunakan Metode *Sorting*

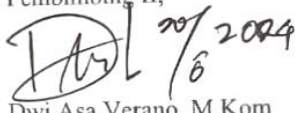
Mahasiswa yang namanya tercantum diatas, telah selesai melakukan penulisan SKRIPSI dan dinyatakan telah memenuhi persyaratan untuk mengikuti sidang SKRIPSI.

Palembang, 20 Juni 2024

Pembimbing I,

  
H. Rudi Heriansyah, S.T., M.Eng., Ph.D  
NIK. 2022.01.0315

Pembimbing II,

  
Dwi Asa Verano, M.Kom  
NIK. 2000.01.0022

Menyetujui,  
Ka. Prodi Teknik Informatika

Zaid Romegar Mair, S.T., M.Cs  
NIK. 2021.01.0307

# ***BENCHMARKING KUALITAS PERFORMA JAVA DAN KOTLIN DI ANDROID RUNTIME MENGGUNAKAN METODE SORTING***

## **ABSTRAK**

Kualitas performa adalah salah satu masalah utama dan biasanya ditemukan di semua aplikasi *mobile*. Banyak kriteria yang mempengaruhi performa dalam pengembangan sebuah aplikasi *mobile*, salah satunya adalah bahasa pemrograman yang berbeda dapat mempengaruhi kualitas performa aplikasi *mobile*. Pada bulan Mei 2017, *Google* mengumumkan dukungannya bahwa bahasa pemrograman *Kotlin* akan menjadi bahasa pengembangan resmi di platform *Android* yang sebelumnya adalah *Java*. Penelitian ini melakukan *benchmarking* performa antara *Java* dan bahasa pemrograman *Android* yang secara resmi adalah *Kotlin*. Untuk mengkarakterisasi *benchmarking*, penulis membuat dua aplikasi *Android*, satu menggunakan *Java* dan satu lagi menggunakan *Kotlin*, dengan fokus pada beberapa aspek yaitu apakah ada perbedaan antara *Java* dan *Kotlin* dalam hal *execution times*, *cpu usage*, *memory usage*, *application size* dan *lines of code* dilingkungan *Android Runtime*. Tiga *algoritma sorting* (*bubble sort*, *insertion sort* dan *shell sort*) diimplementasikan dalam kedua bahasa tersebut untuk mengukur *execution times*, *CPU usage* dan *memory usage*. Dengan menggunakan jumlah *dataset* yang sama, setiap *algoritma* dijalankan sebanyak 100 kali percobaan. Untuk mengukur *application size*, akan menganalisis hasil akhir aplikasi. Terakhir, untuk mengukur *lines of code* menggunakan *plugin "statictic" Android Studio*. Sebagai kesimpulan penulis menemukan bahwa, rata-rata aplikasi *Android Java* lebih cepat dalam *execution time*, dan lebih sedikit penggunaan *resource CPU* dan *memory* dibandingkan *Kotlin*. Peneliti juga menemukan bahwa aplikasi *Android Java* memiliki *application size* yang lebih kecil dibandingkan *Kotlin*. Namun, *Kotlin* memiliki satu keunggulan yaitu, *Line of Code* yang lebih sedikit dibandingkan *Java*

**Kata Kunci :** *Benchmarking*, performa *Java* dan *Kotlin*, *Android Runtime*

# ***BENCHMARKING KUALITAS PERFORMA JAVA DAN KOTLIN DI ANDROID RUNTIME MENGGUNAKAN METODE SORTING***

## ***ABSTRACT***

Performance quality is one of the primary concerns typically encountered in all mobile applications. Numerous factors influence performance in mobile app development, one of which is the choice of programming language, which can significantly affect the quality of application performance. In May 2017, Google announced its support for Kotlin as the official development language for the Android platform, replacing Java. This research conducts performance benchmarking between Java and the officially designated Android programming language, Kotlin. To characterize the benchmarking process, the author develops two Android applications, one using Java and the other Kotlin, focusing on several aspects including differences in execution times, CPU usage, memory usage, application size, and lines of code within the Android Runtime environment. Three sorting algorithms (bubble sort, insertion sort, and shell sort) are implemented in both languages to measure execution times, CPU usage, and memory usage. Using the same dataset, each algorithm is executed 100 times. Application size is measured by analyzing the final application output. Finally, lines of code are measured using the "statictic" plugin in Android Studio. In conclusion, the author found that, on average, Java Android applications had faster execution times and used fewer CPU and memory resources compared to Kotlin. The researcher also found that Java Android applications had a smaller application size compared to Kotlin. However, Kotlin has one advantage: fewer lines of code compared to Java.

***Keywords*** : *Benchmarking, performa Java dan Kotlin, Android Runtime*



## KATA PENGANTAR

Puji dan Syukur penulis persembahkan kepada Tuhan Yang Maha Esa karena akhirnya dapat menyelesaikan penulisan skripsi ini dengan baik dan bisa terselesaikan dengan tepat pada waktunya. Skripsi yang penulis buat dengan judul *BENCHMARKING KUALITAS PERFORMA JAVA DAN KOTLIN DI ANDROID RUNTIME MENGGUNAKAN METODE SORTING* dibuat sebagai salah satu syarat untuk menyelesaikan studi pada Program Studi Teknik Informatika (TI), Fakultas Ilmu Komputer dan Sains, Universitas Indo Global Mandiri (UIGM). Tidak lupa penulis mengucapkan terima kasih atas bantuan yang diberikan selama penyusunan skripsi ini kepada:

1. Bapak Dr. Marzuki Alie, SE., MM, selaku Rektor Universitas Indo Global Mandiri Palembang
2. Bapak Rudi Heriansyah, S.T., M.Eng., Ph.D sebagai Dekan Fakultas Ilmu Komputer dan Sains Universitas Indo Global Mandiri dan sekaligus Dosen Pembimbing I.
3. Bapak Zaid Romegar Mair, S.T., M.Cs sebagai Ketua Program Studi Teknik Informatika dan sekaligus Pembimbing Akademik
4. Bapak Dwi Asa Verano, M.Kom sebagai dosen Pembimbing II
5. Dosen-dosen yang ada di Fakultas Ilmu Komputer dan Sains Universitas IGM

Dengan segala kerendahan hati penulis menyadari bahwa skripsi ini masih jauh dari kata sempurna, oleh karena itu dibutuhkan kritik dan saran untuk perbaikan dan pengembangan skripsi ini sangat diharapkan. Akhir kata, semoga skripsi ini bermanfaat bagi semua pihak terima kasih.

Penulis,

## DAFTAR ISI

HALAMAN JUDUL LUAR	
HALAMAN JUDUL DALAM	
LEMBAR PENGESAHAN SKRIPSI.....	iii
LEMBAR PERSETUJUAN DEWAN PENGUJI .....	iv
SURAT PERSETUJUAN UJIAN SKRIPSI.....	v
SURAT KETERANGAN SIAP SIDANG SKRIPSI.....	vi
ABSTRAK .....	vii
<i>ABSTRACT</i> .....	viii
KATA PENGANTAR.....	ix
DAFTAR ISI.....	x
DAFTAR TABEL .....	xiv
DAFTAR GAMBAR .....	xv
DAFTAR LAMPIRAN.....	xvii
BAB 1 PENDAHULUAN .....	1
1.1    Latar Belakang .....	1
1.2    Rumusan Masalah.....	3
1.3    Batasan Masalah .....	4
1.4    Tujuan Penelitian .....	4
1.5    Manfaat Penelitian .....	4
1.6    Sistematika Penulisan .....	5
BAB II LANDASAN TEORI .....	7
2.1 <i>Benchmarking</i> .....	7

2.2	<i>Java</i> .....	7
2.3	<i>Kotlin</i> .....	8
2.4	Perbedaan Kode <i>Java</i> dan <i>Kotlin</i> .....	10
2.5	<i>Android Runtime</i> .....	13
2.6	<i>Android Application Package (APK)</i> .....	14
2.7	<i>JavaScript Object Notation (JSON)</i> .....	15
2.8	Metode <i>Sorting</i> .....	15
	2.8.1 <i>Algoritma Bubble Sort</i> .....	15
	2.8.2 <i>Algoritma Insertion Sort</i> .....	20
	2.8.3 <i>Algoritma Shell Sort</i> .....	21
2.9	Penelitian Terdahulu .....	24
2.10	<i>Standar ISO/IEC 25010</i> .....	26
2.11	<i>Android</i> .....	26
2.12	<i>Snapdragon Profiler</i> .....	28
BAB III METODE PENELITIAN.....		29
3.1	Tahapan Penelitian.....	29
3.2	Analisis Masalah.....	29
3.3	Studi Literatur .....	30
3.4	Pendefinisian <i>Benchmarking</i> .....	31
	3.4.1 Parameter Performa .....	31
	3.4.2 Pengukuran Performa .....	32
	3.4.3 Perangkat Pengujian Performa Aplikasi .....	33
3.5	Pengembangan Aplikasi.....	34
	3.5.1 Analisis .....	34

3.5.1.1	Kebutuhan <i>Hardware</i> dan <i>Software</i> .....	34
3.5.1.2	Kebutuhan Fungsional .....	35
3.5.1.3	Kebutuhan <i>Non-fungsional</i> .....	35
3.5.2	Perancangan .....	36
3.5.1	Perancangan Desain Aplikasi.....	36
3.5.2	Perancangan <i>Algoritma</i> .....	38
3.5.3	Implementasi .....	45
3.5.4	Pengujian .....	45
3.6	Eksperimen .....	46
3.6.1	<i>Execution Times</i> .....	47
3.6.2	<i>CPU Usage</i> .....	48
3.6.3	<i>Memory Usage</i> .....	49
3.6.4	<i>Application Size</i> .....	50
3.6.5	<i>Line of Code (LOC)</i> .....	51
3.7	Analisis Hasil Eksperimen.....	51
3.8	Penarikan Kesimpulan .....	52
BAB IV HASIL DAN PEMBAHASAN.....		53
4.1	Implementasi.....	53
4.2	Pengujian .....	57
4.2.1	<i>Blackbox Testing</i> .....	57
4.2.2	<i>Whitebox Testing</i> .....	64
4.3	Analisis Hasil Eksperimen.....	83
4.3.1	<i>Execution Times</i> .....	83
4.3.2	<i>CPU</i> .....	87

4.3.3	<i>Memory</i> .....	94
4.3.4	<i>Application Size</i> .....	101
4.3.5	<i>Line of Codes</i> .....	103
BAB V KESIMPULAN DAN SARAN.....		104
5.1	Kesimpulan .....	104
5.2	Saran .....	105
DAFTAR PUSTAKA .....		106

## DAFTAR TABEL

**Tabel 2.1** Simulasi *Algoritma Bubble Sort*

**Tabel 2.2** Simulasi *Algoritma Insertion Sort*

**Tabel 2.3** Penelitian Terdahulu

**Tabel 2.4** Sejarah *Versi Android*

**Tabel 3.1** Parameter Performa yang Diukur

**Tabel 3.2** Pengukuran Performa

**Tabel 3.3** Kebutuhan *Hardware* untuk Pengembangan Aplikasi

**Tabel 3.4** Kebutuhan *Software* untuk Pengembangan Aplikasi

**Tabel 4.1** *Output* yang Diharapkan

**Tabel 4.2** Hasil Pengujian Fungsional Aplikasi

**Tabel 4.3** 25 Sampel Penggunaan *CPU Java* dan *Koltin*

**Tabel 4.4** Hasil Penggunaan *CPU Java* dan *Koltin*

**Tabel 4.5** 25 Sampel Penggunaan *Memory Java* dan *Kotlin*

**Tabel 4.6** Hasil Penggunaan *Memory Java* dan *Kotlin*

## DAFTAR GAMBAR

- Gambar 2.1** Perbedaan *Type Declaration* di *Java* dan *Kotlin*
- Gambar 2.2** Perbedaan *Nullable Type Declaration* di *Java* dan *Kotlin*
- Gambar 2.3** Perbedaan *Null Safety* di *Java* dan *Kotlin*
- Gambar 2.4** Perbedaan *String Template* di *Java* dan *Kotlin*
- Gambar 2.5** Perbedaan *Function Declaration* di *Java* dan *Kotlin*
- Gambar 2.6** *Smart Casting* di *Java* dan *Kotlin*
- Gambar 2.7** *Instance Object* di *Java* dan *Kotlin*
- Gambar 2.8** Karakteristik Kualitas Aplikasi *ISO/IEC 25010*
- Gambar 2.9** Tampilan *Software Snapdragon Profiler*
- Gambar 3.1** Tahapan Penelitian
- Gambar 3.2** Rancangan Desain Aplikasi *Kotlin*
- Gambar 3.3** Rancangan Desain Aplikasi *Java*
- Gambar 3.4** *Flowchart* Program Algoritma *Bubble Sort*
- Gambar 3.5** *Pseudocode Bubble Sort*
- Gambar 3.6** *Flowchart* Program Algoritma *Insertion Sort*
- Gambar 3.7** *Pseudocode Insertion Sort*
- Gambar 3.8** *Flowchart* Program Algoritma *Shell Sort*
- Gambar 3.9** *Pseudocode Shell Sort*
- Gambar 3.10** Contoh Proses *CPU* pada *Realtime Performance Analysis*
- Gambar 3.11** Contoh Proses *Memory* pada *Realtime Performance Analysis*
- Gambar 3.12** Tampilan *Plugin Statistic*
- Gambar 4.1** Implementasi Aplikasi *Android Java* dan *Kotlin* Saat  
Pertamakali Dijalankan
- Gambar 4.2** Implementasi Aplikasi *Android Java* dan *Kotlin* Sebelum  
Memulai Prose *Benchmarking*
- Gambar 4.3** Implementasi Aplikasi *Android Java* dan *Kotlin* Setelah Proses  
*Benchmarking* Se

## DAFTAR GAMBAR

- Gambar 4.4** Implementasi Aplikasi *Android Java* dan *Kotlin* Hasil *Execution Times*
- Gambar 4.5** Hasil Pengujian Fungsional Aplikasi *Java* dan *Kotlin* REQ.1
- Gambar 4.6** Hasil Pengujian Fungsional Aplikasi *Java* dan *Kotlin* REQ.1
- Gambar 4.7** Hasil Pengujian Fungsional Aplikasi *Java* dan *Kotlin* REQ.2
- Gambar 4.8** Hasil Pengujian Fungsional Aplikasi *Java* dan *Kotlin* REQ.3
- Gambar 4.9** Hasil Pengujian Fungsional Aplikasi *Java* dan *Kotlin* REQ.4
- Gambar 4.10** Hasil Pengukuran Performa *Execution Times Java* dan *Kotlin* saat menjalankan *Bubble Sort*
- Gambar 4.11** Hasil Pengukuran Performa *Execution Times Java* dan *Kotlin* saat menjalankan *Insertion Sort*
- Gambar 4.12** Hasil Pengukuran Performa *Execution Times Java* dan *Kotlin* saat menjalankan *Shell Sort*
- Gambar 4.13** Rekapitulasi Data Performa *Execution Time Java* dan *Kotlin*
- Gambar 4.14** Hasil Pengukuran Performa *CPU Java* dan *Kotlin* saat menjalankan *Bubble Sort*
- Gambar 4.15** Hasil Pengukuran Performa *CPU Java* dan *Kotlin* saat menjalankan *Insertion Sort*
- Gambar 4.16** Hasil Pengukuran Performa *CPU Java* dan *Kotlin* saat menjalankan *Shell Sort*
- Gambar 4.17** Rekapitulasi Data Performa *CPU Java* dan *Kotlin*
- Gambar 4.18** Hasil Pengukuran Performa *Memory Java* dan *Kotlin* saat menjalankan *Bubble Sort*
- Gambar 4.19** Hasil Pengukuran Performa *Memory Java* dan *Kotlin* saat menjalankan *Insertion Sort*
- Gambar 4.20** Hasil Pengukuran Performa *Memory Java* dan *Kotlin* saat menjalankan *Shell Sort*
- Gambar 4.21** Rekapitulasi Data Performa *Memory Java* dan *Kotlin*



## DAFTAR GAMBAR

**Gambar 4.22** Perincian Hasil Build *APK Java* dan *Kotlin*

**Gambar 4.23** Data *Application Size Java* dan *Kotlin*

**Gambar 4.24** Hasil Pengukuran *Line of Codes Java* dan *Kotlin* dengan Plugin *Statistic*

**Gambar 4.25** Data *Line of Codes Java* dan *Kotlin*

**Gambar 5.1** Rekapitulasi Performa *Java* dan *Kotlin*

## **DAFTAR LAMPIRAN**

Lampiran 1 : Surat Tidak Plagiat

Lampiran 2 : Kartu Bimbingan

Lampiran 3 : Surat keterangan revisi skripsi

Lampiran 4 : curriculum vitae

