

Tugas Besar Pemrograman Pada Perangkat Bergerak

“Calculator”



Disusun oleh:

Lanang Galih S (2016730035)

Frederick (2016730040)

Reynaldi Irfan A (2016730045)

Kelas: A

Fakultas Teknologi Informasi dan Sains

Universitas Katolik Parahyangan

Bandung

2019

PRAKATA

Puji Syukur ke hadirat Tuhan Yang Maha Esa, yang telah memberikan berkatnya sehingga penulis dapat menyelesaikan laporan tugas besar mata kuliah Pemrograman Pada Perangkat Bergerak ini dengan baik.

Dalam prakata ini juga, penulis menyadari bahwa penyusunan laporan ini tidak lepas dari banyak pihak yang telah memberikan bantuan baik moril maupun materiil. Maka bersama ini penulis juga mengucapkan terima kasih yang sebesar-besarnya kepada:

1. Bapak Raymond Chandra Putra, S.T., M.T. selaku dosen praktikum.
2. Rekan-rekan mahasiswa Teknik Informatika UNPAR dan semua pihak yang telah membantu dalam proses penyelesaian laporan ini.

Tentunya dalam laporan ini pasti akan ada banyak hal yang kurang sempurna, maka dari itu kami mengharapkan saran dan kritik yang membangun guna koreksi di masa yang akan datang. Terimakasih atas perhatian dan kerjasamanya.

Bandung, 13 September 2019

Hormat Kami,

Penulis

DAFTAR ISI

PRAKATA	i
DAFTAR ISI	ii
DAFTAR GAMBAR	iv
BAB I PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Ruang Lingkup	1
1.3 Tujuan dan Manfaat	1
1.4 Sistematika Penulisan	1
BAB II LANDASAN TEORI	3
2.1 Sejarah Sistem Operasi Android	3
2.2 Versi Sistem Operasi Android	3
2.3 Kelebihan dan Kekurangan Android	4
2.3.1 Kelebihan Android	4
2.3.2 Kekurangan Android	5
2.4 Android SDK	5
2.5 Android Virtual Device	5
2.6 Java Development Kit	5
2.7 Android Studio	5
BAB III IMPLEMENTASI DAN PEMBAHASAN	7
3.1 Latar Belakang aplikasi	7
3.2 Tampilan Aplikasi	7
3.2.1 Halaman Utama	8

3.2.2	Halaman AddFragment	9
3.2.3	Halaman Result	11
3.2.4	Drawer (Menu)	12
3.3	Cara Menggunakan Aplikasi	13
3.4	Diagram Kelas	13
3.5	Penjelasan berbagai Kelas, Atribut, dan Method	18
3.6	Definisi JSON dan Penggunaannya	28
3.6.1	Definisi JSON	28
3.6.2	Penggunaan JSON	28
BAB IV PENUTUP		29
4.1	Kesimpulan	29
4.2	Saran	29
DAFTAR PUSTAKA		30
LAMPIRAN		31

DAFTAR GAMBAR

GAMBAR 3.2.1 TAMPILAN HALAMAN UTAMA	8
GAMBAR 3.2.2 TAMPILAN HALAMAN ADDFRAGMENT	9
GAMBAR 3.2.3 TAMPILAN HALAMAN ADDFRAGMENT(2).....	9
GAMBAR 3.2.4 TAMPILAN HALAMAN UTAMA(2)	10
GAMBAR 3.2.5 TAMPILAN HALAMAN RESULT	11
GAMBAR 3.2.6 TAMPILAN DRAWER	12
GAMBAR 3.4.1 ILUSTRASI MVVM	14
GAMBAR 3.4.2 KELAS FRAGMENTLISTENER	15
GAMBAR 3.4.3 KELAS IMAINACTIVITY.....	16
GAMBAR 3.4.4 KELAS NUMOP	16
GAMBAR 3.4.5 KELAS ADAPTER	17
GAMBAR 3.4.6 KELAS MAIN ACTIVITY	17
GAMBAR 3.4.7 KELAS SAVEDISPLAY	18
GAMBAR 3.4.8 KELAS VIEW HOLDER	18
GAMBAR 3.4.9 KELAS ADD FRAGMENT	19
GAMBAR 3.4.10 KELAS HALAMAN UTAMA	20
GAMBAR 3.4.11 KELAS LEFTFRAGMENT	20
GAMBAR 3.4.12 KELAS RESULT DIALOG FRAGMENT	20
GAMBAR 3.4.13 KELAS DIAGRAM KESELURUHAN	21
GAMBAR L.1 ILUSTRASI PENGGAMBARAN ADAPTER	31

BAB I

PENDAHULUAN

1.1 Latar Belakang

Kalkulator telah menjadi sumber berbagai penyelesaian masalah kehidupan sehari-hari kita. Berbagai jenis kalkulator juga sudah digunakan oleh berbagai kalangan masyarakat. Ada kalkulator *scientific*, kalkulator biasa, bahkan kalkulator digital.

Pada tugas ini, kami mengembangkan sebuah aplikasi kalkulator yang bisa dipakai pada smartphone yang berbasis Android. Mengapa Android? Selain Android merupakan sistem operasi yang banyak dipakai masyarakat umum, Android juga memudahkan kami dalam pengembangannya dibanding sistem operasi iOS.

1.2 Ruang Lingkup

Ruang lingkup dari perancangan kalkulator ini adalah sebagai berikut:

1. Kalkulator ditujukan untuk Sistem Operasi Android minimal versi 5.0 Jelly Bean dengan memakai IDE Android Studio dan menggunakan Java sebagai bahasa pemrograman.
2. Kalkulator ini hanya digunakan dalam lingkup perkuliahan, tidak dikomersilkan

1.3 Tujuan dan Manfaat

Tujuan dari aplikasi ini adalah mengembangkan aplikasi kalkulator pada sistem operasi Android, dan untuk memenuhi salah satu tugas pada mata kuliah Pemrograman Pada Perangkat Bergerak. Sedangkan manfaat yang dapat kami peroleh adalah sebagai berikut:

1. Mengembangkan aplikasi kalkulator sebagai *tools* yang bisa dipakai dengan mudah.
2. Melatih kemampuan kami untuk lebih memahami pengembangan aplikasi pada Sistem Operasi Android.

1.4 Sistematika Penulisan

BAB I : PENDAHULUAN

Bab ini mencakup tentang latar belakang, ruang lingkup, tujuan dan manfaat, dan sistematika penulisan.

BAB II : LANDASAN TEORI

Pada bab ini berisi uraian penjelasan tentang pembuatan calculator. Meliputi sejarah sistem operasi Android, versi sistem operasi Android, kelebihan dan kekurangan Android, Android SDK, Android Virtual Device, Java Development Kit, dan Android Studio.

BAB III : IMPLEMENTASI DAN PEMBAHASAN

Pada bab ini diuraikan tentang tampilan halaman awal, tampilan Fragment Halaman Input calculator, tampilan, diagram kelas, penjelasan kelas-kelas, penjelasan atribut-atribut, penjelasan method-method

BAB IV : PENUTUP

Pada bab ini terdapat kesimpulan dan saran mengenai aplikasi yang telah dibuat.

BAB II

LANDASAN TEORI

Pada bab ini akan dibahas landasan teori yang berhubungan dengan permasalahan dan penyelesaian yang akan dibahas dalam tugas ini. Teori yang akan dibahas meliputi sejarah sistem operasi Android, versi sistem operasi Android, kelebihan dan kekurangan Android, Android SDK, Android Virtual Device, Java Development Kit, dan Android Studio.

2.1 Sejarah Sistem Operasi Android

***Android** adalah sistem operasi berbasis linux yang dirancang untuk perangkat layar sentuh seperti smartphone dan pc tablet.*

Android sendiri awalnya dikembangkan oleh Android, Inc., yang semua dananya ditanggung oleh Google. Tujuan pengembangan android awalnya sebenarnya untuk kamera digital, namun dikarenakan pasar untuk perangkat tersebut tidak terlalu besar maka pengembangan android dialihkan untuk smartphone.

Cara itu dilakukan untuk menyaingi sistem operasi Symbian dan Windows Mobile yang saat itu sedang naik daun. Dengan hadirnya Android, diharapkan pasar sistem operasi smartphone akan lebih variatif. Kemudian pada tanggal 17 Agustus 2005, Android Inc. resmi diakuisisi oleh Google dan menjadikannya anak perusahaan yang sepenuhnya dimiliki oleh Google.

Pada tahun 2007 sistem operasi android dirilis secara resmi oleh Google bersamaan dengan didirikannya Open Handset Alliance (OHA). OHA didirikan dengan tujuan untuk mengembangkan standar terbuka bagi perangkat seluler.

2.2 Versi Sistem Operasi Android

Untuk bagian ini, kami membahas Sistem Operasi Android dimulai dari versi yang banyak dipakai, dimulai dari Jelly Bean.

1. Jelly Bean (Versi 4.1 - 4.3)

Pembaruan ini berupa peningkatan terhadap kemampuan on screen keyboard yang lebih cepat dan responsif, serta adanya beberapa penambahan fitur lainnya yang lebih menarik dari versi sebelumnya.

2. KitKat (Versi 4.4)

KitKat merupakan versi Android yang memiliki keunggulan dari sisi resolusi yang dinilai mampu meningkatkan kualitas permainan pada aplikasi 3D.

3. Lollipop (Versi 5.0)

Versi Lollipop dinilai sebagai pembaruan yang difokuskan pada desain tampilan dan performa, agar lebih nyaman digunakan. Terutama kemampuan untuk aplikasi dengan kualitas tinggi dan aplikasi-aplikasi yang memerlukan resolusi tinggi.

4. Marshmallow (Versi 6.0)

Pada versi ini terdapat penambahan fitur berupa *Native Finger Scanner Support* dan *Individual app permission*. *Native Finger Scanner Support* adalah fitur yang digunakan untuk keamanan sidik jari pada ponsel yang sudah mengadopsi fitur ini. *App Permission* adalah sebuah fitur yang akan memberikan akses khusus pada user untuk menentukan apakah aplikasi yang akan digunakan diberikan izin oleh penggunaannya.

5. Nougat (Versi 7.0)

Pada versi ini terdapat banyak perubahan terutama pada performa yang lebih baik berkat adanya fitur JIT (just In Time) untuk meningkatkan performa perangkat runtime dan mengurangi kapasitas penyimpanan yang diperlukan aplikasi. Versi ini juga memiliki Mode VR. Android juga menambahkan API Vulkan yang membuat tampilan grafis 3D pada smartphone semakin baik dan tampak hidup dengan grafis yang lebih tajam.

6. Oreo (Versi 8.0)

Pada versi ini menjadikan perangkat berjalan lebih cepat, lebih halus, dan lebih menghemat daya baterai dari sebelumnya. Keamanan ditingkatkan berkat adanya Google Play Protect, layanan yang secara berkala memeriksa risiko keamanan perangkat dan aplikasi serta akan memberitahukan jika ada risiko ditemukan.

2.3 Kelebihan dan Kekurangan Android

2.3.1 Kelebihan Android:

1. Open Source

Kelebihan pertama yang dimiliki oleh sistem operasi ini adalah sifatnya yang open source, terbuka untuk dikembangkan. Artinya, android mudah dikembangkan oleh para developer, dan mudah dimodifikasi sesuai kebutuhan.

2. Mudah dimodifikasi dan dikustomisasi

Kustomisasi dan modifikasi itu bisa dilakukan dengan tujuan untuk mempercepat fungsi dan menyesuaikan dengan kebutuhan pengguna. Adapun detail kustomisasi [aplikasi android](#) yang sering dilakukan adalah lock screen, font, aplikasi, dan sebagainya.

3. Harga beragam

Harga smartphone android yang variatif dan terjangkau, yaitu mulai dari harga sekitar Rp500.000,00

4. Fleksibel

Android dapat diterapkan dalam berbagai jenis hardware dengan berbagai jenis spesifikasi yang akan menguntungkan pengguna. Developer bisa menerapkannya pada hardware berjenis low end, medium, hingga yang high quality sekaligus. Semua level kualitas hardware bisa menggunakan sistem operasi android yang kaya versi.

5. Banyak dukungan aplikasi

Hal lain yang menarik dari sistem operasi android adalah beragamnya [aplikasi](#) yang disediakan. Inilah yang menjadi alasan mengapa smartphone berbasis android cukup diminati. Ini juga berhubungan dengan sifat dari OS android sendiri yang open source sehingga mudah dikembangkan, dikustomisasi dan dimodifikasi sesuai dengan keinginan.

6. Mudah dipahami

Cara kerjanya cukup mudah dengan petunjuk yang lengkap. Ini akan memberikan kemudahan bagi user meski baru kali pertama menggunakannya.

7. Cepat dan responsif

Jika dilihat dari analisis statistiknya, sistem operasi android memiliki respon yang cepat dan responsif dibandingkan dengan sistem operasi yang lain. Meskipun, secara fakta ini sangat bergantung pada spesifikasi dan kualitas hardwarenya, apakah low end, medium, atau high end.

2.3.2 Kekurangan Android

1. Sistem bekerja dengan cukup berat

Untuk smartphone dengan harga murah, spesifikasi rendah, dan memory yang kecil, sudah pasti sistem ini akan berjalan lambat. Salah satu solusinya adalah membeli smartphone dengan mempertimbangkan besar kecilnya kapasitas RAM dan ROM yang dimiliki sebagai prioritas.

2. Kurang cocok dengan hardware yang buruk

Jika sistem operasi android dikombinasikan dengan kualitas hardware yang kurang bagus, respon yang diberikan akan lebih lambat karena ini berhubungan dengan ROM, RAM, dan kapasitas prosesor. Untuk itu, jika pertimbangan yang digunakan adalah efektivitas waktu, smartphone medium atau high class dengan sistem operasi android yang mudah dioperasikan dapat menjadi pilihan.

2.4 Android SDK

Android SDK (*software development kit*) adalah satu set alat pengembangan yang digunakan untuk mengembangkan aplikasi untuk platform [Android](#). Android SDK multiple library yang diperlukan, debugger, emulator, dokumentasi yang relevan untuk antarmuka program aplikasi Android (API), kode sumber sampel, tutorial untuk OS Android.

2.5 Android Virtual device

Android Virtual Device (AVD) yaitu semacam emulator untuk menjalankan virtual Android. Jadi tanpa menggunakan / mempunyai android phone pun, pengembang bisa merasakan apa itu android. dan juga bisa membuat program di android. tetapi untuk yang ingin berkecimpung di dunia android coding sebaiknya minimal punya phone android karena jika menggunakan AVD maka akan memakan memori ram, sehingga akan berjalan agak berat.

2.6 Java Development Kit

Java Development Kit (JDK) harus ada, karena Android merupakan aplikasi yang dibangun dengan menggunakan bahasa pemrograman Java. Untuk mendapatkan installer JDK bisa mengunduh / download langsung di situs resminya.

2.7 Android Studio

[Android Studio](#) adalah *integrated development environment* (IDE) resmi untuk Android yang diluncurkan oleh Google pada tahun 2013. Tools ini memiliki Virtual Device Manager yang memungkinkan untuk melakukan pengujian secara langsung tanpa perlu menggunakan perangkatnya langsung. Kita bisa *men-debug* aplikasi yang dibuat secara langsung menggunakan Android Device Manager. Tool ini tersedia untuk digunakan di Windows, Linux dan Mac OS X.

BAB III

IMPLEMENTASI DAN PEMBAHASAN

Pada bab ini diuraikan tentang tampilan Fragment **HalamanUtama**, tampilan Fragment **AddFragment**, tampilan Fragment **ResultDialogFragment**, tampilan Drawer (Menu), diagram kelas, penjelasan kelas-kelas, penjelasan atribut-atribut, dan penjelasan method-method.

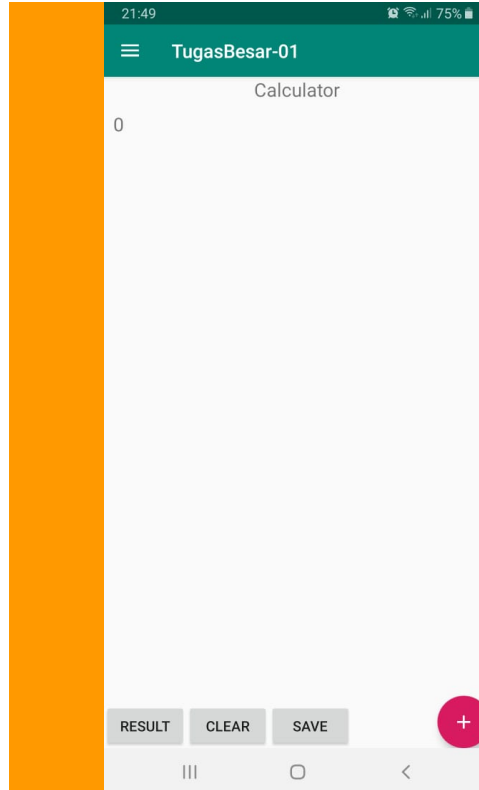
3.1 Latar Belakang Aplikasi

Aplikasi yang dibuat merupakan replika sederhana kalkulator yang biasa digunakan sehari-hari. Kalkulator yang dikembangkan dapat melakukan operasi tambah, kurang, kali, dan bagi. Namun datatype hasil yang ditampilkan merupakan Integer, hal ini menyebabkan tidak adanya angka desimal pada input maupun output dari hasil pengoperasian. Kalkulator yang dikembangkan juga dapat menampilkan input yang dimasukan user kedalam list dan menampilkan result dari setiap input yang diberikan. Setiap input yang dimasukan kedalam list dapat dihapus dan mempengaruhi hasil perhitungan (result). Kalkulator ini juga memiliki fitur-fitur lainnya yang memiliki fungsinya masing-masing.

3.2 Tampilan Aplikasi

Aplikasi kami terdiri dari tiga buah fragment yaitu Fragment **HalamanUtama** sebagai Fragment untuk menampilkan list input beserta hasil sementara dari operasi matematika dan juga sebagai halaman utama aplikasi, Fragment **AddFragment** untuk user memilih operator dan input nilai yang akan dimasukan kedalam list perhitungan, dan Fragment result untuk menampilkan hasil perhitungan list yang berada di **HalamanUtama**.

3.2.1 Halaman Utama

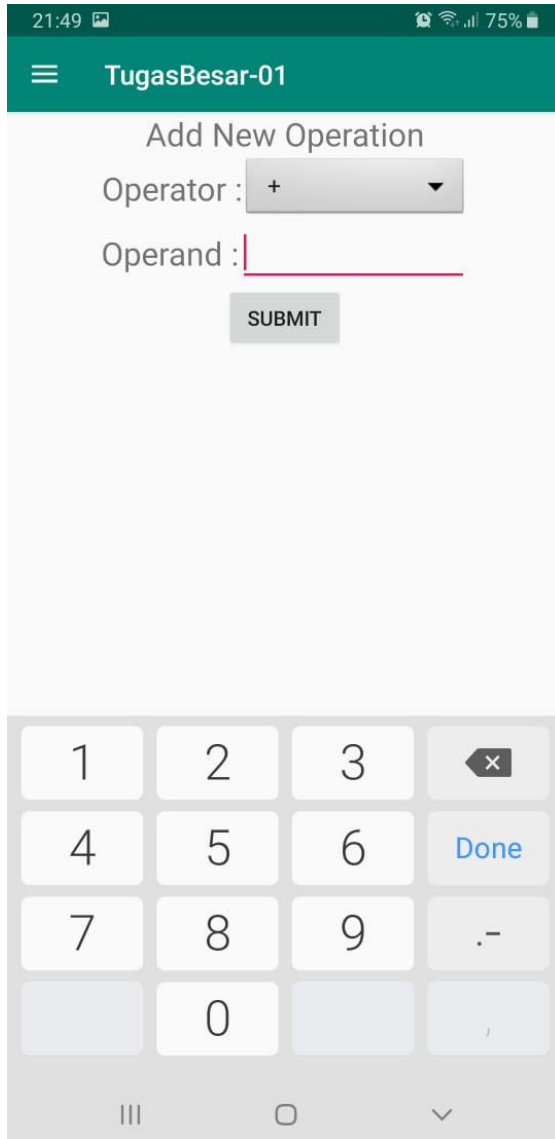


Gambar 3.2.1 Halaman Utama

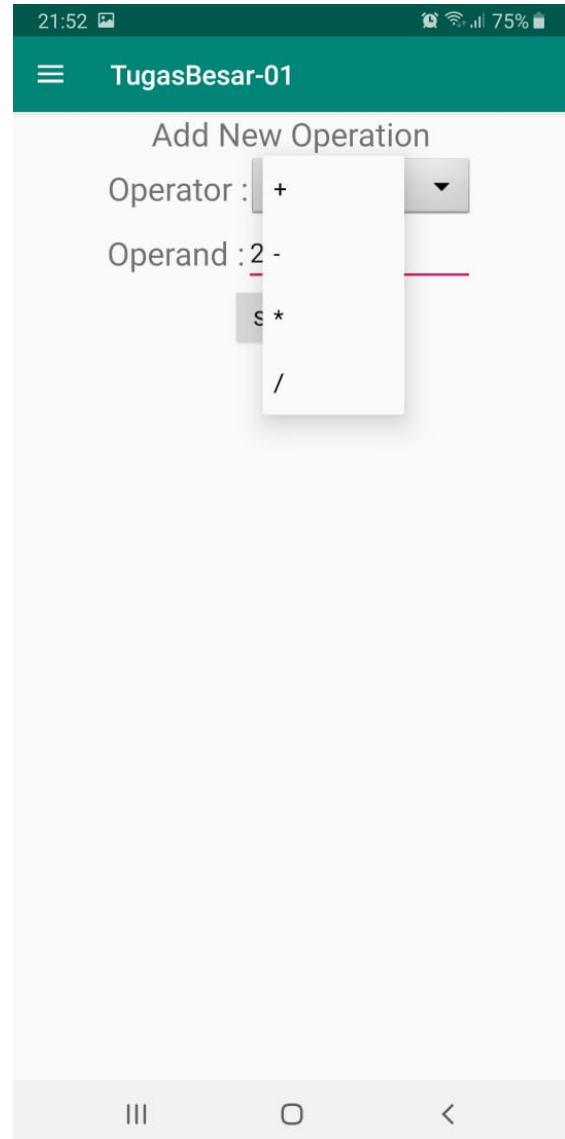
Pada Fragment ini terdapat View:

1. TextView “Calculator” sebagai judul aplikasi
2. TextView “0” sebagai hasil sementara
3. Button “RESULT” untuk menampilkan hasil operasi
4. Button “CLEAR” untuk membersihkan seluruh list dan mereset nilai current result
5. Button “SAVE” untuk menyimpan semua list sehingga ketika aplikasi ditutup dan dibuka lagi akan menunjukkan kondisi terakhir
6. Floating Button “+” untuk membuka fragment AddFragment
7. Toolbar Tiga garis untuk membuka menu drawer

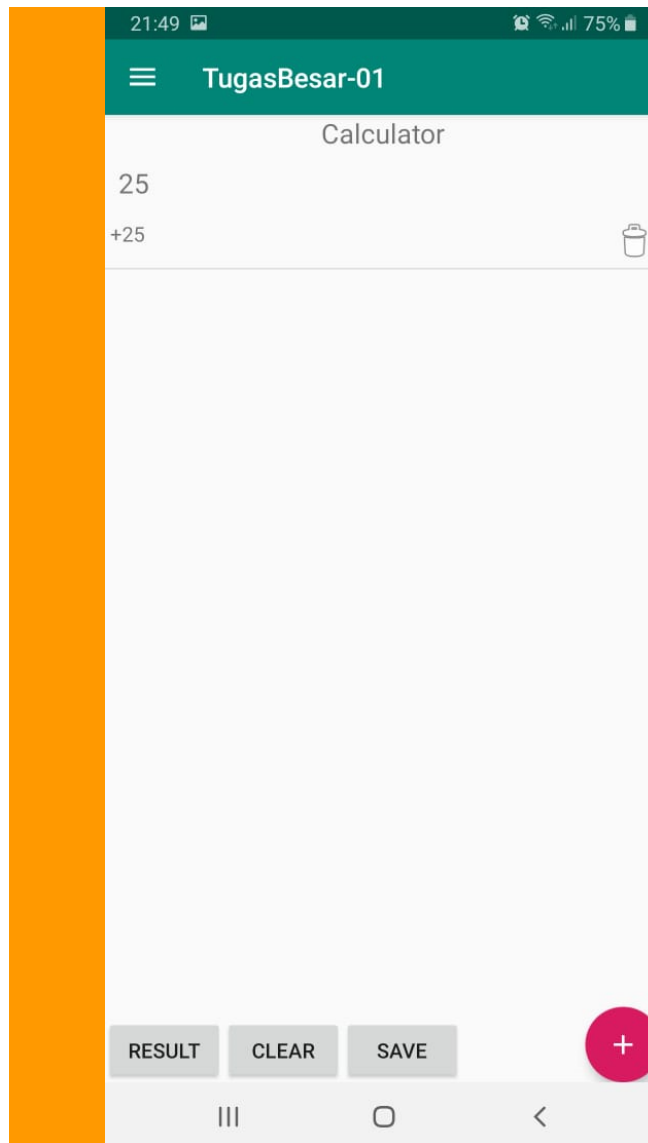
3.2.2 Halaman AddFragment



Gambar 3.2.2 Halaman AddFragment



Gambar 3.2.3 Halaman AddFragment dengan spinner dan input



Gambar 3.2.4 Halaman Utama dengan ListView

Tampilan pada Gambar 3.2.2 merupakan tampilan saat user telah menekan floating tombol “+” pada halaman utama. Pada tampilan ini terdapat View:

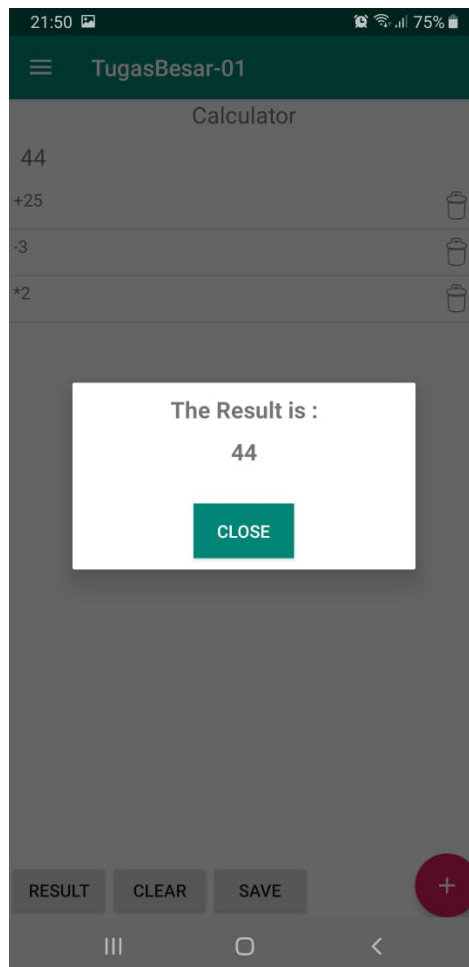
1. TextView “Add New Operation” sebagai judul fragment
2. TextView “Operator :” sebagai keterangan untuk spinner “+”
3. Spinner “+” untuk menampilkan dropdown operator yang akan digunakan untuk melakukan operasi matematika
4. TextView “Operand :” sebagai keterangan untuk EditText yang berada di sebelahnya
5. EditText untuk user memasukan input operand (angka) yang akan digunakan dalam operasi matematika

6. Button “Submit” untuk menambahkan objek yang memiliki operator dan operand yang telah dipilih oleh user kedalam list untuk ditampilkan pada listView pada HalamanUtama dan fragment berpindah ke halaman utama.

Penjelasan fragment ini:

Ketika user telah menentukan operator dan operand yang dipilih untuk melakukan operasi matematika dan telah menekan tombol submit maka ListView pada halaman utama akan diperbarui, seperti pada Gambar 3.2.4

3.2.3 Halaman Result

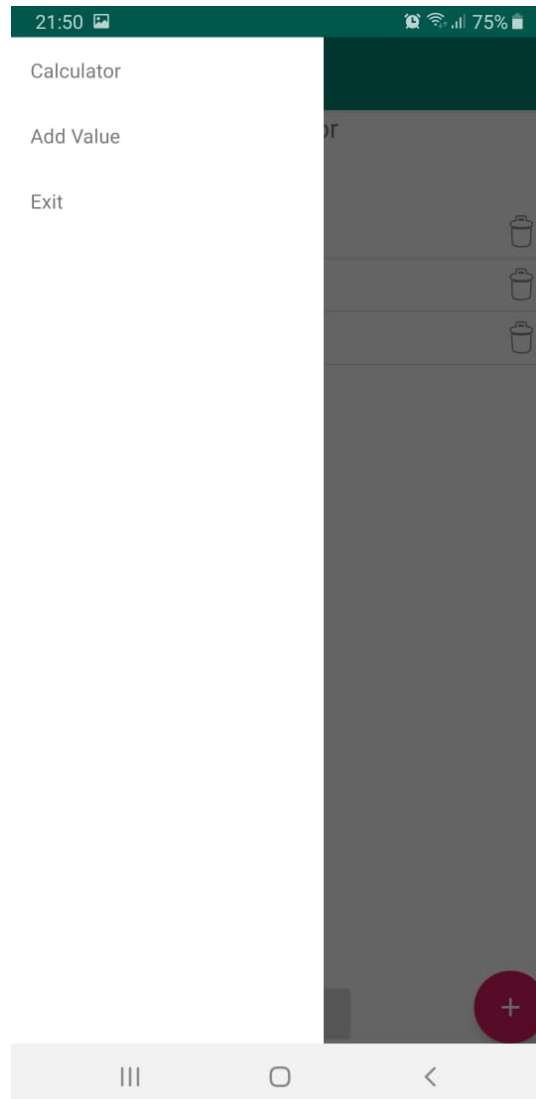


Gambar 3.2.5 Halaman Result

Tampilan pada Gambar 3.2.5 merupakan tampilan saat user telah menekan tombol “RESULT” pada halaman utama. Tampilan ini bertujuan untuk menampilkan dialog yang berisikan perhitungan seluruh angka di dalam listview. Pada tampilan ini terdapat View:

1. TextView “The Result is :” sebagai keterangan hasil dari pengoperasian yang berada di dalam ListView
2. TextView “0” untuk menampilkan hasil perhitungan seluruh angka di dalam ListView
3. Button “CLOSE” untuk menutup dialog yang sedang ditampilkan

3.2.4 Drawer (Menu)



Gambar 3.2.6 Drawer

Tampilan pada Gambar 3.2.6 merupakan tampilan saat user telah menekan toolbar garis tiga pada halaman apapun. Pada tampilan ini terdapat View:

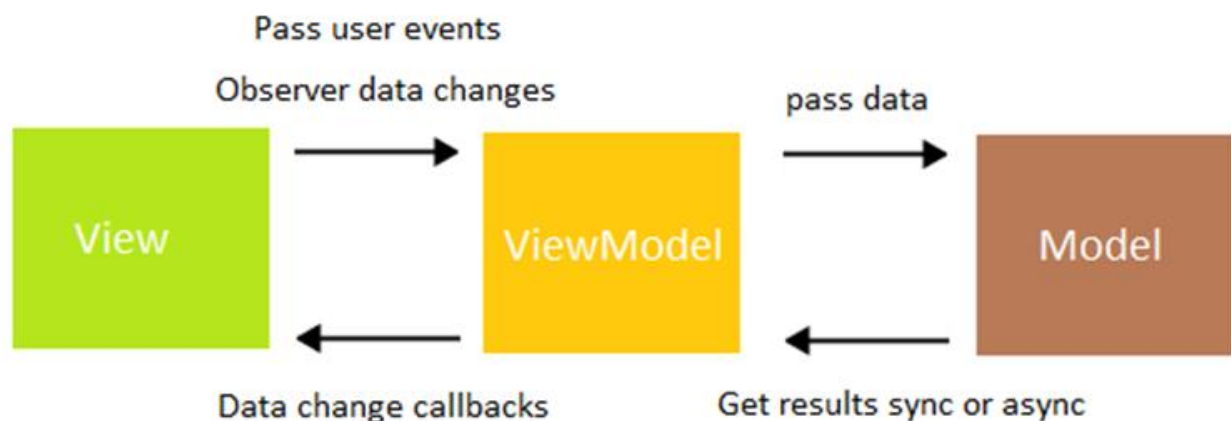
1. TextView “Calclator” untuk berpindah ke fragment halaman utama
2. TextView “Add Value” untuk berpindah ke fragment halaman AddFragment
3. TextView “Exit” untuk keluar dari aplikasi

3.3 Cara Menggunakan Aplikasi

1. Buka aplikasi
2. Tekan *floating button* '+' untuk memulai operasi matematika
3. Pilih operator yang tersedia dan masukan angka yang ingin dioperasikan
4. Tekan tombol "submit" untuk memasukan angka dan operator yang telah dipilih ke dalam list operasi dan ditampilkan di halaman utama
5. Tekan tombol "clear" apabila ingin membersihkan semua list dan mengembalikan result menjadi default value (0)
6. Tekan tombol "save" apabila ingin menyimpan data history dari angka yg telah kita input
7. Swipe layar dari kiri ke kanan atau tekan toolbar (tiga garis pada header) untuk menampilkan drawer
8. Pilih "Calculator" untuk berpindah ke halaman utama
9. Pilih "Add Value" untuk menambah atau melanjutkan operasi matematika
10. Pilih "Exit" untuk keluar dari aplikasi

3.4 Diagram Kelas

Aplikasi kami menggunakan jenis arsitektur desain kelas MVVM (Model View View-Model), dengan View yang akan menangani *user interaction*, ViewModel akan melakukan *passing* input yang diberikan oleh user pada View ke Model maupun output yang diberikan Model untuk ditampilkan ke user di View, dan Model sebagai tempat pengolahan seluruh data dan interaksi yang diberikan dari View



Gambar 3.4.1 Ilustrasi MVVM

1. Kelas yang berfungsi sebagai Model
 - a. FragmentListener.java



Base	Stereotype	Attribute	Operation	Generalization	Dependency
Name	Return Value	Type Modifier	Visibility		
changePage	void				public
changeValue	void				public
closeApp	void				public
addList	void				public
clearList	void				public
deleteListPos	void				public
saveList	void				public

Gambar 3.4.2 FragmentListener.java

- b. IMainActivity.java



Base	Stereotype	Attribute	Operation	Generalization	Dependency
Name	Return Value	Type Modifier	Visibility		
updateList	void				public
resetAddForm	void				public

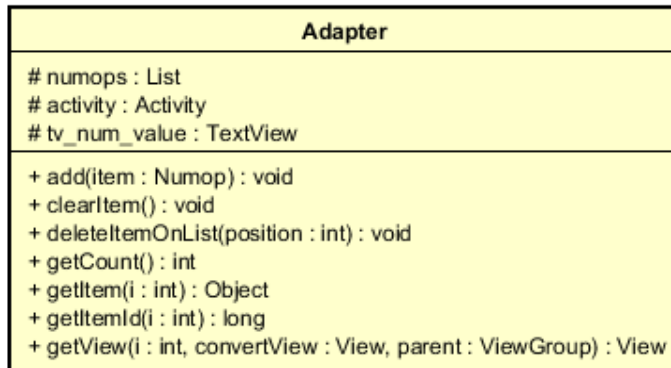
Gambar 3.4.3 IMainActivity.java

- c. Numop.java

Numop
operator : String # value : int
+ getOperator() : String + setOperator(operator : String) : void + getValue() : int + setValue(value : int) : void

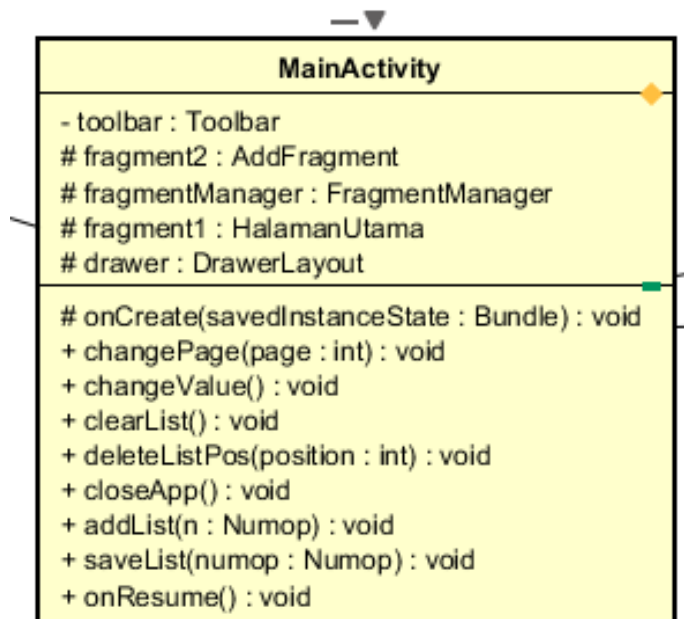
Gambar 3.4.4 Numop.java

2. Kelas yang berfungsi sebagai View-Model
 - a. Adapter.java



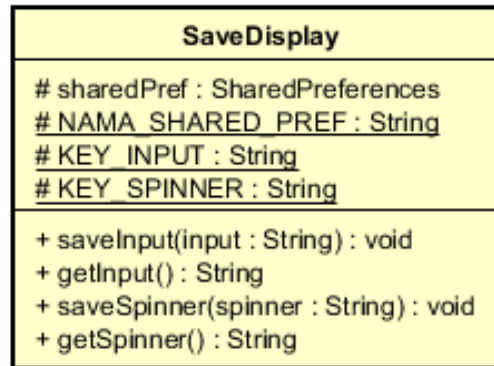
Gambar 3.4.5 Adapter.java

b. MainActivity.java



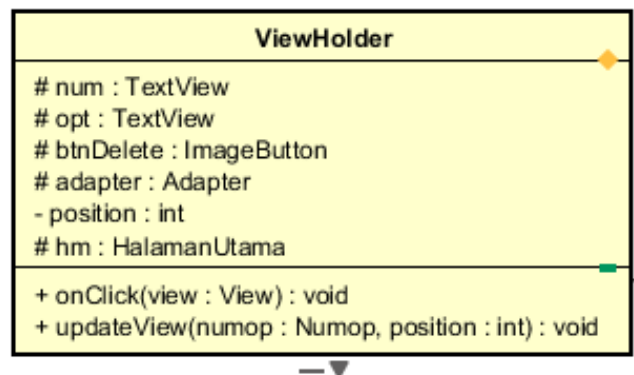
Gambar 3.4.6 MainActivity.java

c. SaveDisplay.java



Gambar 3.4.7 SaveDisplay.java

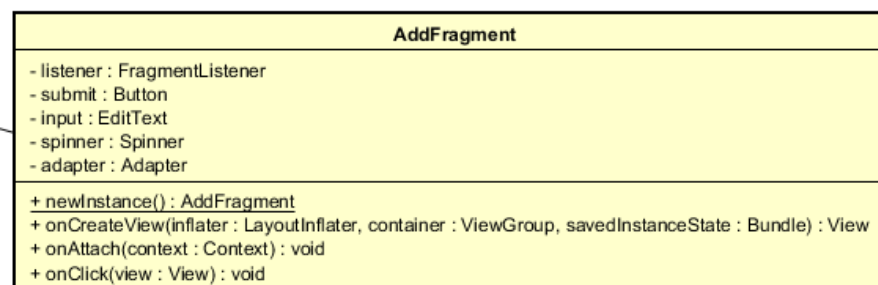
d. ViewHolder.java



Gambar 3.4.8 Ilustrasi kelas ViewHolder.java

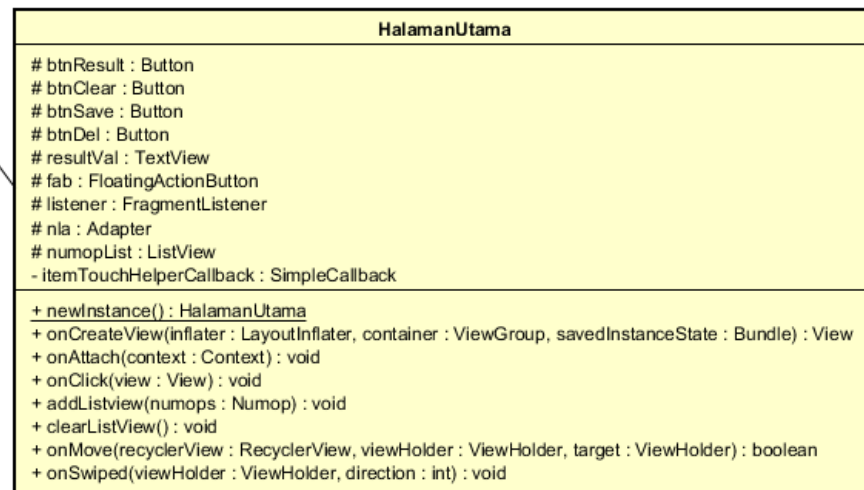
3. Kelas yang berfungsi sebagai View

a. AddFragment.java



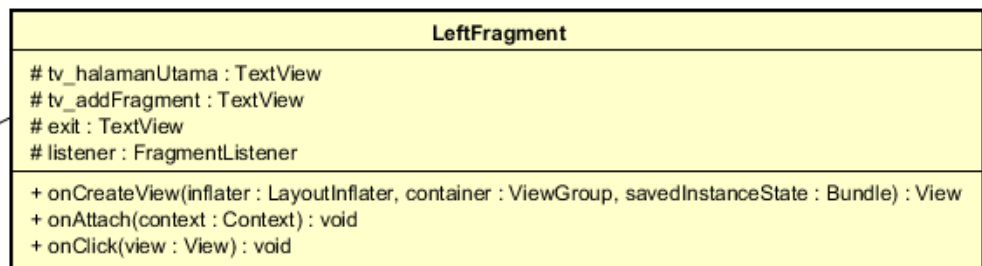
Gambar 3.4.9 AddFragment.java

b. HalamanUtama.java



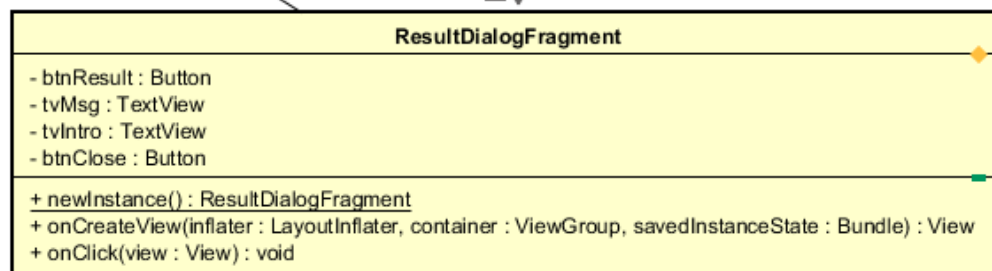
Gambar 3.4.10 HalamanUtama.java

c. LeftFragment.java



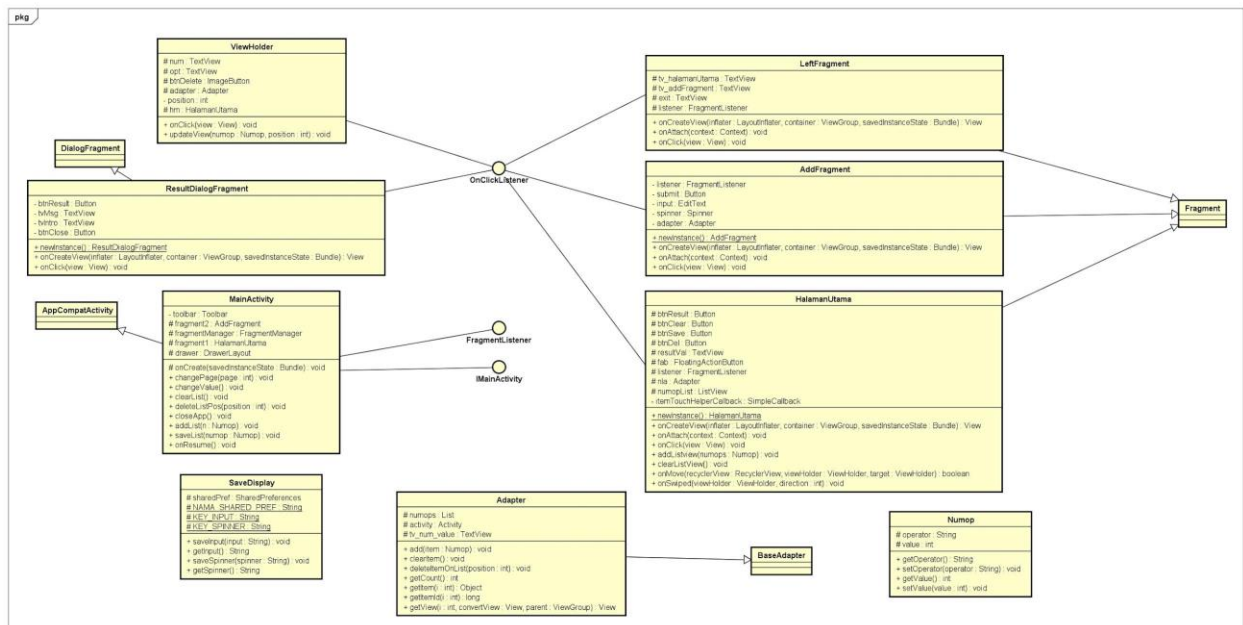
Gambar 3.4.12 LeftFragment.java

d. ResultDialogFragment.java



Gambar 3.4.13 ResultDialogFragment.java

4. Diagram Kelas keseluruhan proyek



Gambar 3.4.12 Kelas Diagram keseluruhan

3.5 Penjelasan berbagai Kelas, Atribut, dan Method

Berikut merupakan kelas-kelas yang terdapat pada aplikasi kami beserta dengan atribut dan methodnya:

Main Activity	
Kelas Main Activity berfungsi untuk menjalankan segala activity dan menyimpan berbagai macam bagian – bagian model aplikasi yang penting	
Atribut	Penjelasan
fragment2	Atribut untuk mengambil atribut dan method di fragment addFragment
fragmentManager	Atribut yang digunakan untuk memindahkan fragment(page)

fragment1	Atribut untuk mengambil atribut dan method di fragment HalamanUtama
drawer	Atribut untuk memunculkan button menu
toolbar	Atribut untuk membuat button menu
Method	Penjelasan
onCreate	Untuk menginiasialisasi semua yang berada di activity
changePage	Method void implement dari dari kelas fragment listener yang berguna untuk mengganti halaman
changeValue	Method void implement dari kelas FragmentListener yang berguna untuk membersihkan ListView
clearList	Method void implement dari kelas FragmentListener yang berguna untuk membersihkan ListView
deleteListPo	Method void implement dari kelas FragmentListener yang berguna untuk menghapus satu elemen dari ListView
closeApp	Method void implement dari kelas FragmentListener yang berguna untuk menutup aplikasi
addList	Method void implement dari kelas FragmentListener yang berguna untuk menambah ListView
saveList	Method void implement dari kelas FragmentListener yang berguna untuk menyimpan isi ListView

onResume	Method void implement dari kelas FragmentListener yang berguna untuk mengembalikan hasil dari method saveList
HalamanUtama	
Kelas HalamanUtama berfungsi sebagai tampilan pertama aplikasi tersebut yang dapat menayangkan result dan list view nya	
Atribut	Penjelasan
btnResult	Atribut yang digunakan untuk menginisialisasikan button result
btnClear	Atribut yang digunakan untuk menginisialisasikan button clear
btnSave	Atribut yang digunakan untuk menginisialisasikan button save
btnDel	Atribut yang digunakan untuk menginisialisasikan button delete
resultVal	Atribut yang digunakan untuk menginisialisasikan TextView resultval
fab	Atribut yang digunakan untuk menginisialisasikan floating action button
listener	Atribut ini digunakan untuk mengambil semua method pada kelas FragmentListener
nla	Atribut ini digunakan untuk mengambil semua method pada kelas Adapter

numopList	Atribut yang digunakan untuk menginisialisasikan ListView resultval
itemTouchHelperCallback	Atribut yang digunakan untuk menginisialisasikan sensor swipe
Method	Penjelasan
newInstance	Method ini digunakan untuk memperbaharui fragment
onCreateView	Method ini digunakan untuk menginisialisasikan atribut dalam fragment ini
onAttach	Method ini meng-extend dari kelas onAttach untuk mengganti page
onClick	Method ini digunakan untuk memberi tugas pada button setelah di click
addListView	Method ini digunakan untuk menambah ListView
clearListView	Method ini digunakan untuk membersihkan ListView
onMove	Method ini digunakan untuk gerak yang dideteksi
onSwiped	Method ini digunakan untuk menghilangkan element dari ListView dengan cara melakukan swipe element tersebut
AddFragment	

Kelas AddFragment berfungsi sebagai untuk menambahkan ListView dengan mengisi dropdown dan edit text yang disediakan

Atribut	Penjelasan
listener	Atribut ini digunakan untuk mengambil semua method pada kelas Fragment listener
submit	Atribut yang digunakan untuk menginisialisasikan button submit
input	Atribut yang digunakan untuk menginisialisasikan EditText input
spinner	Atribut yang digunakan untuk menginisialisasikan dropdown
adapter	Atribut ini digunakan untuk mengambil semua method pada kelas Adapter
Method	Penjelasan
newInstance	Method ini digunakan untuk memperbaharui fragment
onCreateView	Method ini digunakan untuk menginisialisasikan atribut dalam fragment ini
onAttach	Method ini melakukan extend dari kelas onAttach untuk mengganti page
onClick	Method ini digunakan untuk memberi tugas pada button setelah di click
LeftFragment	

Kelas LeftFragment berfungsi sebagai kelas yang menampilkan fragment apabila aplikasi di swipe dari kiri ke kanan

Atribut	Penjelasan
tv_halamanUtama	Atribut yang digunakan untuk menjalankan EditText
tv_addFragment	Atribut yang digunakan untuk menjalankan EditText
exit	Atribut yang digunakan untuk menjalankan EditText
listener	Atribut yang digunakan untuk mengambil semua method kelas listener
Method	Penjelasan
onCreateView	Method ini digunakan untuk menginisialisasikan atribut dalam fragment ini
onAttach	Method ini melakukan extend dari kelas onAttach untuk mengganti page
onClick	Method ini digunakan untuk memberi tugas pada button setelah diklik
ViewHolder	
Kelas ViewHolder berfungsi sebagai	
Atribut	Penjelasan

num	Atribut ini digunakan untuk menampung nilai operand yang dimiliki object Numop yang akan dimasukan ke dalam ListView
opt	Atribut ini digunakan untuk menampung nilai operator yang dimiliki object Numop yang akan dimasukan ke dalam ListView
btnDelete	Atribut ini digunakan untuk memanggil button delete yang berada pada layout numop_list_view untuk menghapus current position list
adapter	Atribut ini digunakan untuk menginisialisasi kelas Adapter
position	Atribut ini digunakan untuk mengetahui posisi object pada ListView
hm	Atribut ini digunakan untuk menginisialisasi kelas HalamanUtama
Method	Penjelasan
onClick	Method ini digunakan untuk memberi tugas pada button setelah di click
updateView	Method ini digunakan untuk memperbarui ListView jika terjadi perubahan, seperti delete salah satu list yang berada di listView
ResultDialogFragment	
Kelas ResultDialogFragment berfungsi untuk menampilkan result dari method changeValue()	

Atribut	Penjelasan
btnResult	Atribut ini digunakan untuk menginisialisasikan buton result
tvMsg	Atribut ini digunakan untuk menginisialisasikan TextView tvmsg
tvIntro	Atribut ini digunakan untuk menginisialisasikan TextView tvIntro
btnClose	Atribut ini digunakan untuk menginisialisasikan button close
Method	Penjelasan
newInstance	Method ini digunakan untuk memperbaharui fragment
onCreateView	Method ini digunakan untuk menginisialisasikan atribut dalam fragment ini
onClick	Method ini digunakan untuk memberi tugas pada button setelah diklik
Adapter	
Kelas Adapter ini berfungsi untuk mengisi aktivitas ListView	
Atribut	Penjelasan
numops	Atribut ini digunakan untuk menginisialisasikan ArrayList yang berisi numop

activity	Atribut ini digunakan untuk mengambil method dan atribut dari kelas activity
tv_num_value	Atribut ini digunakan untuk menginisialisasikan TextView tvnumvalue
Method	Penjelasan
add	Method untuk digunakan untuk mengisi ArrayList
clearItem	Method untuk digunakan untuk membersihkan semua element ArrayList
deleteItemOnList	Method untuk digunakan untuk membersihkan hanya 1 element ArrayList
getCount	Method untuk digunakan untuk menghitung isi dari ArrayList
getItem	Method untuk digunakan untuk mengambil item dalam ArrayList
getItemId	Method untuk digunakan untuk mendapatkan posisi item dalam ArrayList
getView	Method untuk digunakan untuk inisialisasi dan menjalankan banyak fungsi
Numop	
Kelas Numop ini berfungsi sebagai untuk menginisialisasikan isi dari ListView	
Atribut	Penjelasan

operator	Atribut ini untuk menginisialisasikan dari String operator
value	Atribut ini untuk menginisialisasikan dari int value
Method	Penjelasan
getOperator	Method ini digunakan untuk mengambil nilai dari operator
setOperator	Method ini digunakan untuk melakukan set nilai dari operator
getValue	Method ini digunakan untuk mengambil nilai dari value
setValue	Method ini digunakan untuk melakukan set nilai dari value
SaveDisplay	
Kelas SaveDisplay ini berfungsi sebagai menyimpan input edit Text dan Spinner	
Atribut	Penjelasan
sharedPref	Atribut untuk mengambil isi dari kelas shared preferences
NAMA_SHARED_PREF	Atribut static yang untuk melakukan set nilai NAMA_SHARED_PREF
KEY_INPUT	Atribut static yang untuk melakukan set nilai KEY_INPUT
KEY_SPINNER	Atribut static yang untuk melakukan set nilai KEY_SPINNER
Method	Penjelasan

saveInput	Method ini digunakan menyimpan nilai input
getInput	Method ini digunakan untuk <i>load</i> nilai dari method saveInput
saveSpinner	Method ini digunakan menyimpan nilai dropdown
getSpinner	Method ini digunakan <i>load</i> nilai dari method saveSpinner

3.6 Definisi JSON dan Penggunaannya

3.6.1 Definisi JSON

JSON (JavaScript Object Notation) adalah format pertukaran data yang ringan, mudah dibaca dan ditulis oleh manusia, serta mudah diterjemahkan dan dibuat (generate) oleh komputer. Format ini dibuat berdasarkan bagian dari Bahasa Pemrograman JavaScript, Standar ECMA-262 Edisi ke-3 - Desember 1999. JSON merupakan format teks yang tidak bergantung pada bahasa pemrograman apapun karena menggunakan gaya bahasa yang umum digunakan oleh programmer keluarga C termasuk C, C++, C#, Java, JavaScript, Perl, Python dll. Oleh karena sifat-sifat tersebut, menjadikan JSON ideal sebagai bahasa pertukaran-data.

3.6.2 Penggunaan JSON

1. Method save()

Pada method ini, kami menggunakannya untuk menyimpan array of String ke dalam SharedPreferences. Seperti telah kita ketahui, SharedPreferences tidak dapat menerima array of String, maka kami menggunakan kelas JSONArray untuk mengubah array of String tadi menjadi satu String saja, yang kemudian disimpan dalam SharedPreferences.

2. Method load()

Setelah method save() dilakukan, String yang disimpan dalam SharedPreferences dapat kita ubah kembali menjadi array of String dengan menggunakan kelas JSONArray.

BAB IV

PENUTUP

4.1 Kesimpulan

Berdasarkan bab-bab sebelumnya, penulis memberi kesimpulan sebagai berikut:

1. Aplikasi ini cukup mudah untuk digunakan karena pengguna hanya perlu mengikuti cara penggunaan aplikasi
2. Dengan teknologi android aplikasi ini mampu menampilkan ListView, Pop-Up dan Spinner

4.2 Saran

Berdasarkan kesimpulan yang telah diperoleh, maka saran untuk pengembangan aplikasi ini yaitu:

1. Untuk pengembangan ke depan, akan diperbanyak fitur yang dapat diperbaiki dan tampilan agar lebih menarik.
2. Aplikasi kalkulator ini bisa ditambah fungsi lainnya, misal menambahkan sebuah fungsi yang sudah disimpan sebelumnya, misalnya fungsi *pythagoras*, *relativitas*, dan sebagainya

DAFTAR PUSTAKA

Pembuatan laporan:

Admin. *Sejarah Sistem Operasi Android*, web: <https://www.intanblog.com/sejarah-sistem-operasi-android/>

Ramadhan Purnama. *Kelebihan dan Kekurangan Android*, web: <https://www.smartphoneku.com/kelebihan-dan-kekurangan-android>

Inet. *Informasi Internet*, web: <https://www.informasi-internet.com/2016/09/android-sdk.html>

Binus University. (2015). *Pengenalan Mobile 2*, web: https://www.academia.edu/11217588/LAPORAN_PROYEK_APLIKASI_ANDROID

json. *Pengenalan JSON*, web: <https://www.json.org/json-id.html>

Pembuatan aplikasi:

[Hisham Muneer](#). (7 September 2015). *15 - JSON Parsing, Using For Loop on JSONArray - Android Studio*, web: <https://m.youtube.com/watch?v=bFMvE-zSjGE&t=15s>

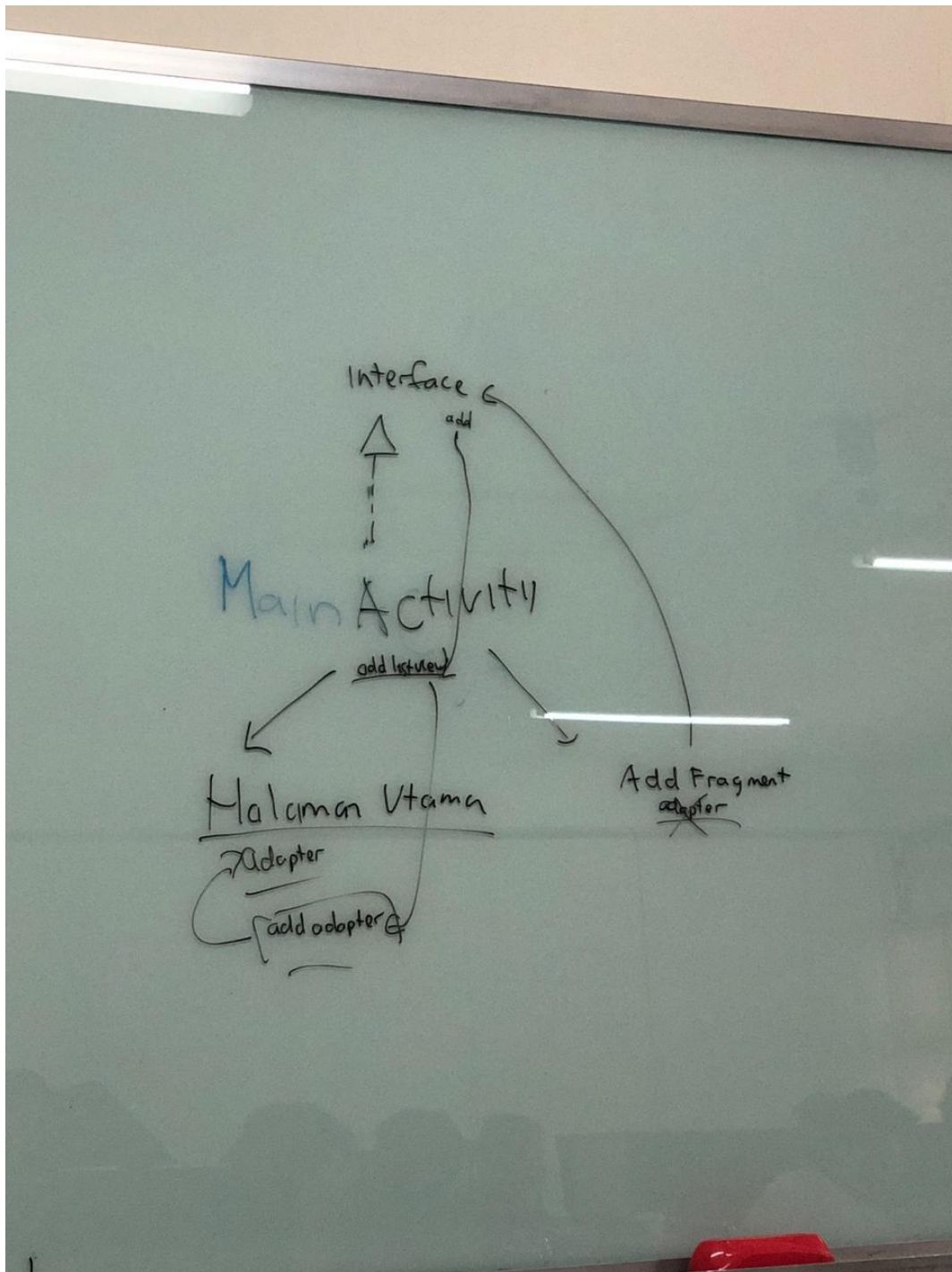
<https://stackoverflow.com/questions/26284160/saving-listview-item-permanently-in-the-android-application>

<https://medium.com/@zackcosborn/step-by-step-recyclerview-swipe-to-delete-and-undo-7bbae1fce27e>

<https://stackoverflow.com/questions/40624554/difference-between-appsrcompat-and-androidsrc-in-androids-layout-xml>

<https://material.io/develop/android/components/floating-action-button/>

LAMPIRAN



GAMBAR L.1 Ilustrasi Penggambaran Penggunaan Adapter