

# **Tugas Besar Pemrograman Pada Perangkat Bergerak**

## **“Flying Plane”**



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, 02 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>5</b>
<b>BAB I</b>	<b>6</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	7
<b>BAB II</b>	<b>8</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:	9
2.3.2 Kekurangan Android	10
2.4 Android SDK	11
2.5 Android Virtual device	11
2.6 Java Development Kit	11
2.7 Android Studio	11
<b>BAB III</b>	<b>12</b>
<b>IMPLEMENTASI DAN PEMBAHASAN</b>	<b>12</b>
3.1 Latar Belakang Aplikasi	12
3.2 Tampilan Aplikasi	12
3.2.1 MainActivity	13
3.2.2 Dialog MenuFragment	14
3.2.3 Dialog PauseFragment	15
3.2.4 Dialog GameOverFragment	16
3.3 Cara Menggunakan Aplikasi	17
3.4 Diagram Kelas	17
3.5 Penjelasan berbagai Kelas, Atribut, dan Method	21
3.6 Definisi Web Service dan Penggunaannya	32
3.6.1 Definisi Web Service	32
3.6.2 Penggunaan Web Service	32

<b>BAB IV</b>	<b>33</b>
<b>PENUTUP</b>	<b>33</b>
4.1 Kesimpulan	33
4.2 Saran	33
<b>DAFTAR PUSTAKA</b>	<b>34</b>
<b>LAMPIRAN</b>	<b>36</b>

## DAFTAR GAMBAR

GAMBAR 3.2.1 TAMPILAN MAINACTIVITY .....	13
GAMBAR 3.2.2 TAMPILAN HALAMAN MENUFRAGMENT .....	14
GAMBAR 3.2.3 TAMPILAN HALAMAN PAUSEFRAGMENT .....	15
GAMBAR 3.2.4 TAMPILAN HALAMAN GAMEOVERFRAGMENT .....	16
GAMBAR 3.4.1 ILUSTRASI MVVM .....	17
GAMBAR 3.4.2 KELAS BOMB .....	18
GAMBAR 3.4.3 KELAS ENEMYPLANE .....	18
GAMBAR 3.4.4 KELAS FUEL .....	18
GAMBAR 3.4.5 KELAS REWARD .....	19
GAMBAR 3.4.6 KELAS GAMEOBJECT .....	19
GAMBAR 3.4.7 KELAS FRAGMENTLISTENER .....	19
GAMBAR 3.4.8 KELAS MYASYNCTASK .....	19
GAMBAR 3.4.9 KELAS MAINACTIVITY .....	20
GAMBAR 3.4.10 KELAS MENUFRAGMENT .....	20
GAMBAR 3.4.11 KELAS PAUSEFRAGMENT .....	20
GAMBAR 3.4.12 KELAS GAMEOVERFRAGMENT .....	21
GAMBAR 3.4.13 KELAS DIAGRAM KESELURUHAN .....	21



## **BAB I**

### **PENDAHULUAN**

#### **1.1 Latar Belakang**

Game telah menjadi salah-satu hal umum di dalam aktivitas keseharian masyarakat. Terdapat banyak platform game yang berkembang seperti, PC games, console games, dan mobile games. Pada awalnya game hanya dimainkan pada perangkat PC dan console, namun dewasa ini perkembangan games yang paling populer di dunia digital adalah mobile game.

Mobile game adalah game yang dimainkan pada perangkat mobile devices, seperti PDA, smartphone, tablet PCs, dan portable media player. Sekarang ini, mobile game telah dibuat di berbagai macam sistem operasi seperti Symbian, Apple iOS, Android serta Windows Phone. Salah satu kelebihan mobile game adalah pengguna dapat bermain game dimana saja, dengan ketentuan mobile devices yang digunakan memiliki kemampuan untuk menjalankan mobile games yang ingin dimainkan. Hal ini membuat mobile game sangat mudah untuk dimainkan dibandingkan perangkat game lainnya.

Pada tugas ini kami mengembangkan mobile game sederhana untuk perangkat berbasis Android. Android telah menjadi sistem operasi mobile yang banyak diminati pengguna, hal ini dibuktikan dari riset yang dilakukan NetMarketShare. Berdasarkan data mereka hingga akhir Desember 2017, Android memiliki market share sebesar 68,39%. Di peringkat kedua terdapat iOS dengan pangsa pasar sebesar 30,56%.

#### **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 dan karakter game yang berbasis 2D.
3. Pada saat ini game yang dikembangkan hanya digunakan dalam ruang lingkup perkuliahan, namun masih banyak potensi untuk dikembangkan dan dipasarkan.

#### **1.3 Tujuan dan Manfaat**

Tujuan dari aplikasi ini adalah mengembangkan aplikasi game plane 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 game berbasis android sebagai sarana hiburan.
2. Melatih kemampuan untuk lebih memahami pengembangan game 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 **MenuFragment**, tampilan Fragment **PauseFragment**, tampilan Fragment **GameOverFragment**, diagram kelas, penjelasan kelas-kelas, penjelasan atribut-atribut, dan penjelasan method-method.

#### 3.1 Latar Belakang Aplikasi

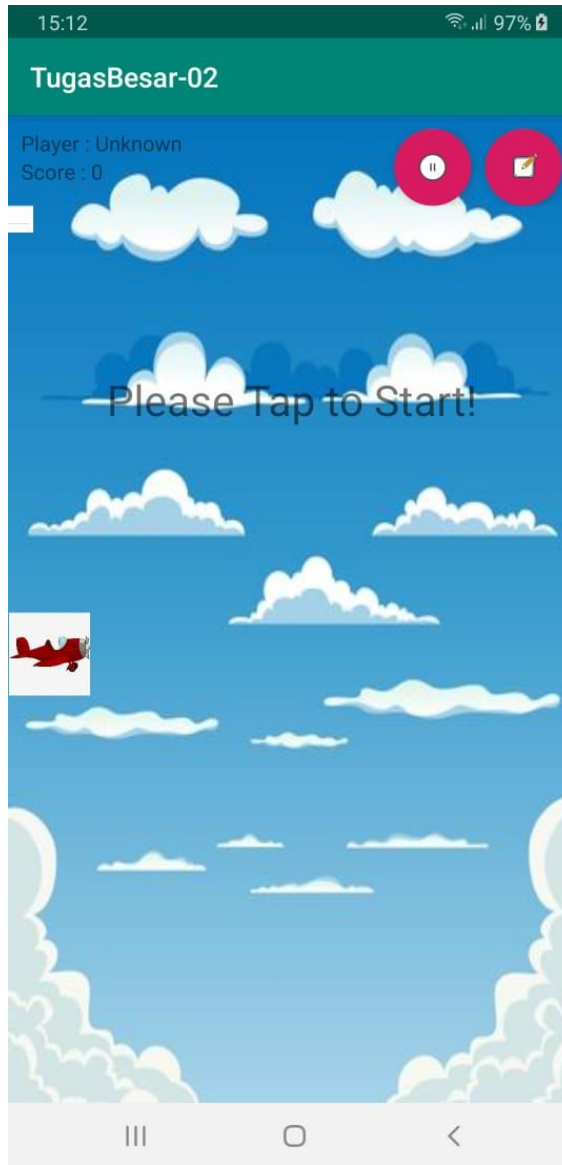
Aplikasi yang dibuat merupakan mobile-game sederhana dimana player harus menggerakkan objek pesawat dengan cara menekan layar untuk membuat objek terbang lebih tinggi dan tidak menekan apa-apa untuk membuat objek terbang lebih rendah. Player akan berusaha menggerakkan objek pesawat agar tidak bertabrakan dengan objek bomb ataupun objek pesawat lainnya. Jika objek pesawat yang dikontrol oleh player bertabrakan dengan objek bomb atau objek pesawat lainnya maka score player akan dikurangi berdasarkan objek yang 'ditabrak' dan game akan berakhir. Player harus mengumpulkan score sebanyak banyaknya untuk meraih high score. Score dapat diperoleh oleh user dengan cara menabrakan objek pesawat yang dikendalikan user dengan objek 'gift' atau 'fuel'. Score akan bertambah sesuai dengan objek yang diambil/ditabrakkan.

#### 3.2 Tampilan Aplikasi

Aplikasi kami terdiri dari satu kelas main yaitu kelas **MainActivity** yang juga merupakan halaman utama dimana game dimainkan dan tiga buah fragment. Pada kelas **MainActivity** terdapat **FloatingActionButton** bergambar 'rumah' untuk menampilkan Fragment **MenuFragment**, **FloatingActionButton** bergambar 'play' untuk menampilkan Fragment **PauseFragment**, nama player yang sedang bermain dan score sementara permainan, serta **ImageView** untuk menaruh objek-objek game yang dimainkan. Fragment **MenuFragment** berfungsi sebagai Fragment yang menampilkan form untuk memasukkan nama player, tombol start untuk memulai permainan, dan tombol exit untuk menutup aplikasi. Fragment **PauseFragment** berfungsi sebagai Fragment untuk menampilkan status game 'paused' (permainan sementara dihentikan), dan tombol 'unpause' untuk kembali melanjutkan game; dan Fragment **GameOverFragment** berfungsi sebagai Fragment untuk menampilkan high score, hasil perhitungan score dan nama player yang berada di **MainActivity**, dan tombol 'play again' untuk bermain kembali.

Sementara itu untuk gambar-gambar objek dan icon, kami melakukan pencarian di halaman website [www.images.google.com](http://www.images.google.com) lalu melakukan pemilihan pada gambar yang menurut kami bagus (sumber tertera pada daftar pustaka). Gameplay untuk permainan ini terinspirasi dari Game "Flappy Bird", dengan melakukan beberapa perubahan pada permainan ini

### 3.2.1 MainActivity

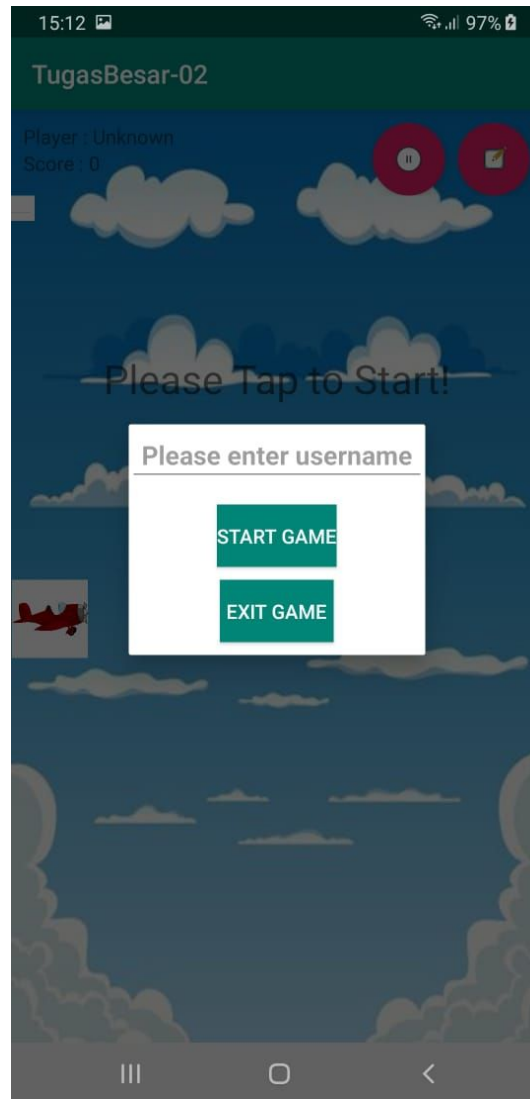


**Gambar 3.2.1** MainActivity

Pada layout ini terdapat View:

1. TextView "Player : Unknown" sebagai nama player yang sedang bermain
2. TextView "Score : 0" sebagai score sementara
3. Floating Button "PAUSE (play)" untuk membuka fragment PauseFragment
4. Floating Button "MENU (rumah)" untuk membuka fragment MenuFragment
5. TextView "Please Tap to Start !" untuk memulai permainan
6. FrameLayout untuk menaruh objek-objek game kedalam canvas
7. ImageView sebagai objek-objek game seperti pesawat, fuel, gift, dan lain-lain.

### 3.2.2 Dialog MenuFragment

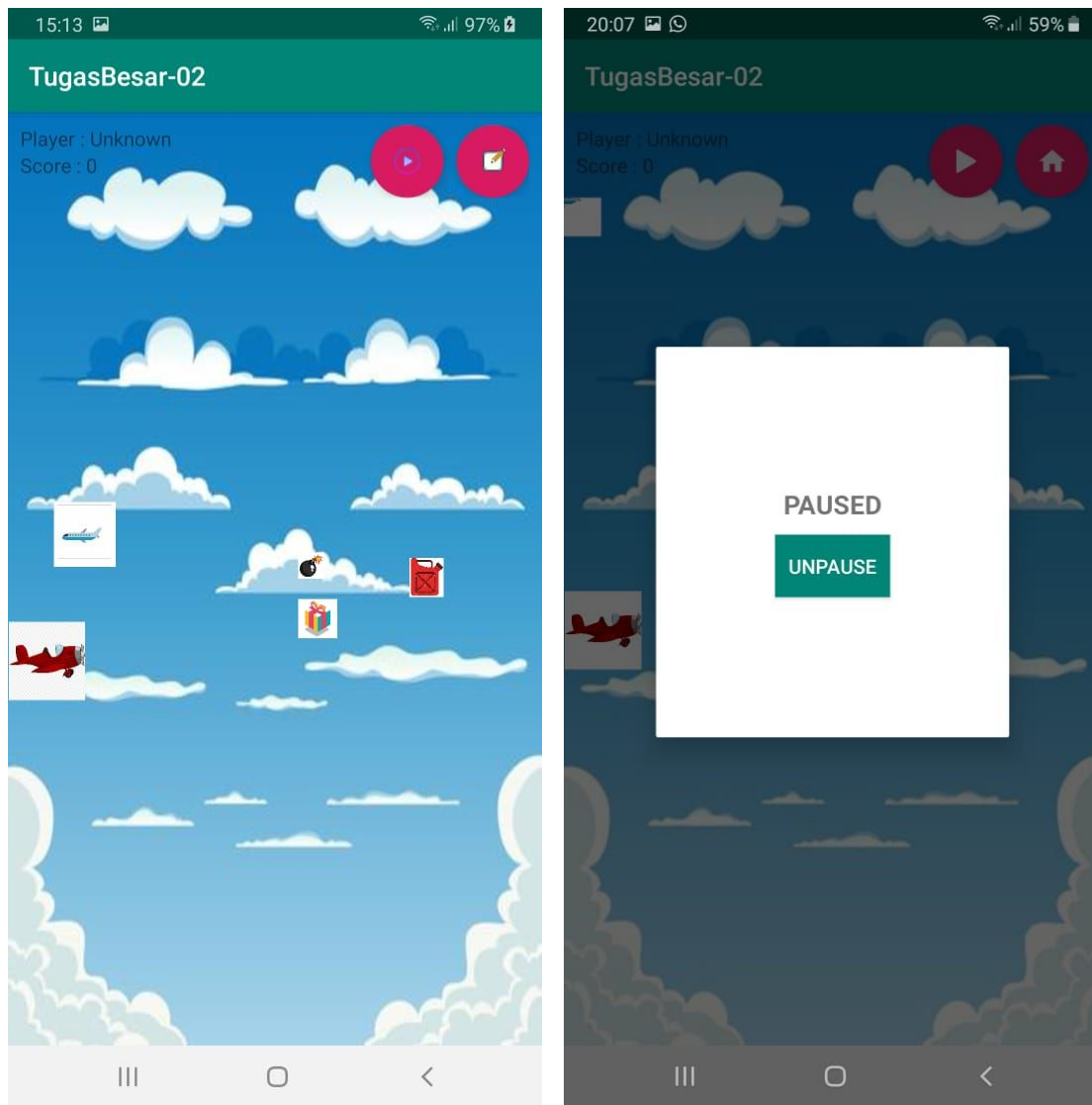


Gambar 3.2.2 Halaman MenuFragment

Tampilan pada Gambar 3.2.2 merupakan tampilan saat user telah menekan floating tombol “menu (rumah)” pada halaman utama. Tampilan ini bertujuan untuk menampilkan dialog form untuk menginput nama player dan memulai game ataupun keluar dari game. Pada Dialog ini terdapat View:

1. EditText untuk user memasukan nama player dan ditampilkan pada halaman utama
2. Button “START” untuk menutup dialog dan memulai permainan
3. Button “EXIT” untuk keluar dari permainan

### 3.2.3 Dialog PauseFragment



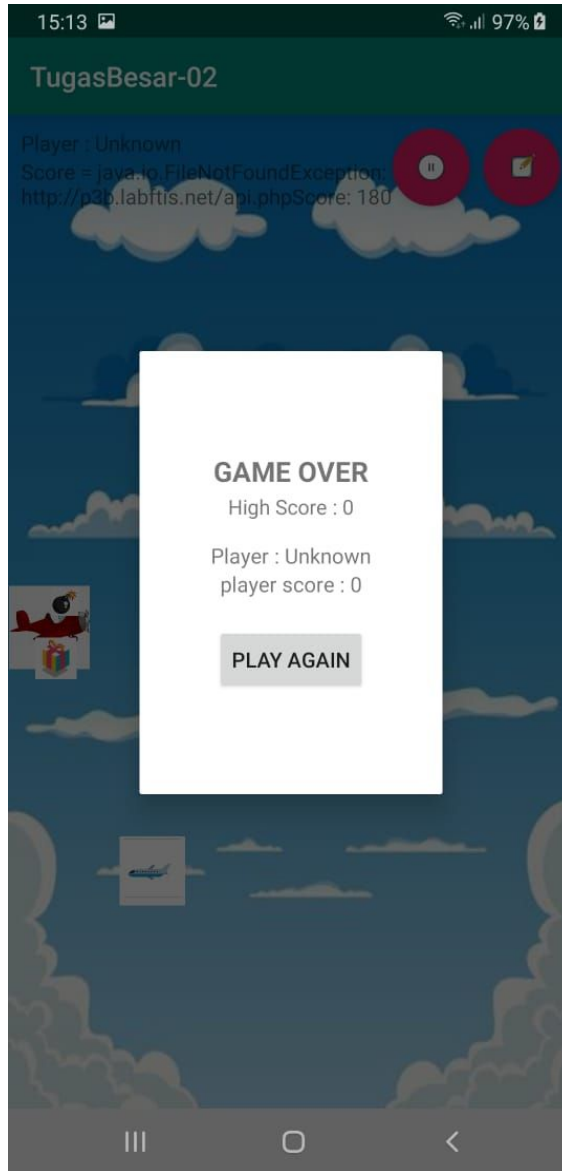
Gambar 3.2.3 Halaman Pause

Tampilan pada Gambar 3.2.3 merupakan tampilan saat user telah menekan tombol "PAUSE (play)" pada halaman utama. Tampilan ini bertujuan untuk menampilkan dialog yang berisikan status permainan dan memulai kembali permainan. Pada dialog ini terdapat View:

1. TextView "PAUSED" sebagai keterangan status game yang dimainkan sedang dihentikan sementara oleh player
2. Button "UNPAUSE" untuk menutup dialog yang sedang ditampilkan dan memulai kembali permainan



### 3.2.4 Dialog GameOverFragment



Gambar 3.2.4 GameOverFragment

Tampilan pada Gambar 3.2.4 merupakan tampilan saat objek pesawat yang digerakan oleh player telah menabrak objek pesawat lain atau objek bomb. Pada tampilan ini terdapat View:

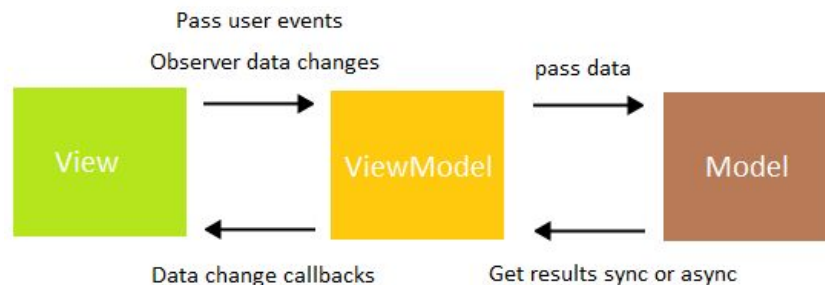
1. TextView "GAME OVER" sebagai keterangan status game yang dimainkan telah berakhir
2. TextView "High Score : " untuk menampilkan score tertinggi yang pernah didapatkan oleh user
3. TextView "Player : " untuk menampilkan nama player yang bermain
4. TextView "Player Score : " untuk menampilkan score terakhir yang didapatkan oleh player
5. Button "PLAY AGAIN" : untuk memulai permainan baru

### 3.3 Cara Menggunakan Aplikasi

1. Buka aplikasi
2. Tekan *floating button* 'MENU (rumah)' untuk menampilkan menu
3. Input nama player pada form yang berada di tampilan
4. Tekan tombol 'START' untuk menutup pop-up menu
5. Tap pada layar untuk memulai permainan
6. Hindari objek pesawat lain dan bomb
7. Ambil objek gift atau fuel untuk menambah score
8. Tekan floating button 'PAUSE' untuk menampilkan pop-up pause dan menghentikan game sementara
9. Tekan floating button 'UNPAUSE' untuk melanjutkan game
10. Game akan berakhir jika objek pesawat yang dikendalikan user menabrak objek pesawat lain atau bomb, lalu akan menampilkan pop-up 'game over'
11. Tekan tombol "PLAY AGAIN" untuk memulai permainan baru
12. Tekan *floating button* 'MENU (rumah)' untuk menampilkan menu
13. Pilih "Exit" untuk keluar dari aplikasi

### 3.4 Diagram Kelas

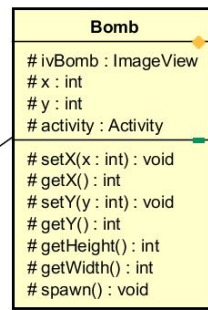
Aplikasi kami menggunakan jenis arsitektur desain kelas MVVM (Model View View-Model), dengan View yang akan menangani *user interaction*, ViewModel akan melakukan *passing* input yang diberikan oleh user pada View ke Model maupun output yang diberikan Model untuk ditampilkan ke user di View, dan Model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



Gambar 3.4.1 Ilustrasi MVVM

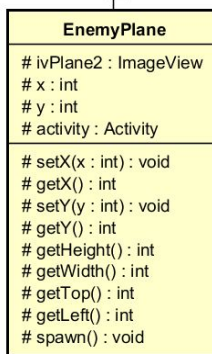
1. Kelas yang berfungsi sebagai Model

a. Bomb.java



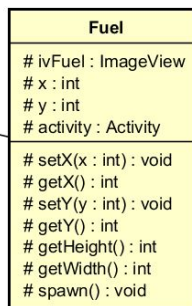
Gambar 3.4.2 Bomb.java

b. EnemyPlane.java



Gambar 3.4.3 EnemyPlane.java

c. Fuel.java



Gambar 3.4.4 Fuel.java

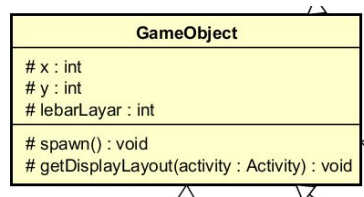
d. Reward.java



Gambar 3.4.5 Reward.java

2. Kelas yang berfungsi sebagai View-Model

a. GameObject.java



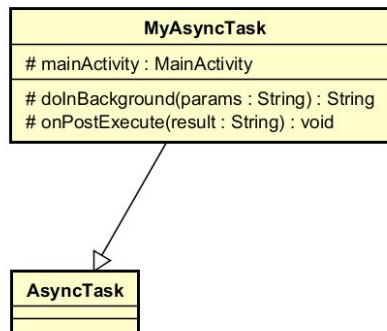
Gambar 3.4.6 GameObject.java

b. FragmentListener.java



Gambar 3.4.7 FragmentListener.java

c. MyAsyncTask.java



Gambar 3.4.8 MyAsyncTask.java

3. Kelas yang berfungsi sebagai View
  - a. MainActivity.java

MainActivity
# fabPause : FloatingActionButton # fabMenu : FloatingActionButton # tvPlayerName : TextView # tvScore : TextView # tvWelcome : TextView # ivPlane : ImageView # flCanvas : FrameLayout # handler : Handler # timer : Timer # enemyPlane : EnemyPlane # bomb : Bomb # fuel : Fuel # reward : Reward # requester : MyAsyncTask # url : 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 + gameSelect() : void + getScore() : int

Gambar 3.4.9 MainActivity.java

- b. MenuFragment.java

MenuFragment
# etUsername : EditText # btnStart : Button # btnExit : Button # listener : FragmentListener  + newInstance() : MenuFragment + onCreateView(inflater : LayoutInflater, container : ViewGroup, savedInstanceState : Bundle) : View + onClick(view : View) : void

Gambar 3.4.10 MenuFragment.java

- c. PauseFragment.java

PauseFragment
# btnUnpause : Button  + newInstance() : PauseFragment + onCreateView(inflater : LayoutInflater, container : ViewGroup, savedInstanceState : Bundle) : View + onClick(view : View) : void

Gambar 3.4.11 PauseFragment.java




\_\_\_\_\_

Atribut	Penjelasan
fabPause	Atribut untuk floating button pause
fabMenu	Atribut untuk floating button pause
tvPlayerName	Atribut TextView untuk menampilkan player name
tvScore	Atribut TextView untuk menampilkan score
tvWelcome	Atribut TextView untuk menampilkan welcome text
ivPlane	Atribut ImageView untuk menampilkan objek pesawat
flCanvas	Atribut FrameLayout yang berfungsi sebagai canvas untuk menampilkan objek
handler	Atribut untuk mengakses kelas Handler dari java library
timer	Atribut untuk mengakses kelas Timer di java library
enemyPlane	Atribut untuk mengakses kelas EnemyPlane
bomb	Atribut untuk mengakses kelas Bomb
fuel	Atribut untuk mengakses kelas Fuel
reward	Atribut untuk mengakses kelas Reward
requester	Atribut untuk mengakses kelas MyAsyncTask
url	Atribut untuk mendapatkan url untuk mengakses web service

statusButtonPause	Atribut untuk set button pause
planeHeight	Atribut untuk mendapatkan tinggi objek plane
canvasHeight	Atribut untuk mendapatkan tinggi canvas
planeY	Atribut untuk mendapatkan pergerakan objek plane
action_flg	Atribut untuk mengetahui status layar sedang disentuh atau tidak
activityStart	Atribut untuk mengetahui status activity
score	Atribut untuk mendapatkan nilai score yang didapatkan player
<b>Method</b>	<b>Penjelasan</b>
onCreate	Method untuk menghubungkan atribut di kelas dengan atribut di layout
changePosition	Method yang mengatur pergerakan objek di layout
canvasLimit	Method untuk mengatur pergerakan objek ivPlane agar tidak keluar dari layout
hitStatus	Method yang mengatur permainan ketika objek ivPlane mengenai objek lainnya
onTouchEvent	Method ini untuk mengatur pergerakan setiap objek melalui kelas Handler dan Timer



onClick	Method ini untuk mengatur permainan ketika floating action button “pause” ditekan
updateScore	Method untuk mengubah nilai atribut score
gameSelesai	Method untuk mengirimkan score ke server via web service
getScore	Method untuk mendapatkan nilai atribut score
<b>GameObject</b>	
Kelas GameObject berfungsi sebagai superclass yang mengindikasikan objek yang ada	
<b>Atribut</b>	<b>Penjelasan</b>
x	Atribut untuk mengetahui posisi objek pada sumbu-x
y	Atribut untuk mengetahui posisi objek pada sumbu-y
lebarLayar	Atribut untuk mengetahui ukuran lebar layout canvas yang disediakan
tinggiLayar	Atribut untuk mengetahui ukuran tinggi layout canvas yang disediakan
<b>Method</b>	<b>Penjelasan</b>
spawn	Method kosong untuk menampilkan objek di layout
getDisplayLayout	Method untuk mengukur ukuran canvas agar objek tidak muncul di luar canvas, atau objek muncul separuh
<b>Bomb</b>	

Kelas Bomb berfungsi sebagai kelas yang merepresentasikan objek bomb	
<b>Atribut</b>	<b>Penjelasan</b>
ivBomb	Atribut untuk ImageView objek bomb
x	Atribut untuk mengetahui posisi objek pada sumbu-x
y	Atribut untuk mengetahui posisi objek pada sumbu-y
activity	Atribut untuk mengetahui Activity yang sekarang sedang berjalan
<b>Method</b>	<b>Penjelasan</b>
setX	Method untuk mengubah posisi sumbu-x dari objek bomb
getX	Method untuk mendapatkan posisi sumbu-x dari objek bomb
setY	Method untuk mengubah posisi sumbu-y dari objek bomb
getY	Method untuk mendapatkan posisi sumbu-y dari objek bomb
getHeight	Method untuk mendapatkan tinggi dari objek bomb
getWidth	Method untuk mendapatkan lebar tinggi dari objek bomb
spawn	Method untuk menampilkan objek di layout canvas
<b>EnemyPlane</b>	

Kelas EnemyPlane berfungsi sebagai kelas yang merepresentasikan objek plane2	
<b>Atribut</b>	<b>Penjelasan</b>
ivPlane2	Atribut untuk ImageView objek plane2
x	Atribut untuk mengetahui posisi objek plane2 pada sumbu-x
y	Atribut untuk mengetahui posisi objek plane2 pada sumbu-y
activity	Atribut untuk mengetahui Activity yang sekarang sedang berjalan
<b>Method</b>	<b>Penjelasan</b>
setX	Method untuk mengubah posisi sumbu-x dari objek plane2
getX	Method untuk mendapatkan posisi sumbu-x dari objek plane2
setY	Method untuk mengubah posisi sumbu-y dari objek plane2
getY	Method untuk mendapatkan posisi sumbu-y dari objek plane2
getHeight	Method untuk mendapatkan tinggi dari objek plane2
getWidth	Method untuk mendapatkan lebar dari objek plane2
getTop	Method untuk mendapatkan panjang bagian atas dari objek plane2
getLeft	Method untuk mendapatkan panjang bagian kiri dari objek plane2

spawn	Method untuk menampilkan objek di layout
<b>Fuel</b>	
Kelas Fuel berfungsi sebagai sebagai kelas yang merepresentasikan objek fuel	
<b>Atribut</b>	<b>Penjelasan</b>
ivFuel	Atribut untuk ImageView objek fuel
x	Atribut untuk mengetahui posisi objek fuel pada sumbu-x
y	Atribut untuk mengetahui posisi objek fuel pada sumbu-y
activity	Atribut untuk mengetahui Activity yang sekarang sedang berjalan
<b>Method</b>	<b>Penjelasan</b>
setX	Method untuk mengubah posisi sumbu-x dari objek fuel
getX	Method untuk mendapatkan posisi sumbu-x dari objek fuel
setY	Method untuk mengubah posisi sumbu-y dari objek fuel
getY	Method untuk mendapatkan posisi sumbu-y dari objek fuel
getHeight	Method untuk mendapatkan tinggi dari objek fuel
getWidth	Method untuk mendapatkan lebar dari objek fuel

spawn	Method untuk menampilkan objek di layout
<b>Reward</b>	
Kelas Reward ini berfungsi sebagai kelas yang merepresentasikan objek reward	
<b>Atribut</b>	<b>Penjelasan</b>
ivReward	Atribut untuk ImageView objek reward
x	Atribut untuk mengetahui posisi objek gift pada sumbu-x
y	Atribut untuk mengetahui posisi objek gift pada sumbu-y
activity	Atribut untuk mengetahui Activity yang sekarang sedang berjalan
<b>Method</b>	<b>Penjelasan</b>
setX	Method untuk mengubah posisi sumbu-x dari objek reward
getX	Method untuk mendapatkan posisi sumbu-x dari objek reward
setY	Method untuk mengubah posisi sumbu-y dari objek reward
getY	Method untuk mendapatkan posisi sumbu-y dari objek reward
getHeight	Method untuk mendapatkan tinggi dari objek reward
getWidth	Method untuk mendapatkan lebar dari objek reward

spawn	Method untuk menampilkan objek di layout
<b>MyAsyncTask</b>	
Kelas MyAsyncTask ini berfungsi sebagai kelas yang merepresentasikan thread	
<b>Atribut</b>	<b>Penjelasan</b>
mainActivity	Atribut untuk mengakses kelas MainActivity
<b>Method</b>	<b>Penjelasan</b>
doInBackground	Menerima parameter dengan type data sesuai Type1 dan mengembalikan return value sesuai type Type3.
onPostExecute	Method ini menerima parameter dengan type data sesuai Type3. Parameter ini adalah return value dari method doInBackground dan digunakan untuk memberi respon ke pengguna bahwa proses sudah selesai.
<b>PauseFragment</b>	
Kelas PauseFragment ini berfungsi sebagai kelas yang merepresentasikan dialog fragment ketika player memberi jeda pada game/ melakukan pause game	
<b>Atribut</b>	<b>Penjelasan</b>
btnUnpause	Atribut untuk floating button pause
<b>Method</b>	<b>Penjelasan</b>

onCreateView	Method untuk menghubungkan atribut di kelas dengan atribut di layout
onClick	Method untuk mengeksekusi perintah setelah tombol tertentu ditekan
<b>MenuFragment</b>	
Kelas MenuFragement ini berfungsi sebagai kelas yang merepresentasikan fragment ketika floating button menu ditekan	
<b>Atribut</b>	<b>Penjelasan</b>
etUsername	Atribut untuk EditText username
btnStart	Atribut untuk Button start
btnExit	Atribut untuk Button exit
listener	Atribut yang digunakan untuk mengakses interface FragmentListener
<b>Method</b>	<b>Penjelasan</b>
newInstance	Method untuk menginisialisasi Fragment Menu yang diwakili oleh kelas ini
onCreateView	Method untuk menghubungkan atribut di kelas dengan atribut di layout
onClick	Method untuk mengeksekusi perintah setelah tombol tertentu ditekan
<b>GameOverFragment</b>	

Kelas GameOverFragment ini berfungsi sebagai kelas yang merepresentasikan dialog fragment game over ketika game telah selesai	
<b>Atribut</b>	<b>Penjelasan</b>
tvHighScore	Atribut untuk TextView high score
tvPlayerName	Atribut untuk TextView player name
tvPlayerScore	Atribut untuk TextView player score
btnPlayAgain	Atribut untuk Button play again
activity	Atribut untuk mengetahui Activity yang sekarang sedang berjalan
<b>Method</b>	<b>Penjelasan</b>
newInstance	Method untuk menginisialisasi Fragment kelas yang diwakili oleh objek Class ini
onCreateView	Method untuk menghubungkan atribut di kelas dengan atribut di layout
onClick	Method untuk mengeksekusi perintah setelah tombol tertentu ditekan
closeApp	Method untuk menutup aplikasi
updateScore	Method untuk mengganti atribut score
<b>FragmentManager</b>	
Kelas FragmentListener ini berfungsi sebagai interface yang mengatur pergantian antar-fragment	



Method	Penjelasan
closeApp	Method untuk menutup aplikasi
updateScore	Method untuk mengganti atribut score

### 3.6 Definisi Web Service dan Penggunaannya

#### 3.6.1 Definisi Web Service

Web service adalah suatu sistem perangkat lunak yang dirancang untuk mendukung interoperabilitas dan interaksi antar sistem pada suatu jaringan. Web service digunakan sebagai suatu fasilitas yang disediakan oleh suatu website untuk menyediakan layanan (dalam bentuk informasi) kepada sistem lain, sehingga sistem lain dapat berinteraksi dengan sistem tersebut melalui layanan-layanan (service) yang disediakan oleh suatu sistem yang menyediakan web service. Web service menyimpan data informasi dalam format XML, sehingga data ini dapat diakses oleh sistem lain walaupun berbeda platform, sistem operasi, maupun bahasa compiler.

#### 3.6.2 Penggunaan Web Service

Method gameSelesai()

Method ini digunakan untuk mengirimkan score ke server via web service dengan cara menginisialisasi kelas `MyAsyncTask`, lalu memanggil method `execute` yang dimiliki kelas `MyAsyncTask` untuk mengirimkan nilai dari atribut `score` ke dalam url yang telah ditentukan.

## **BAB IV**

### **PENUTUP**

#### **4.1 Kesimpulan**

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

1. Game ini cukup mudah dimainkan karena player hanya perlu menekan layar untuk membuat objek pesawat terbang lebih tinggi dan tidak menekan layar untuk membuat pesawat terbang lebih rendah dengan tujuan mengambil objek yang menambah score dan menghindar dari objek yang akan membuat game berakhir.
2. Dengan teknologi android aplikasi ini mampu mengimplementasikan ImageView , Pop-Up, Thread dan Web-Service

#### **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. Fitur dasar yang masih memiliki masalah (bug) secepatnya diperbaiki

## 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>

### Pembuatan aplikasi:

Plane Image:

[https://www.google.com/search?tbm=isch&sxsr=ACYBGNQwyt8UkOyn0ljPy1hxx7k9zqDBYA:1572970749605&q=plane&chips=q:plane,g\\_1:cartoon:4Xy0Gb18jAk%3D&usg=AI4\\_-kSM8q0MLbTkgJw9yPOJQ4UeGuyAHg&sa=X&ved=0ahUKEwi1upbWvNPIAhUaVH0KHc6GAAEQ4IYIKigA&biw=756&bih=635&dpr=0.9](https://www.google.com/search?tbm=isch&sxsr=ACYBGNQwyt8UkOyn0ljPy1hxx7k9zqDBYA:1572970749605&q=plane&chips=q:plane,g_1:cartoon:4Xy0Gb18jAk%3D&usg=AI4_-kSM8q0MLbTkgJw9yPOJQ4UeGuyAHg&sa=X&ved=0ahUKEwi1upbWvNPIAhUaVH0KHc6GAAEQ4IYIKigA&biw=756&bih=635&dpr=0.9)

Bomb Image:

[https://www.google.com/url?sa=i&source=images&cd=&cad=rja&uact=8&ved=2ahUKEwiR7PaEvdPIAhUafisKHTrXBaUQjRx6BAgBEAQ&url=%2Furl%3Fsa%3Di%26source%3Dimages%26cd%3D%26ved%3D%26url%3Dhttps%253A%252F%252Fwww.vectorstock.com%252Froyalty-free-vector%252Fa-classic-cartoon-style-black-round-bomb-lit-on-vector-20631273%26psig%3DAOvVaw2GbxQ\\_P7NXLbc750m23V73%26ust%3D1573057223907203&psig=AOvVaw2GbxQ\\_P7NXLbc750m23V73&ust=1573057223907203](https://www.google.com/url?sa=i&source=images&cd=&cad=rja&uact=8&ved=2ahUKEwiR7PaEvdPIAhUafisKHTrXBaUQjRx6BAgBEAQ&url=%2Furl%3Fsa%3Di%26source%3Dimages%26cd%3D%26ved%3D%26url%3Dhttps%253A%252F%252Fwww.vectorstock.com%252Froyalty-free-vector%252Fa-classic-cartoon-style-black-round-bomb-lit-on-vector-20631273%26psig%3DAOvVaw2GbxQ_P7NXLbc750m23V73%26ust%3D1573057223907203&psig=AOvVaw2GbxQ_P7NXLbc750m23V73&ust=1573057223907203)

Reward Image:

[https://www.google.com/search?biw=756&bih=635&tbm=isch&sxsr=ACYBGNSwgcNixQzT1b7wuDAmJBCM9NgdfQ%3A1572970905539&sa=1&ei=maHBXY\\_MIMX\\_rQGGp7OQDw&q=gift&oq=gift&gs\\_l=img.3..35i39j0i67l9.8036.8500..9051...0.0..0.88.260.4.....0....1..gws-wiz-img.....0.JHfnEURjVkJQ&ved=0ahUKEwjP\\_MOGvdPIAhXFfysKHYbTDPIQ4dUDCAc&uact=5](https://www.google.com/search?biw=756&bih=635&tbm=isch&sxsr=ACYBGNSwgcNixQzT1b7wuDAmJBCM9NgdfQ%3A1572970905539&sa=1&ei=maHBXY_MIMX_rQGGp7OQDw&q=gift&oq=gift&gs_l=img.3..35i39j0i67l9.8036.8500..9051...0.0..0.88.260.4.....0....1..gws-wiz-img.....0.JHfnEURjVkJQ&ved=0ahUKEwjP_MOGvdPIAhXFfysKHYbTDPIQ4dUDCAc&uact=5)

Fuel Image:

[https://www.google.com/search?tbm=isch&sxsr=ACYBGNQFGGZobaSjuV2qgOOHNHN-CiiUGg:1572970978493&q=fuel&chips=q:fuel,g\\_1:cartoon:U0YTBoemexE%3D&usg=AI4\\_-kQIKovDIm-7rhhxO659C-G-TiIF6g&sa=X&ved=0ahUKEwjg2ajDvdPIAhXNAnIKHVPFBDQ4IYILigC&biw=756&bih=635&dpr=0.9](https://www.google.com/search?tbm=isch&sxsr=ACYBGNQFGGZobaSjuV2qgOOHNHN-CiiUGg:1572970978493&q=fuel&chips=q:fuel,g_1:cartoon:U0YTBoemexE%3D&usg=AI4_-kQIKovDIm-7rhhxO659C-G-TiIF6g&sa=X&ved=0ahUKEwjg2ajDvdPIAhXNAnIKHVPFBDQ4IYILigC&biw=756&bih=635&dpr=0.9)

<https://stackoverflow.com/questions/1921514/how-to-run-a-runnable-thread-in-android-at-defined-intervals>

<https://stackoverflow.com/questions/4680499/how-to-get-the-width-and-height-of-an-android-widget-imageview>

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

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

## LAMPIRAN