



# **PERBANDINGAN PERFORMA APLIKASI ANDROID NATIVE MENGGUNAKAN ARSITEKTUR MVP DAN MVVM**



## ***Comparison of Android Native Application Performance Using MVP and MVVM Architecture***



### **TUGAS AKHIR**

Laporan ini disusun untuk memenuhi salah satu syarat menyelesaikan pendidikan Diploma Empat Program Studi Teknik Informatika di  
Jurusan Teknik Komputer dan Informatika



**Oleh:**

**Ujang Wahyu**

**NIM: 151524031**



**POLBAN**



**POLITEKNIK NEGERI BANDUNG**

**2019**




**PERBANDINGAN PERFORMA APLIKASI ANDROID  
NATIVE MENGGUNAKAN ARSITEKTUR MVP DAN MVVM**

*Comparison of Android Native Application Performance Using  
MVP and MVVM Architecture*

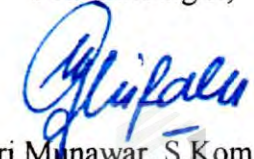
Oleh:  
**UJANG WAHYU**  
NIM. 151524031

Menyetujui  
Bandung, Agustus 2019

Pembimbing I,

  
Bambang Wisnuadhi, S.Si., M.T.  
NIP.19720106 199903 1 002

Pembimbing II,

  
Ghifari Munawar, S.Kom., M.T.  
NIP.19860412 201404 1 001

Ketua Jurusan Teknik Komputer dan Informatika,



Bambang Wisnuadhi, S.Si., M.T.  
NIP.19720106 199903 1 002



**PERBANDINGAN PERFORMA APLIKASI ANDROID  
NATIVE MENGGUNAKAN ARSITEKTUR MVP DAN MVVM**



*Comparison of Android Native Application Performance using  
MVP and MVVM Architecture*



Oleh:  
**UJANG WAHYU**  
NIM: 151524031



Tugas Akhir ini telah disidangkan pada tanggal 22 Juli 2019 sesuai dengan ketentuan.

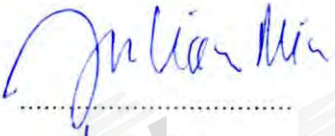


Ketua

Joe Lian Min, M.Eng.  
NIP. 19661018 199512 1 001

Anggota

Urip Teguh Setijohatmo, BSCS., M.Kom.  
NIP. 19600928 199403 1 001

  
.....  
  
.....





## PERNYATAAN PENULIS

Dengan ini menyatakan bahwa laporan Tugas Akhir dengan judul Perbandingan Performa Aplikasi Android Native Menggunakan Arsitektur MVP dan MVVM adalah karya ilmiah yang bebas dari unsur tindakan plagiarisme, dan sesuai dengan ketentuan tata tulis yang berlaku.

Apabila dikemudian hari ditemukan adanya unsur plagiarisme, maka hasil penilaian dari Tugas Akhir ini dicabut dan bersedia menerima sanksi sesuai dengan ketentuan yang berlaku.

Demikian pernyataan ini dibuat dengan sesungguhnya dalam keadaan sadar sepenuhnya.

Bandung, Agustus 2019



Ujang Wahyu

NIM: 151524031



Feedback Studio - Google Chrome  
https://ev.tumitin.com/app/carta/en\_us/?e=1158053100&lang=en\_us&s=1&u=1065722671

PERBANDINGAN PERFORMA APLIKASI ANDROID  
NATIVE MENGGUNAKAN ARSITEKTUR MVP DAN MVVM

*Comparison of Android Native Application Performance Using  
MVP and MVVM Architecture*

Oleh:  
UJANG WAHYU  
NTM. 151524031

Menyetujui  
Bandung, 2019

Pembimbing I: Pembimbing II:

Page 2 of 166 Word Count: 28671

Match Overview

Rank	Team	Score
1	Dark Blue (1)	1%
2	www.tumitin.com	<1%
3	www.pikmadikare.com	<1%
4	www.pikmadikare.com	<1%
5	www.pikmadikare.com	<1%
6	www.pikmadikare.com	<1%
7	www.pikmadikare.com	<1%
8	www.pikmadikare.com	<1%
9	www.pikmadikare.com	<1%
10	www.pikmadikare.com	<1%

Page 2 of 166 Word Count: 28671

Page 2 of 166 Word Count: 28671







Nama : Ujang Wahyu

NIM : 151524031

Tempat, Tanggal Lahir : Bandung, 01 Agustus 1996.

SD Lulus Tahun : 2009 dari SDN Cibeber.

SLTP Lulus Tahun : 2012 dari SMPN 1 Kalibunder.

SLTA Lulus Tahun : 2015 dari SMAN 1 Jampangkulon.

Prestasi yang pernah dicapai :

- Juara 1 Mobile App Development Vocompest 2018
- Finalis Hackathon Indonesia Android Kejar By Google
- Finalis Hackathon Bandung Lautan API By Bekraf



## ABSTRAK

Perangkat android adalah perangkat yang saat ini sangat terbatas baik dari daya, memori, maupun resource lainnya. Performa adalah aspek yang saat ini harus diperhatikan dalam pengembangan aplikasi android agar aplikasi yang dibangun dapat berjalan dengan maksimal pada perangkat android. Arsitektur yang digunakan dalam pengembangan aplikasi dapat berpengaruh terhadap performa aplikasi. Hal ini menjadi tantangan besar bagi para mobile developer untuk membuat aplikasi dengan performa yang baik.

Penelitian ini bertujuan untuk mengetahui arsitektur terbaik pada aplikasi android berdasarkan performanya. Arsitektur yang diteliti adalah MVP dan MVVM. Adapun aspek performa yang diukur adalah CPU usage, memory usage dan execution time aplikasi pada perangkat android.

Hasil eksperimen dan analisis menunjukkan bahwa aplikasi dengan arsitektur MVVM memiliki ukuran nilai yang lebih kecil pada kedua aspek performa yang diukur yaitu CPU usage dan Execution time. CPU usage aplikasi MVVM lebih rendah sekitar 0.5-1%. Execution time aplikasi MVVM lebih cepat sekitar 1-145 ms, sedangkan pada aspek memory usage MVVM lebih besar sekitar 0.5-1 Mb. Artinya, penggunaan arsitektur MVVM lebih baik pada dua aspek yaitu CPU usage dan execution time, sedangkan pada aspek memory usage arsitektur MVVM lebih buruk. Alasan arsitektur MVVM unggul dalam *CPU usage* dan *execution time* dikarenakan adanya *databinding* yang mengakibatkan proses menjadi lebih singkat dan cepat. Akan tetapi, pada *memory usage* akan lebih besar dikarenakan adanya *library* tambahan yaitu *databinding*.

**Kata kunci:** arsitektur MVP, arsitektur MVVM, *CPU usage*, *memory usage*, *execution time*.





## **ABSTRACT**

*Android devices are devices that are currently very limited in terms of storage, power, memory and other resources. Performance is an aspect that currently must be considered in developing an android application in order to get maximum functionality on an android device. The architecture used in application development can support the application performance. This is a big challenge for mobile developers to create applications with good performance.*

*This research is intended to determine the best architecture on Android applications based on performance. The architecture under study is MVP and MVVM. The performance aspects measured are CPU usage, memory usage and application execution time on an Android device.*

*Experiments results and analyzes showed, that the application with MVVM architecture has a smaller value on the two aspects of performance needed, namely CPU usage and execution time. CPU usage of applications are lower around 0.5 - 1%. MVVM application execution time is faster around 1 -145 ms, whereas in the memory usage MVVM is bigger than MVP around 0.5 - 1 Mb. So it can be concluded that the use of MVVM architecture is better in two aspects, namely CPU usage and execution time, while in the aspect of memory usage worse. The reason MVVM architecture is superior in CPU usage and execution time is because there is a data binding process that becomes shorter and faster. However, the use of memory will be greater because there is an additional library of data pooling.*

**Keyword:** *MVP architecture, MVVM architecture, CPU usage, memory usage, execution time*







## KATA PENGANTAR



Assalamu'alaikum Wr. Wb.



Puji dan syukur penulis panjatkan kehadirat Allah SWT yang telah memberi rahmat dan karunia-Nya, tidak lupa shalawat dan salam penulis curah limpahkan kepada Nabi besar Muhammad SAW beserta para keluarga dan sahabatnya, berkat bantuan dan dorongan dari semua pihak yang telah membantu selesainya Tugas Akhir dengan judul “PERBANDINGAN PERFORMA APLIKASI ANDROID NATIVE MENGGUNAKAN ARSITEKTUR MVP DAN MVVM Pada kesempatan ini penulis menyampaikan rasa terima kasihnya atas segala petunjuk, bimbingan, dan bantuannya kepada:

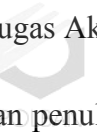


1. Bapak Bambang Wisnuadhi, S.Si., M.T., selaku dosen pembimbing sekaligus Ketua Jurusan Teknik Komputer dan Informatika yang telah membimbing dan memberikan motivasi selama penelitian Tugas Akhir;
2. Ibu Santi Sundari, S.Si., M.T., selaku Ketua Program Studi D-IV Teknik Informatika atas dukungannya;
3. Bapak Ghifari Munawar, S.Kom., M.T. selaku dosen pembimbing yang telah membimbing dan memberikan motivasi selama penelitian Tugas Akhir;
4. Bapak Ade Chandra Nugraha, S.Si., M.T., Ibu Ani Rahmani, S.Si., M.T. dan Bapak Suprihanto, BSEE., M.Sc selaku Koordinator Tugas Akhir yang telah memberikan arahan dan motivasi dalam penyusunan Tugas Akhir;
5. Ibu Transmissia Semiawan, BSCS., MIT., Ph.D. selaku dosen yang memberikan pencerahan penelitian pada mata kuliah Metodologi Penelitian;
6. Keluarga yang selalu memberikan dukungannya yang besar khususnya selama penulis melaksanakan praktik kerja lapangan;
7. Rekan-rekan mahasiswa/i D-IV Teknik Informatika angkatan 2015 yang telah memberikan dukungan dalam menyelesaikan penelitian dan penulisan laporan Tugas Akhir;

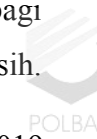
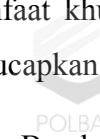
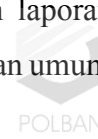




8. Serta seluruh pihak yang telah memberikan do'a dan dukungan demi kelancaran dan kemudahan saya selama mengerjakan penelitian dan penulisan laporan Tugas Akhir.



Mengingat akan kemampuan penulis yang sangat terbatas, penulis mengharapkan kritik dan sarannya dari semua pihak demi peyempurnaan pembuatan laporan in.



Demikian laporan ini penulis susun, semoga dapat bermanfaat khususnya bagi penulis dan umumnya untuk semua pihak. Akhir kata penulis ucapkan terima kasih.

Bandung, Juli 2019

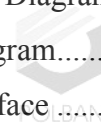
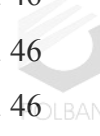


Ujang Wahyu

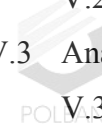
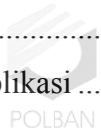


## DAFTAR ISI

Abstrak .....	i
<i>Abstract</i> .....	ii
Kata Pengantar .....	iii
Daftar Isi .....	v
Daftar Gambar .....	viii
Daftar Tabel .....	x
Daftar Istilah .....	xiii
Bab I Pendahuluan .....	1
I.1 Latar Belakang .....	1
I.2 Rumusan Masalah .....	3
I.3 Research Question .....	3
I.4 Hipotesis .....	3
I.5 Tujuan .....	3
I.6 Manfaat .....	4
I.7 Ruang Lingkup dan Batasan Penelitian .....	4
I.1 Sistematika Penulisan .....	5
Bab II Tinjauan Pustaka .....	7
II.1 Dasar Teori .....	7
II.1.1 Android .....	7
II.1.2 Aplikasi Native .....	7
II.1.3 Konsep Android Native .....	8
II.1.4 Performa .....	8
II.1.5 Bahasa Pemrograman Android Native .....	11
II.1.6 Arsitektur MVP .....	12
II.1.7 Arsitektur MVVM .....	15
II.1.8 Point Of Sales .....	20
II.2 Karya Ilmiah Terkait .....	22
Bab III Metodologi Penelitian .....	24



III.1 Jenis Penelitian .....	24
III.2 Variabel Penelitian .....	24
III.3 Objek Penelitian .....	25
III.4 Data Penelitian .....	25
III.5 Perangkat Pendukung .....	25
III.6 Tahapan Penelitian .....	25
III.6.1 Studi Pustaka .....	27
III.6.2 Eksplorasi Tools Untuk Memonitoring Performa .....	27
III.6.3 Analisis Arsitektur MVP dan MVVM .....	28
III.6.4 Penentuan Aplikasi Objek Penelitian .....	29
III.6.5 Pengumpulan Data .....	31
III.6.6 Pengembangan Aplikasi Objek Penelitian .....	31
III.6.7 Rancangan Eksperimen .....	36
III.6.8 Pelaksanaan Eksperimen .....	44
III.6.9 Evaluasi Hasil Eksperimen .....	45
Bab IV Pengembangan Objek Penelitian .....	46
IV.1 Deskripsi Aplikasi .....	46
IV.2 Analisis .....	46
IV.2.1 Analisis Arsitektur MVP dan MVVM .....	46
IV.2.2 Analisis Existing System .....	53
IV.2.3 Analisis Aplikasi To-be .....	55
IV.3 Perancangan .....	64
IV.3.1 Perancangan Model Data .....	64
IV.3.2 Perancangan Struktur Data .....	65
IV.3.1 Perancangan Arsitektur Diagram .....	66
IV.3.2 Perancangan Class Diagram .....	68
IV.3.3 Perancangan User Interface .....	68
IV.3.4 Perancangan Model Perilaku Aplikasi .....	75
IV.3 Implementasi .....	76
IV.3.5 Implementasi Aplikasi MVP .....	79
IV.3.6 Implementasi Aplikasi MVVM .....	83



IV.4 Pengujian .....	86
Bab V Hasil dan Pembahasan .....	87
V.1 Hasil Eksperimen .....	87
V.1.1 Eksperimen Berdasarkan Fungsional Aplikasi .....	87
V.1.2 Eksperimen Berdasarkan Jumlah Data .....	95
V.2 Pembahasan Hasil Eksperimen .....	97
V.2.1 Perbandingan Hasil Eksperimen .....	97
V.3 Analisis Hasil Eksperimen .....	100
V.3.1 CPU Usage .....	101
V.3.2 Memory Usage .....	102
V.3.3 Execution Time .....	107
V.4 Kesimpulan Eksperimen .....	109
Bab VI Penutup .....	110
VI.1 Kesimpulan .....	110
VI.2 Saran .....	111
Daftar Pustaka .....	113
LAMPIRAN .....	116

## DAFTAR GAMBAR

Gambar II.1 Ilustrasi Arsitektur MVP secara umum [6] .....	13
Gambar II.2 Ilustrasi Arsitektur MVP pada android [6].....	13
Gambar II.3 Class Diagram Arsitektur MVP [6].....	14
Gambar II.4 Arsitektur Diagram MVP [9].....	15
Gambar II.5 Ilustrasi Arsitektur MVVM [6] .....	16
Gambar II.6 Ilustrasi Arsitektur MVVM pada android [6].....	17
Gambar II.7 contoh penerapan data binding [9] .....	18
Gambar II.8 Contoh XML yang menerapkan databinding [9].....	19
Gambar II.9 Class Diagram Arsitektur MVVM [6].....	19
Gambar II.10 Arsitektur Diagram MVVM [9] .....	20
Gambar II.11 Contoh aplikasi POS [16].....	21
Gambar III.1 Keterkaitan antar variabel penelitian .....	25
Gambar III.2 Diagram metodologi penelitian.....	26
Gambar III.3 Flowchart tahapan eksperimen.....	39
Gambar III.4 Contoh file hasil pengukuran .....	39
Gambar III.5 Contoh data hasil pengukuran.....	40
Gambar IV.1 Model linear sequential pada SDLC .....	46
Gambar IV.2 Alur proses arsitektur MVP .....	48
Gambar IV.3 Alur proses arsitektur MVVM.....	50
Gambar IV.4 Aplikasi POS Loyverse.....	53
Gambar IV.5 Domain Model Aplikasi Point Of Sales.....	55
Gambar IV.6 <i>Business Process</i> penjualan barang .....	56
Gambar IV.7 <i>Business Process</i> Struk penjualan dan <i>Refund</i> .....	57
Gambar IV.8 <i>Business Process</i> Laporan Penjualan.....	58
Gambar IV.9 Gambaran interaksi antara aktor .....	64
Gambar IV.10 Rancangan model data konseptual aplikasi yang dibangun.....	65
Gambar IV.11 Arsitektur diagram Model View Presenter .....	67
Gambar IV.12 Arsitektur diagram Model View ViewModel .....	67



Gambar IV.13 Rancangan Aplikasi Class Diagram .....	68
Gambar IV.14 <i>Sequence diagram</i> melakukan transaksi penjualan.....	75
Gambar IV.15 <i>Package Diagram</i> aplikasi MVP .....	77
Gambar IV.16 <i>Package Diagram</i> aplikasi MVVM .....	78
Gambar IV.17 Struktur project aplikasi MVP .....	79
Gambar V.1 Proses eksperimen aplikasi .....	90
Gambar V.2 Data mentah hasil eksperimen.....	91
Gambar V.3 Perbandingan <i>CPU Usage</i> aplikasi MVP dan MVVM.....	97
Gambar V.4 Perbandingan <i>Memory Usage</i> aplikasi MVP dan MVVM.....	98
Gambar V.5 Perbandingan <i>Execution Time</i> aplikasi MVP dan MVVM .....	98
Gambar V.6 Perbandingan <i>CPU usage</i> berdasarkan jumlah data .....	99
Gambar V.7 Perbandingan <i>CPU usage</i> berdasarkan jumlah data .....	100
Gambar V.8 Perbandingan <i>Execution Time</i> berdasarkan jumlah data.....	100

## DAFTAR TABEL

Tabel IV.1 Persamaan dan Perbedaan Arsitektur MVP dan MVVM.....	50
Tabel IV.2 perbedaan <i>traditional</i> MVP dan <i>databinding</i> MVVM.....	51
Tabel IV.3 Perbedaan <i>View</i> MVP dengan <i>View</i> MVVM .....	52
Tabel IV.4 fungsional aplikasi POS Loyverse.....	54
Tabel IV.5 Data barang.....	59
Tabel IV.6 Data kategori barang.....	59
Tabel IV.7 Data transaksi .....	60
Tabel IV.8 Data Detail Transaksi .....	60
Tabel IV.9 Data keranjang.....	60
Tabel IV.10 Data inventory .....	61
Tabel IV.11 Data Refund.....	61
Tabel IV.12 Fungsional Requirement.....	62
Tabel IV.13 Struktur data barang.....	65
Tabel IV.14 Struktur data kategori barang.....	65
Tabel IV.15 Struktur data keranjang.....	66
Tabel IV.16 Struktur data transaksi .....	66
Tabel IV.17 Struktur data detail transaksi .....	66
Tabel IV.18 Struktur data refund .....	66
Tabel IV.19 UI-01 Halaman Utama.....	69
Tabel IV.20 UI-02 Daftar Menu Fitur .....	69
Tabel IV.21 UI-03 Popup Jika Memesan Barang.....	70
Tabel IV.22 UI-04 Halaman Setelah Melakukan Pemesanan.....	70
Tabel IV.23 UI-05 Halaman Keranjang Pesanan.....	71
Tabel IV.24 UI-06 Halaman Pembayaran Tunai .....	71
Tabel IV.25 UI-07 Halaman Pembayaran Selesai .....	72
Tabel IV.26 UI-08 Halaman Struk Penjualan.....	72
Tabel IV.27 UI-09 Halaman Detail Struk Penjualan .....	73
Tabel IV.28 UI-010 Halaman Refund .....	73
Tabel IV.29 UI-011 Popup Jika Menekan Checkbox.....	74

Tabel IV.30 UI-012 Halaman Laporan Penjualan .....	74
Tabel IV.31 <i>Sequence diagram</i> melakukan transaksi penjualan .....	75
Tabel IV.32 Perangkat keras yang digunakan .....	76
Tabel IV.33 Perangkat lunak yang digunakan .....	76
Tabel IV.34 Requirement yang berhasil di dimplementasikan.....	76
Tabel IV.35 Penjelasan <i>Package Diagram</i> aplikasi MVP.....	78
Tabel IV.36 Penjelasan <i>Package Diagram</i> aplikasi MVVM.....	79
Tabel IV.37 <i>Library</i> untuk arsitektur MVP .....	80
Tabel IV.38 <i>Source code</i> pada komponen <i>View</i> .....	81
Tabel IV.39 <i>Source code</i> mewakili <i>class BarangPresenter</i> .....	81
Tabel IV.40 <i>Source code</i> yang menghubungkan Presenter dengan <i>repository</i> ....	82
Tabel IV.41 <i>Source code</i> yang menghubungkan <i>repository</i> dengan <i>database</i> ....	82
Tabel IV.42 <i>Source code</i> yang menyimpan <i>query</i> .....	82
Tabel IV.43 <i>Source code</i> pada komponen <i>View</i> .....	84
Tabel IV.44 <i>Source code</i> pada komponen <i>BarangViewModel</i> .....	84
Tabel IV.45 <i>Source code</i> yang menghubungkan <i>ViewModel</i> dengan <i>repository</i> .	84
Tabel IV.46 <i>Source code</i> yang menghubungkan dengan <i>Database</i> .....	85
Tabel IV.47 <i>Source code</i> dibawah adalah mewakili <i>class Database</i> .....	85
Tabel IV.48 <i>Requirement</i> yang telah berhasil di uji .....	86
Tabel V.1 Skenario eksperimen berdasarkan fungsional aplikasi .....	89
Tabel V.2 Hasil pengukuran Aplikasi MVP .....	91
Tabel V.3 Hasil pengukuran Aplikasi MVVM.....	93
Tabel V.4 Skenario eksperimen berdasarkan jumlah data.....	95
Tabel V.5 Hasil pengukuran berdasarkan jumlah data .....	96
Tabel V.6 Nilai rata-rata CPU usage setiap test case.....	101
Tabel V.7 Nilai rata-rata CPU usage dengan jumlah data .....	102
Tabel V.8 Nilai rata-rata memory usage setiap test case .....	103
Tabel V.9 Nilai rata-rata <i>memory usage</i> dengan jumlah data.....	103
Tabel V.10 Perbandingan ukuran APK arsitektur MVP dan MVVM.....	104
Tabel V.11 Perbedaan arsitektur MVP dan MVVM pada <i>package class</i> .....	105
Tabel V.12 Perbedaan Presenter dengan <i>ViewModel</i> .....	106

Tabel V.13 Alokasi memori aplikasi .....	106
Tabel V.14 Nilai rata-rata execution time setiap test case .....	107
Tabel V.15 Nilai rata-rata execution time dengan jumlah data .....	108
Tabel V.16 Penerapan <i>timer</i> untuk <i>exection time</i> .....	109
Tabel V.17 <i>Method</i> untuk menampilkan timer atau <i>execution time</i> .....	109
Tabel 0.1 <i>Use case scenario</i> mencari barang berdasarkan kategori barang .....	1
Tabel 0.2 <i>Use case scenario</i> mencari barang berdasarkan <i>keyword</i> .....	1
Tabel 0.3 <i>Use case scenario</i> memilih barang .....	2
Tabel 0.4 <i>Use case scenario</i> melakukan tambah keranjang .....	3
Tabel 0.5 <i>Use case scenario</i> melakukan pembayaran secara tunai .....	4
Tabel 0.6 <i>Use case scenario</i> melihat histori transaksi penjualan .....	5
Tabel 0.7 <i>Use case scenario</i> melihat detail transkasi penjualan .....	5
Tabel 0.8 <i>Use case scenario</i> melakukan refund / pengembalian dana .....	6
Tabel 0.9 <i>Use case scenario</i> melihat laporan penjualan harian .....	7
Tabel 0.10 <i>Use case scenario</i> melihat laporan penjualan bulanan .....	8
Tabel 0.11 <i>Use case scenario</i> melihat laporan penjualan tahunan .....	9
Tabel IV.49 Pengujian fungsional aplikasi .....	12



## DAFTAR ISTILAH



*Class Activity*

: *Activity* adalah sebuah komponen dalam aplikasi android yang berperan sebagai halaman *user interface* untuk berinteraksi antara pengguna dengan aplikasi



*Class Fragment*

: *Fragment* adalah sebuah komponen yang berperan sama seperti *activity* untuk berinteraksi dengan pengguna akan tetapi *fragment* hanya dapat berjalan di dalam sebuah *activity*.



*CPU usage*

: *CPU usage* adalah sebuah indikator yang memperlihatkan penggunaan CPU dalam sebuah *device*.



*Memory usage*

: *Memory usage* adalah sebuah indikator yang memperlihatkan penggunaan memori dalam sebuah *device*.



*Execution time*

: *Execution Time* adalah sebuah indikator yang memperlihatkan kecepatan waktu eksekusi dalam sebuah program atau aplikasi dalam sebuah



*Databinding*

: *Databinding* adalah proses membangun koneksi *application UI* dan *business logic*.

