

# **Tugas Besar Pemrograman Pada Perangkat Bergerak**

## **“Manga Reader”**



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, 20 November 2019

Hormat Kami,

Penulis

# DAFTAR ISI

<b>PRAKATA</b>	<b>2</b>
<b>DAFTAR ISI</b>	<b>3</b>
<b>DAFTAR GAMBAR</b>	<b>4</b>
<b>BAB I</b>	<b>5</b>
<b>PENDAHULUAN</b>	<b>6</b>
1.1 Latar Belakang	6
1.2 Ruang Lingkup	6
1.3 Tujuan dan Manfaat	6
1.4 Sistematika Penulisan	6
<b>BAB II</b>	<b>7</b>
<b>LANDASAN TEORI</b>	<b>8</b>
2.1 Sejarah Sistem Operasi Android	8
2.2 Versi Sistem Operasi Android	8
2.3 Kelebihan dan Kekurangan Android	9
2.3.1 Kelebihan Android:	10
2.3.2 Kekurangan Android	11
2.4 Android SDK	11
2.5 Android Virtual device	11
2.6 Java Development Kit	11
2.7 Android Studio	12
<b>BAB III</b>	<b>12</b>
<b>IMPLEMENTASI DAN PEMBAHASAN</b>	<b>13</b>
3.1 Latar Belakang Aplikasi	13
3.2.1 Halaman Utama	13
3.2.2 Halaman Utama (2)	14
3.2.3 Drawer Category	15
3.2.4 Halaman Detail Manga	16
<b>3.2.5 Halaman Baca Manga</b>	<b>17</b>
3.3 Cara Menggunakan Aplikasi	18
3.4 Diagram Kelas	19
3.5 Penjelasan berbagai Kelas, Atribut, dan Method	24
3.6 Library	40
3.6.1 Definisi Library	40

3.6.2 Penggunaan Library	40
BAB IV	41
PENUTUP	42
4.1 Kesimpulan	42
4.2 Saran	42
DAFTAR PUSTAKA	42
LAMPIRAN	43

# DAFTAR GAMBAR

GAMBAR 3.2.1 TAMPILAN MAINACTIVITY .....	13
GAMBAR 3.2.2 TAMPILAN HALAMAN MAINACTIVITY.....	14
GAMBAR 3.2.3 TAMPILAN HALAMAN .....	15
GAMBAR 3.2.4 TAMPILAN HALAMAN .....	16
GAMBAR 3.4.1 ILUSTRASI MVVM .....	17
GAMBAR 3.4.2 KELAS ADAPTER CHAPTER.....	18
GAMBAR 3.4.3 KELAS ADAPTER MAIN .....	18
GAMBAR 3.4.4 KELAS ADAPTER MANGA.....	18
GAMBAR 3.4.5 KELAS ADAPTER PAGE.....	19
GAMBAR 3.4.6 KELAS MANGA CHAPTER DETAIL .....	19
GAMBAR 3.4.7 KELAS MANGA LIST.....	19
GAMBAR 3.4.8 KELAS MANGA MODEL.....	19
GAMBAR 3.4.9 KELAS MANGA PICTURE.....	20
GAMBAR 3.4.10 KELAS MANGA SEARCH.....	20
GAMBAR 3.4.11 KELAS MYSINGLETON.....	20
GAMBAR 3.4.12 KELAS MAIN ACTIVITY.....	21
GAMBAR 3.4.12 KELAS MANGA DETAIL ACTIVITY.....	21
GAMBAR 3.4.13 KELAS DIAGRAM KESELURUHAN .....	21



# BAB I

## PENDAHULUAN

### 1.1 Latar Belakang

Dewasa ini teknologi terus berkembang di berbagai bidang termasuk entertainment, dan salah satunya mempengaruhi bidang percetakan. Mudahnya akses internet saat ini memudahkan masyarakat untuk mendapatkan informasi melalui aplikasi maupun website, sedangkan untuk kalangan muda membaca komik akan lebih mudah jika dapat diakses melalui aplikasi.

Saat ini banyak aplikasi yang memiliki fungsional untuk memudahkan masyarakat dalam membaca komik secara gratis. Aplikasi yang dibuat juga dapat melakukan filter pencarian untuk menampilkan daftar komik yang sesuai dengan keinginan pengguna. Untuk tugas kali ini penulis diharapkan untuk membuat aplikasi yang dapat menampilkan daftar komik yang diakses dari <http://www.mangaeden.com> (salah satu website penyedia layanan baca komik).

### 1.2 Ruang Lingkup

Ruang lingkup dari perancangan mobile-game ini adalah sebagai berikut:

1. Game ini ditujukan untuk Sistem Operasi Android minimal versi 5.0 Jelly Bean dengan memakai IDE Android Studio dan menggunakan Java sebagai bahasa pemrograman.
2. Perancangan tampilan daftar manga
3. Perancangan tampilan membaca manga (pengguna dapat memperbesar gambar dan melakukan *swipe* pada halaman ini)
4. Mengakses komik-komik yang tersedia pada halaman mangaeden.com untuk ditampilkan pada aplikasi yang dibangun menggunakan web service
5. Mengakses chapter komik yang diinginkan pada aplikasi

### 1.3 Tujuan dan Manfaat

Tujuan dari aplikasi ini adalah mengembangkan aplikasi yang memudahkan pengguna untuk memilih dan membaca komik (manga) yang tersedia pada website mangaeden:

1. Mengembangkan aplikasi untuk membaca komik berbasis android
2. Melatih kemampuan untuk lebih memahami penggunaan Web Service dan API
3. Meningkatkan pemahaman perancangan perangkat lunak 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, 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 penggunanya.

### 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 **MainActivity**, tampilan Fragment **MainFragment**, tampilan Fragment **ReadingFragment**, diagram kelas, penjelasan kelas-kelas, penjelasan atribut-atribut, dan penjelasan method-method.

#### 3.1 Latar Belakang Aplikasi

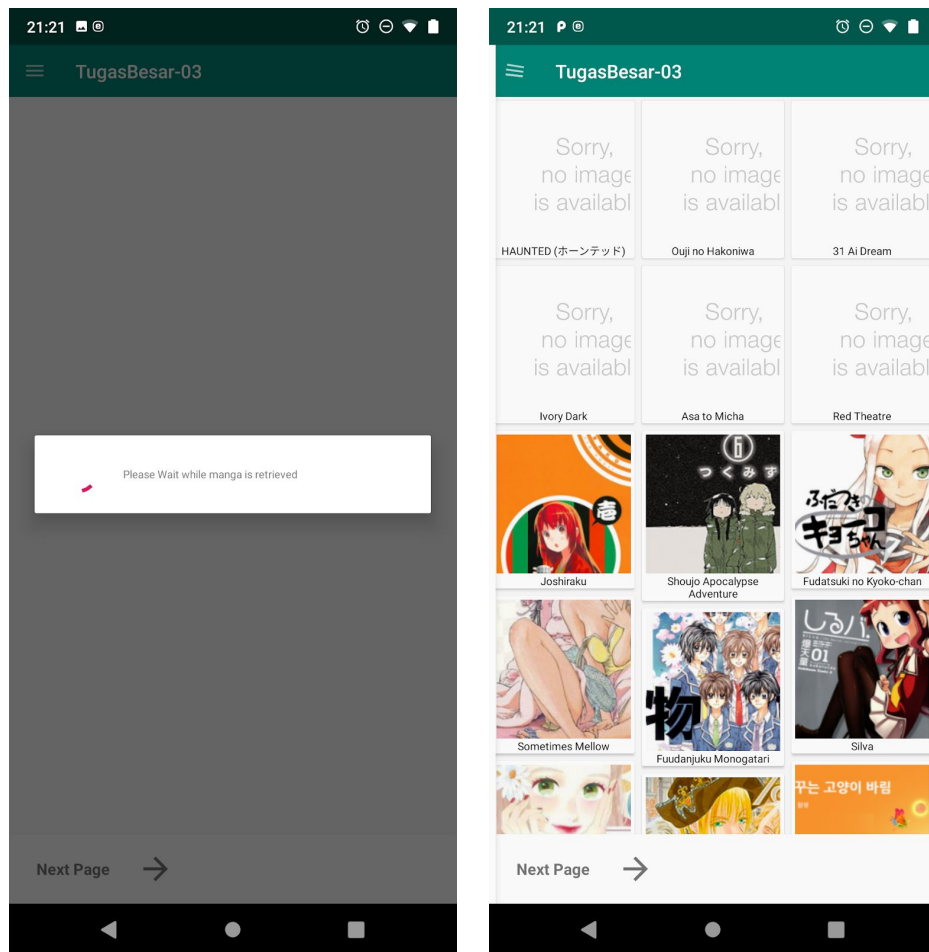
Aplikasi yang dibuat merupakan manga-reader sederhana dimana aplikasi dapat menampilkan daftar komik yang tersedia pada website <http://www.mangaeden.com> dan pengguna dapat membaca komik yang diinginkan melalui aplikasi yang dibangun. Pengguna juga dapat melakukan pencarian komik tertentu dengan memasukkan nama komik yang ingin dicari. Pengguna juga dapat mencari komik yang diinginkan di akses berdasarkan kategori genre komik. Selain melakukan pencarian dan filter, pengguna juga dapat mengakses komik dan membaca komik di aplikasi yang dibangun.

#### 3.2 Tampilan Aplikasi

Aplikasi kami terdiri dari satu kelas main yaitu kelas **MainActivity**, dan **MainFragment** yang mengimplementasikannya, juga merupakan halaman utama dimana list manga akan ditampilkan. Pada kelas **MainFragment** terdapat kumpulan manga yang telah ditampilkan berdasarkan kategori yang dipilih. Secara default, kategori bawaan adalah “All Category”, sehingga semua manga akan ditampilkan. Di bagian bawah ada **TextView** “Next Page” jika user berada di page 1, dan akan muncul “Previous Page” bila user berada di page 2 atau lebih

Berikutnya ada fragment **ReadingFragment**, yang mengimplementasikan ketika user sedang membaca manga. Sebuah list berisi halaman yang tersedia akan ditampilkan, dan bila user melakukan klik pada gambar manga, maka list berisi halaman tersebut akan disembunyikan tampilannya.

### 3.2.1 Halaman Utama

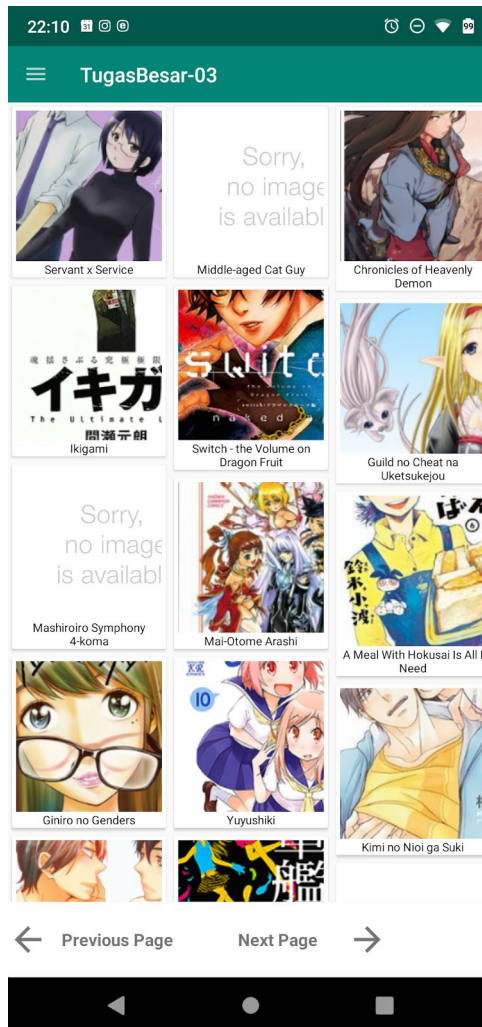


Gambar 3.2.1 MainFragment

Pada layout ini terdapat View:

1. GridListView yang menampilkan daftar manga yang dapat dibaca
2. Button “Next Page” untuk menampilkan list manga pada halaman berikutnya
3. Icon “Menu kategori” untuk menampilkan drawer pilihan kategori manga

### 3.2.2 Halaman Utama (2)

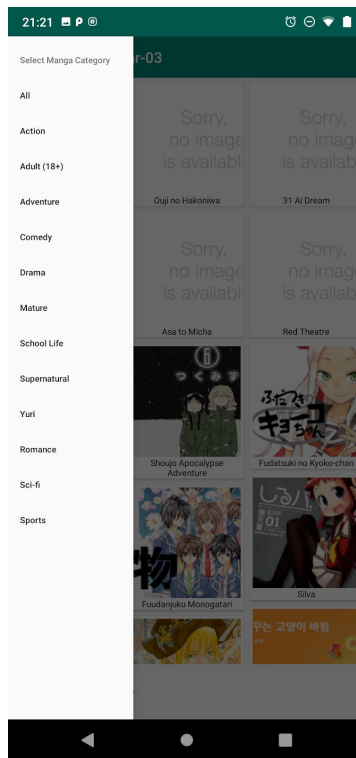


Gambar 3.2.2 Halaman Utama 2

Tampilan pada Gambar 3.2.2 merupakan tampilan saat user telah menekan Next Page pada halaman utama. Pada Dialog ini terdapat View yang sama dengan halaman utama namun untuk kembali ke halaman sebelumnya, pengguna dapat menekan tombol previous page.



### 3.2.3 Drawer Category

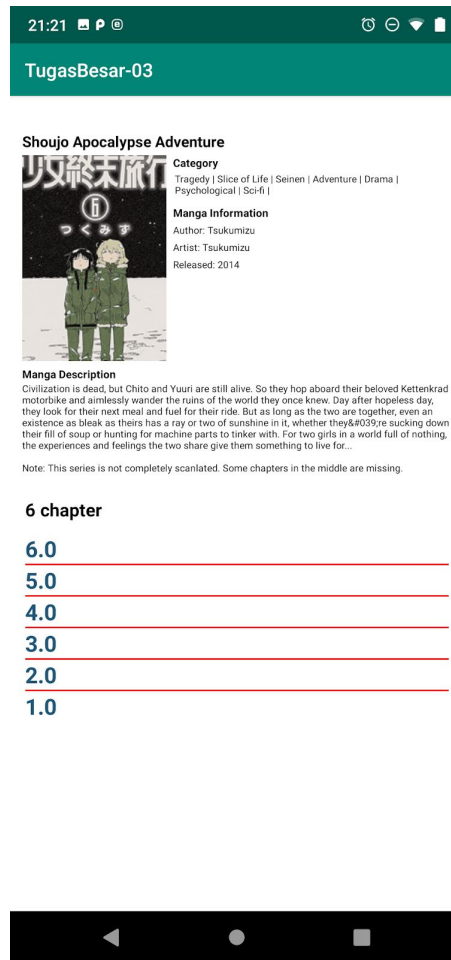


**Gambar 3.2.3** Halaman Drawer

Tampilan pada Gambar 3.2.3 merupakan tampilan saat user telah menekan ikon ‘menu kategori’ pada halaman utama. Tampilan ini bertujuan untuk menampilkan drawer yang berisi list kategori manga. Pada dialog ini terdapat View:

1. Listview yang menampilkan daftar kategori manga yang tersedia

### 3.2.4 Halaman Detail Manga

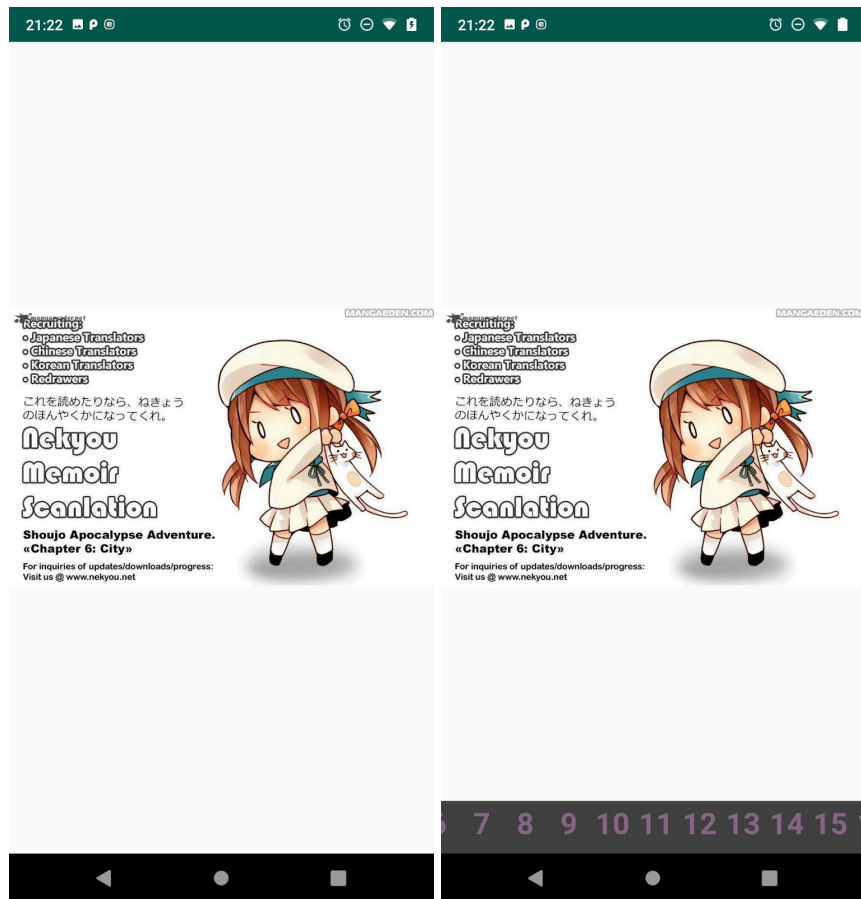


Gambar 3.2.4 Detail Manga

Tampilan pada Gambar 3.2.4 merupakan tampilan ketika pengguna telah memilih dan menekan gambar manga yang ingin dibaca .Pada tampilan ini terdapat View:

1. TextView “Manga Information :” sebagai judul informasi detail manga yang akan dibaca
2. TextView “Author : ” sebagai keterangan informasi nama penulis manga yang akan dibaca
3. TextView “Artist : ” sebagai keterangan informasi nama artist manga yang akan dibaca
4. TextView “Released :” sebagai keterangan informasi waktu terbit manga yang akan dibaca
5. TextView “Manga Description :” sebagai keterangan sinopsis manga yang akan dibaca
6. TextView “Chapter :” sebagai keterangan status game yang dimainkan telah berakhir
7. Listview chapter : untuk menampilkan list chapter yang ingin dibaca

### 3.2.5 Halaman Baca Manga



Gambar 3.2.5 Baca manga

Tampilan pada Gambar 3.2.4 merupakan tampilan saat pengguna telah memilih dan menekan chapter yang ingin dibaca. Pada tampilan ini terdapat View:

1. ImageView komik yang sedang dibaca
2. ListView untuk pindah halaman komik yang dibaca

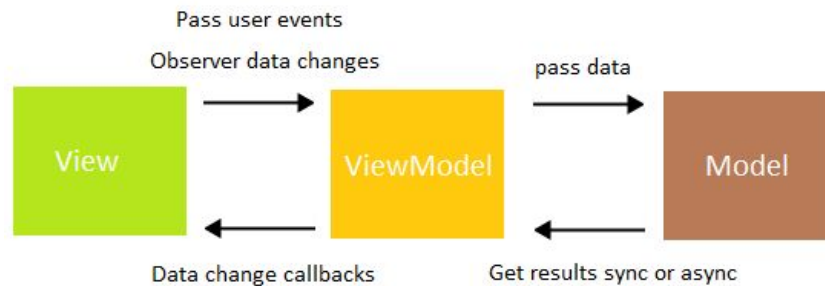
## 3.3 Cara Menggunakan Aplikasi

1. Buka aplikasi
2. Pilih manga yang berada pada list halaman utama atau
3. Klik icon kategori yang berada di atas kiri header aplikasi
4. Pilih category manga yang dicari
5. Tekan gambar manga yang ingin dibaca
6. Pilih dan tekan chapter yang ingin dibaca
7. Untuk zoom in gambar, tarik gambar ke arah luar dengan dua jari

8. Untuk zoom out gambar, tarik gambar kearah dalam dengan dua jari
9. Swipe untuk pindah halaman

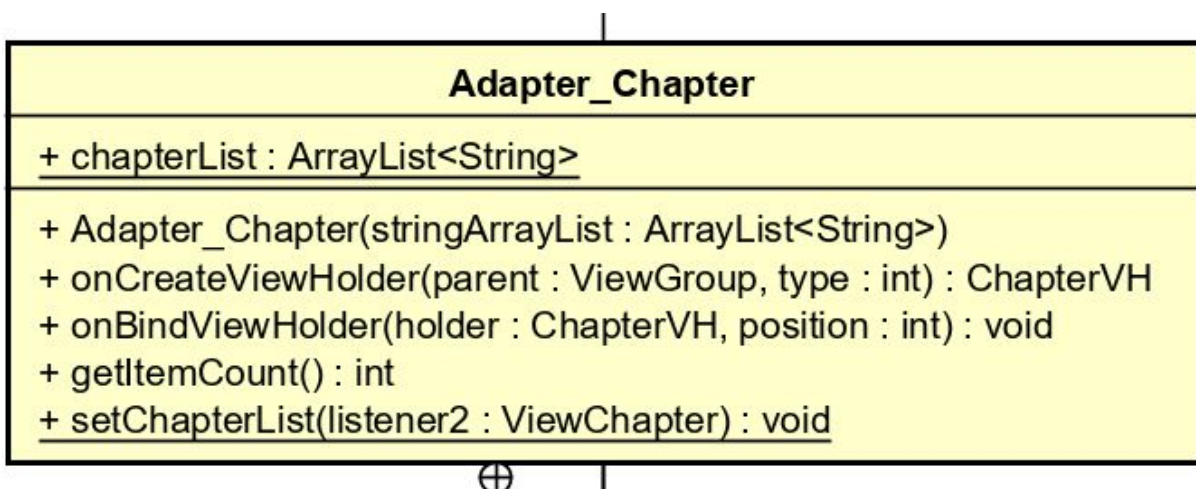
### 3.4 Diagram Kelas

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



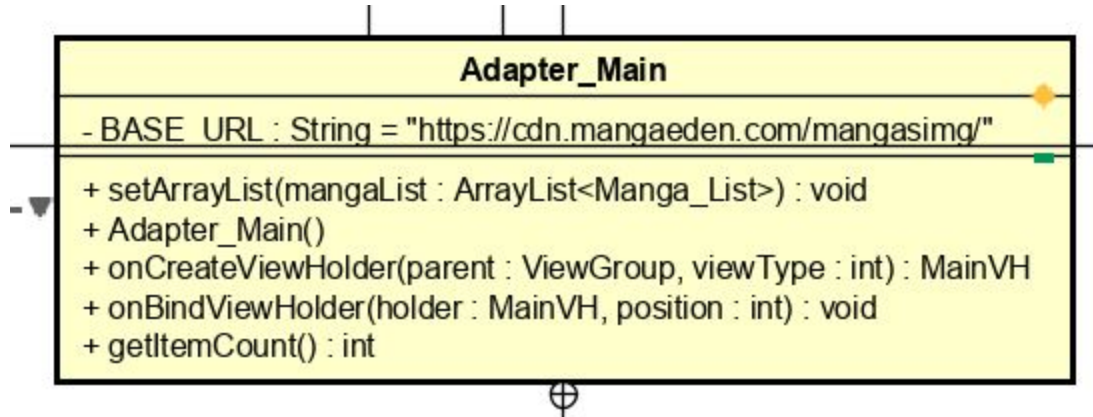
Gambar 3.4.1 Ilustrasi MVVM

1. Kelas yang berfungsi sebagai Adapter
  - a. Adapter\_Chapter.java



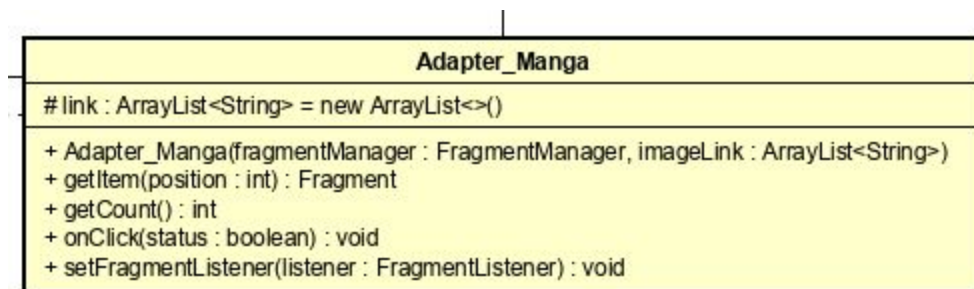
Gambar 3.4.2 Adapter\_Chapter.java

b. Adapter\_Main.java



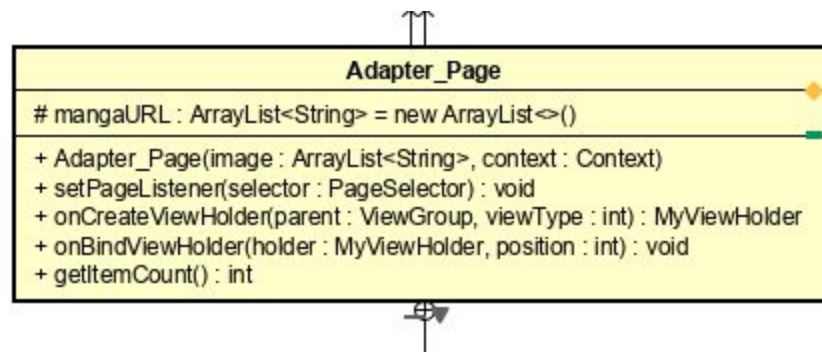
Gambar 3.4.3 Adapter\_Main.java

c. Adapter\_Manga.java



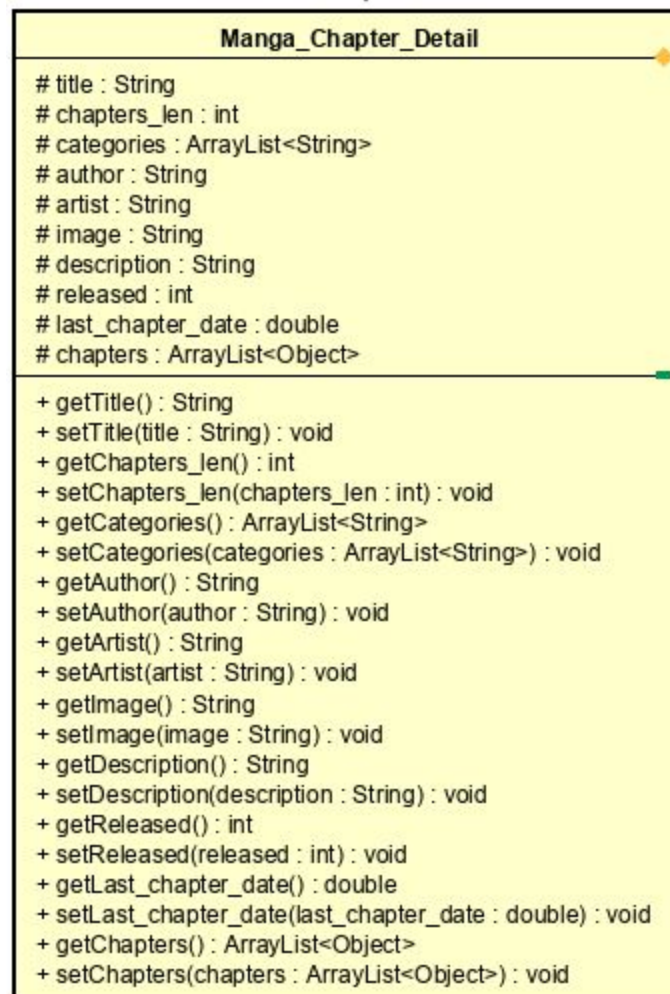
Gambar 3.4.4 Adapter\_Main.java

d. Adapter\_Page.java



Gambar 3.4.5 Adapter-Page.java

2. Kelas yang berfungsi sebagai Model
- a. Manga\_Chapter\_Detail.java



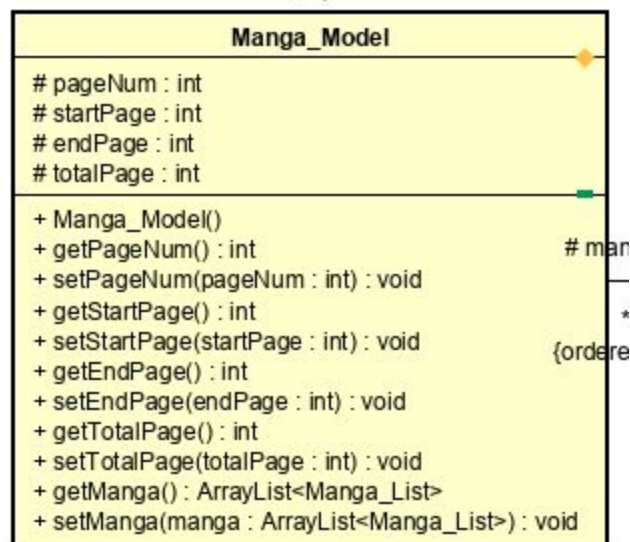
Gambar 3.4.6 Manga\_Chapter\_Detail.java

- b. Manga\_List.java



Gambar 3.4.7 Manga\_List.java

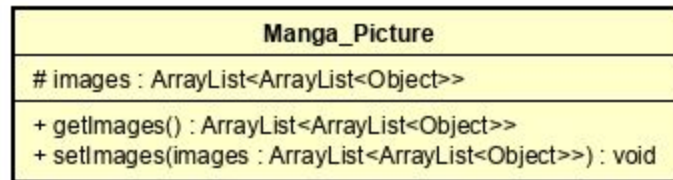
c. Manga\_Model.java



Gambar 3.4.7 Manga\_Model.java

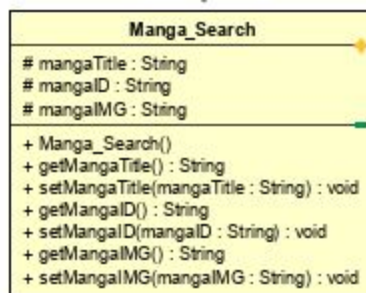


d. Manga\_Picture.java



Gambar 3.4.8 Manga\_Picture.java

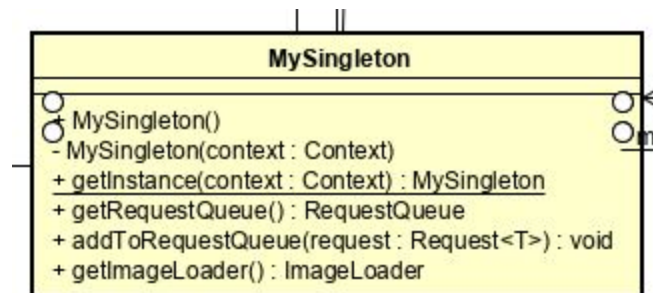
e. Manga\_Search.java



Gambar 3.4.9 Manga\_Search.java

3. Kelas yang berfungsi sebagai Networking (mengatur koneksi device dengan server)

a. MySingleton.java



Gambar 3.4.10 MySingleton.java



4. Kelas yang berfungsi sebagai View

a. MainActivity.java

```
MainActivity
# fabPause : FloatingActionButton
# fabMenu : FloatingActionButton
# tvPlayerName : TextView
# tvScore : TextView
# tvWelcome : TextView
# ivPlane : ImageView
# rcCanvas : FrameLayout
# handler : Handler
# timer : Timer
# enemyPlane : EnemyPlane
# bomb : Bomb
# fuel : Fuel
# reward : Reward
# requester : MyAsyncTask
# uri : String
- planeHeight : int
- canvasHeight : int
- planeY : int
# action_flg : boolean
# activityStart : boolean
# statusButtonPause : boolean
# score : int

# onCreate(savedInstanceState : Bundle) : void
+ changePosition() : void
- canvasLimit() : void
+ hitStatus() : void
+ onTouchEvent(motionEvent : MotionEvent) : boolean
+ run() : void
+ run() : void
+ onClick(view : View) : void
+ run() : void
+ run() : void
+ closeApp() : void
+ updateScore(score : String) : void
+ gameSelesai() : void
+ getScore() : int
```

Gambar 3.4.11 MainActivity.java

b. MangaReader.java

```
MangaReader

- BASE_URL : String = "https://www.mangaeden.com/api/chapter/"
# mangaList : ArrayList<String> = new ArrayList<>()

# onCreate(savedInstanceState : Bundle) : void
- volleyCall() : void
+ onClick(status : boolean) : void
+ onBackPressed() : void
```

Gambar 3.4.12 MangaReader.java

### 3.5 Penjelasan berbagai Kelas, Atribut, dan Method

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

<b>Adapter_Chapter</b>	
Kelas Adapter_Chapter berfungsi sebagai adapter untuk bagian chapter dari manga, ditampilkan ke RecyclerView yang berada di kelas MangaDetailActivity	
<b>Atribut</b>	<b>Penjelasan</b>
chapterList	Sebagai list untuk menampung seluruh chapter yang dimiliki oleh 1 buah manga yang di-klik
listener	Berfungsi sebagai sebuah listener yang menangkap “klik” ketika user melakukan klik pada chapter manga yang diinginkan
context	Atribut ini digunakan untuk menyambungkan atau mengkoneksikan kelas Adapter_Chapter dengan kelas activity yang terkait
<b>Method</b>	<b>Penjelasan</b>
onBindViewHolder	Method ini melakukan perubahan pada TextView judul chapter di ViewHoldernya
getItemCount	Method ini mengembalikan total isi dari array chapterList
setChapterlist	Method ini melakukan set chapterList melalui interface listenernya
<b>Adapter_Page</b>	

Kelas Adapter\_Page berfungsi sebagai adapter untuk bagian halaman (RecyclerView) yang berada pada bagian bawah ketika sedang read manga nya

Atribut	Penjelasan
context	Atribut ini digunakan untuk menyambungkan atau mengkoneksikan kelas Adapter_Chapter dengan kelas activity yang terkait
mangaURL	Atribut ini merupakan base URL yang dibutuhkan untuk melakukan pengambilan data
listener	Berfungsi sebagai sebuah listener yang menangkap “klik” ketika user melakukan klik pada halaman manga yang diinginkan
Method	Penjelasan
setPageListener	Method ini berfungsi sebagai set listener ketika ImageView halaman nya di klik
onCreateViewHolder	Method ini berfungsi sebagai inisialisasi ketika ViewHoldernya dibuat pertama kali
onBindViewHolder	Method ini berfungsi untuk mengisi apa yang dilakukan ketika akan melakukan update pada ViewHolder nya
getItemCount	Method ini berfungsi untuk menghitung total halaman yang ada di list halamannya
setItemSelectListener	Method ini berfungsi sebagai set listener ketika TextView halamannya di klik
<b>Adapter_Manga</b>	

Kelas Adapter_Manga berfungsi sebagai adapter untuk bagian halaman ketika sedang melakukan read pada manga di page tertentu	
Atribut	Penjelasan
link	Berfungsi sebagai atribut yang mengimplementasikan list banyaknya manga dalam sebuah chapter
listener	Berfungsi sebagai implementasi interface dari FragmentListener
Method	Penjelasan
getItem	Method untuk mengambil item ke i dari array link
getCount	Method untuk mengambil panjang array dari array link
OnClick	Method untuk klik pada item di view
setFragmentListener	Method yang melakukan perubahan fragment pada onklik
<b>Adapter_Main</b>	
Kelas Adapter_Main berfungsi sebagai kelas untuk melakukan update pada GridView gambar manga yang berada di MainFragment	
Atribut	Penjelasan
mangaList	Atribut ini berfungsi sebagai list yang berisi manga untuk kemudian di update
context	Atribut ini digunakan untuk menyambungkan atau mengkoneksikan kelas Adapter_Chapter dengan kelas activity yang terkait

BASE_URL	String yg berisi  "https://cdn.mangaeden.com/mangasing/"
<b>Method</b>	<b>Penjelasan</b>
onCreateViewHolder	Untuk mengisi view yang telah di buat
onBindViewHolder	Method ini melakukan perubahan pada TextView judul chapter di ViewHoldernya
getItemCount	Method untuk mengambil panjang array
<b>Manga_Chapter_Detail</b>	
Kelas Manga_Chapter_Detail berfungsi sebagai Activity yang mengimplementasikan kelas MangaReader di ReadingFragment. Karena kelas ini berfungsi sebagai model untuk koneksi langsung ke Webservice, maka atribut nya harus sesuai dengan yang diminta oleh Webservicenya.	
<b>Atribut</b>	<b>Penjelasan</b>
title	Atribut ini untuk mengisi judul manga
chapters_len	Atribut ini untuk berapa banyak halaman pada 1 chapter
categories	Atribut ini berisi kategori dari manga yang di pilih
author	Atribut ini berisi author dari manga yang di pilih
artist	Atribut ini berisi artist dari manga yang di pilih
image	Atribut ini berisi picture dari manga yang di pilih

description	Atribut ini berisi deskripsi dari manga yang di pilih
released	Atribut ini berisi tanggal keluar dari manga yang dipilih
last_chapter_date	Atribut ini berisi tanggal chapter terakhir dari manga yang di pilih
chapters	Atribut ini berisi chapter manga dari manga yang di pilih
<b>Method</b>	<b>Penjelasan</b>
getTitle	Method untuk mengambil atribut title
getChapters_len	Method untuk mengambil atribut Chapters_length
getCategories	Method untuk mengambil atribut categories
getAuthor	Method untuk mengambil atribut author
getArtist	Method untuk mengambil atribut artist
getReleased	Method untuk mengambil atribut released
getChapters	Method untuk mengambil atribut chapters
<b>Manga_Picture</b>	
<p>Kelas Manga_Picture berfungsi sebagai implementasi untuk menerima gambar manga dari server. Karena kelas ini berfungsi sebagai model untuk koneksi langsung ke Webservice, maka atribut nya harus sesuai dengan yang diminta oleh Webservicenya.</p>	

Atribut	Penjelasan
images	Atribut ini berfungsi sebagai ArrayList yang menampung objek
Method	Penjelasan
getImages	Method untuk mengambil atribut images
<b>Manga_Model</b>	
Kelas ini berfungsi sebagai pemodelan yang mengimplementasi ketika kita sedang membaca sebuah manga	
Atribut	Penjelasan
page	Atribut ini merupakan halaman page manga yang sedang dibaca
manga	Atribut ini merupakan list sebuah manga yang dibaca
end	Atribut ini menunjukkan halaman akhir dari sebuah manga
start	Atribut ini menunjukkan halaman pertama dari sebuah manga
total	Atribut ini menunjukkan total halaman yang dimiliki sebuah manga
Method	Penjelasan
getManga	Method untuk mengambil atribut manga
<b>Manga_List</b>	

Kelas Manga\_List berfungsi sebagai implementasi kelas untuk menerima manga yang di klik di MainFragment. Karena kelas ini berfungsi sebagai model untuk koneksi langsung ke Webservice, maka atribut nya harus sesuai dengan yang diminta oleh Webservicenya.

Atribut	Penjelasan
im	Atribut ini berfungsi sebagai atribut untuk menerima url gambar
t	Atribut ini berfungsi sebagai title (judul) dari sebuah manga
i	Atribut ini menunjukkan id dari sebuah manga
a	Atribut ini menunjukkan nama lain (alias) dari sebuah manga
s	Atribut ini menunjukkan status
c	Atribut ini menunjukkan kategori sebuah manga
ld	Atribut ini menunjukkan kapan chapter terakhir dibuat
h	Atribut ini menunjukkan status sebuah manga hits atau tidak
Method	Penjelasan
getId	Method untuk mengambil atribut Id (menunjukkan kapan chapter terakhir dibuat)
getT	Method untuk mengambil atribut title
getI	Method untuk mengambil atribut id
getC	Method untuk mengambil atribut Category



getA	Method untuk mengambil atribut Alias
getIm	Method untuk mengambil atribut url gambar
getH	Method untuk mengambil atribut Hits
getS	Method untuk mengambil atribut Status
getT	Method untuk mengambil atribut Title
<b>Manga_Search</b>	
Kelas Manga_Search berfungsi sebagai implementasi ketika kita melakukan search sebuah manga	
<b>Atribut</b>	<b>Penjelasan</b>
title	Atribut ini merepresentasikan judul manga yang akan dicari
id	Atribut ini merepresentasikan ID manga yang akan dicari
img	Atribut ini merepresentasikan gambar manga yang akan dicari
<b>Method</b>	<b>Penjelasan</b>
getTitle	Method untuk mengambil atribut title
getImg	Method untuk mengambil atribut img
getId	Method untuk mengambil pada atribut id
<b>MainFragment</b>	

Kelas ini berfungsi sebagai representasi halaman utama, yaitu RecyclerView yang dibuat Grid menjadi 3 lajur	
Atribut	Penjelasan
recycleView	Atribut ini merupakan layout untuk menampilkan mangaList
mainAdapter	Atribut ini merupakan adapter untuk GridView yang menampilkan layout manga_list_item
mangaList	Atribut ini merupakan sebuah list dari manga yang diterima dari server
Method	Penjelasan
onCreateView	Method ini melakukan pembuatan view ketika pertama activity dilakukan
<b>MangaDetailActivity</b>	
Kelas ini berfungsi sebagai implementasi ketika kita melakukan klik pada sebuah gambar manga di MainFragment	
Atribut	Penjelasan
BASE_URL	String yg berisi <code>"https://www.mangaeden.com/api/manga/"</code>
mangaDetailLayout	Atribut ini merupakan container layout dari
recyclerView	Merupakan RecyclerView untuk menampilkan list manga
ivManga	Atribut untuk menampilkan gambar manga

adapter	Atribut untuk menampilkan daftar chapter yang dimiliki manga tersebut
loading	Merupakan sebuah ProgressDialog untuk menampilkan sebuah pesan yang mirip Toast, tapi ini lebih gaya
chapterNumber	Merupakan ArrayList untuk menampilkan chapter yang dimiliki manga tersebut
chapterId	Merupakan ArrayList untuk menyimpan id dari chapter
tvTitle	TextView untuk menginformasikan judul komik
tvAuthor	TextView untuk menginformasikan penulis komik
tvChapter	TextView untuk menginformasikan judul chapter komik
tvArtist	TextView untuk menginformasikan artist komik
tvReleased	TextView untuk menginformasikan tanggal terbit komik
tvGenre	TextView untuk menginformasikan kategori komik
tvDescription	TextView untuk menginformasikan sinopsis komik
<b>Method</b>	<b>Penjelasan</b>
volleyCall	Method ini mengimplementasikan pemanggilan webservice melalui library Volley
getChapter	Method ini untuk mendapatkan chapter yang diinginkan dari manga tersebut berdasarkan input yang di klik

dialogBuilder	Method ini untuk membuat sebuah AlertDialog yang menampilkan sebuah pop-up alert
<b>MainActivity</b>	
Kelas MainActivity berfungsi sebagai Activity awal ketika sebuah kelas dijalankan	
<b>Atribut</b>	<b>Penjelasan</b>
recyclerView	Atribut ini untuk mengakses view
BASE_URL	String ini berisi "https://www.mangaeden.com/api/list/0/"
mangaList	Atribut ini berfungsi sebagai list yang berisi manga untuk kemudian di update
mangaList2	Atribut ini berfungsi sebagai list yang berisi manga untuk kemudian diupdate
progressDialog	Atribut ini menampilkan sebuah pop-up yang pertama kali muncul ketika sedang retrieve manga dari server
adapter	Atribut ini untuk menampilkan gambar manga berdasarkan pilihan kategori, secara default adalah semua kategori untuk kemudian ditampilkan di RecyclerView
searchView	Atribut ini di gunakan untuk mencari manga
previousButton	Atribut ini di gunakan untuk mengakses previousbutton
nextButton	Atribut ini di gunakan untuk mengakses nextbutton
categoryPreferences	Atribut ini digunakan untuk menampilkan category

searchEditor	Atribut ini digunakan mengedit hasil di keluarkan search view
mangaSearch	Atribut ini digunakan untuk membuat button search
currentPage	Atribut ini berisi integer = 1
toolbar	Atribut ini untuk memunculkan button menu
drawerLayout	Atribut ini digunakan untuk menggambar langsung ke dalam image view
<b>Methods</b>	<b>Penjelasan</b>
onQueryTextSubmit	Melakukan search pada manga berdasarkan query yang diberikan
onQueryTextChange	Method ini menambahkan hasil search pada pada ArrayList mangaList3
onSearchViewShown	Method ini menentukan apa yang dilakukan ketika SearchView terbuka
onSearchViewClosed	Method ini menentukan apa yang dilakukan ketika SearchView nya ditutup
sortMangaList	Method ini melakukan sorting pada Array yang diinput
volleyRequest	Method ini melakukan request ke webservice
onCreateOptionsMenu	Method ini menunjukkan pembuatan option menu
onActivityResult	Method ini mengatur hasil pencarian manga berdasarkan search

onBackPressed	Mengontrol apa yang dilakukan activity ketika user menekan tombol "Back"
onNavigationItemSelectedListener	Method ini mengatur apa yang dilakukan oleh searchEditor ketika kita memilih kategori dari drawer
searchVolleyRequest	Melakukan request ke server berdasarkan SearchView
showcategoryData	Melakukan show manga berdasarkan kategori yang sudah dipilih
onDestroy	Mengontrol apa yang terjadi ketika user keluar dari aplikasi, yaitu destroy activity
<b>MangaReader</b>	
Kelas MangaReader digunakan untuk memodelkan manga yang ditampilkan untuk user di ReadingFragment	
<b>Atribut</b>	<b>Penjelasan</b>
imageZoom	Menginisialisasi kelas ImageZoom
mangaAdapter	Sebagai adapter gambar manga yang diperoleh dari mangaList
mangaList	Daftar gambar manga
pageAdapter	Sebagai adapter page dari manga tersebut
BASE_URL	Link untuk mengambil gambar
pageList	Layout RecyclerView untuk menampilkan ImageView halaman

Methods	Penjelasan
volleyCall	Method ini melakukan koneksi pada webservice
onBackPressed	Method ini mengatur apa yang dilakukan activity ketika user menekan tombol "Back"
OnClick	Method ini mengatur apa yang dilakukan list halaman ketika user melakukan tap pada gambar manga ( hidden dan show)
<b>MySingleton</b>	
Kelas MySingleton digunakan sebagai desain pattern untuk mengatur koneksi dengan Webservice	
Atribut	Penjelasan
mySingleton	Atribut ini berfungsi sebagai inisialisasi kelas MySingleton
requestQueue	Atribut ini untuk menunjukkan kelas RequestQueue
imageLoader	Atribut ini menunjukkan kelas ImageLoader yang digunakan untuk menerima gambar dari Webservice
context	Atribut ini sebagai penghubung dengan Activity yang sedang berjalan
Methods	Penjelasan
getInstance	Method ini melakukan inisialisasi untuk kelas MySingleton
getRequestQueue	Method ini menyiapkan queue untuk request ke Webservice
<b>ReadingFragment</b>	

Kelas ReadingFragment ini mengimplementasikan ketika user sedang membaca per halaman manga dari sebuah chapter	
Atribut	Penjelasan
ivtManga	Atribut ini sebagai layout sebuah gambar (perhalaman manga)
listener	Atribut ini berfungsi sebagai listener ketika manga ditekan atau tidak
gestureDetector	Atribut yang berfungsi sebagai detector ketika sebuah gestur pinch to zoom dilakukan di manga
progressBar	Atribut ini berfungsi untuk menampilkan tulisan pop-up bahwa manga sedang diterima dari server
Methods	Penjelasan
onLoadFailed	Method ini mengatur apa yang dilakukan oleh progressBar ketika Glide gagal menerima gambar manga dari server
onResourceReady	Method ini mengatur apa yang dilakukan oleh progressBar ketika Glide berhasil menerima gambar dari server
setFragmentManager	Method ini melakukan set pada atribut listener
onSingleTapConfirmed	Method ini melakukan sesuatu ketika gambar manga di-tap 1 kali
PageSelector	
Kelas interface ini berfungsi sebagai implementasi tampilan ketika memilih halaman manga berdasarkan klik	



Methods	Penjelasan
selectPage	Method untuk memilih halaman yang dituju
<b>ViewChapter</b>	
Kelas Interface ini berfungsi sebagai interface untuk mengimplementasikan ketika terjadi klik pada page yang ditampilkan	
getChapter	Method untuk mengambil chapter dari sebuah manga

## 3.6 Library

### 3.6.1 Definisi Library

Dalam ilmu komputer, Library adalah kumpulan sumber daya *non-volatile* yang digunakan oleh program komputer, dan seringkali digunakan untuk pengembangan perangkat lunak. Hal ini meliputi data konfigurasi, dokumentasi, data bantuan, template pesan, kode pra-tertulis dan subrutin, kelas, nilai atau spesifikasi tipe.

Pustaka juga merupakan kumpulan implementasi *behavior*, yang ditulis dalam bahasa pemrograman, yang memiliki antarmuka yang didefinisikan apabila *behavior* tersebut dipanggil. *Behavior* ini disediakan untuk digunakan kembali (*reuse*) oleh beberapa program yang independen. Program tersebut dapat memanggil *behavior* yang disediakan perpustakaan melalui suatu mekanisme bahasa.

### 3.6.2 Penggunaan Library

#### 3.6.2.1 Grid List View Adapters

Library grid list view adapters digunakan untuk memudahkan perancangan antarmuka yang dibangun, terutama pada tampilan list komik yang tersedia pada website mangaeden pada halaman utama aplikasi yang dibangun. Fungsi library ini adalah mengimplementasikan gridview, dimana setiap komik akan dibentuk dalam layout yang menyerupai card di dalam listview.

Dengan menggunakan library ini, aplikasi yang dibangun juga dimungkinkan untuk melakukan paginations, untuk menambahkan header-footer views dan memudahkan implementasi list dan cursor data.

### 3.6.2.2 Material Search View

Library Material Search View adalah library yang digunakan untuk mempercantik tampilan 'search bar' yang terdapat pada header aplikasi. Dengan menggunakan library ini, tampilan search bar menjadi responsive ketika pengguna ingin melakukan pencarian.

### 3.6.2.3 Volley

Volley unggul di operasi tipe RPC yang digunakan untuk mengisi UI, seperti mengambil halaman hasil pencarian sebagai data terstruktur. Ini terintegrasi dengan mudah dengan protokol apa pun dan keluar dari kotak dengan dukungan untuk string mentah, gambar, dan JSON.

# BAB IV

## PENUTUP

### 4.1 Kesimpulan

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

1. Penggunaan Library sangat membantu dalam pembangunan perangkat lunak baik untuk perancangan antarmuka maupun pemrograman fungsional.
2. Dalam menggunakan Webservice, agar bisa terhubung baik dengan API yang disediakan, kita sebaiknya memperhatikan apa yang menjadi syarat oleh penggunaan API tersebut. Sebagai contoh, penulis kurang memperhatikan API yang diberikan. Karena itu dalam pengumpulan kemarin, penulis gagal melakukan *retrieve* manga dari APInya. Rupanya setelah penulis melakukan debugging, penulis menemukan bahwa ada beberapa atribut yang tidak sesuai dengan yang diminta oleh APInya, sehingga menghasilkan *null*.
3. Dalam menggunakan Library, kita sebaiknya memperhatikan *implementationnya* di build.gradle (app) agar tidak keliru atau terdapat *error*. Selain itu, ada baiknya kita memperhatikan apakah library tersebut sudah *support* dengan *gradle* kita atau belum, contohnya dengan androidx

### 4.2 Saran

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

1. Memperbaiki tampilan list pada halaman utama karena masih terlihat tidak rapi
2. Menambahkan image yang tidak membosankan pada skeleton ketika image sedang di load

# 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>

<https://github.com/MagicMicky/FreemiumLibrary/wiki/Import-the-library-in-Android-Studio>

<https://stackoverflow.com/questions/44493378/whats-the-difference-between-implementation-and-compile-in-gradle>

<https://www.youtube.com/watch?v=8VX3f4AWUoI>

<http://miguelcatalan.info/2015/09/23/MaterialSearchView/>

[https://en.wikipedia.org/wiki/Singleton\\_pattern](https://en.wikipedia.org/wiki/Singleton_pattern)

## Pembuatan aplikasi:

Bekerja sama dengan tim Adrian Paceli

## LAMPIRAN