

## REKAYASA PERANGKAT LUNAK PERANCANGAN DAN PEMBUATAN *AUGMENTED REALITY* *SHOOTING GAME*

Clara Isabella Menayang<sup>1)</sup>,  
Jenisa Felisa<sup>2)</sup>

STMIK LIKMI Bandung  
Jl. Ir. H. Juanda no 96 Bandung 40132

E-mail : [clara.mw2@gmail.com](mailto:clara.mw2@gmail.com)<sup>1)</sup>,  
[jenisafelisa@gmail.com](mailto:jenisafelisa@gmail.com)<sup>2)</sup>

---

### ABSTRAK

Penelitian bertujuan untuk merancang sebuah *shooting game* dengan *Augmented Reality* yang diimplementasikan pada *smartphone* dengan sistem operasi android. Proses perancangan *game* menggunakan *game engine* Unity3D. Unity3D didukung oleh berbagai *Software Development Kit* (SDK) yang dibutuhkan untuk membuat *game augmented reality* seperti salah satu contohnya *Vuforia*. Dalam penelitian disimpulkan bahwa hasil dari tugas akhir ini adalah perangkat lunak *game augmented reality* berbasis android menggunakan *game engine* Unity3D dapat dibuat dan berjalan dengan baik dengan adanya *Vuforia* sebagai SDK yang berfungsi untuk mengaktifkan kamera yang memiliki fitur melacak suatu target. Untuk melacak target dapat dilakukan secara *markerless*, dimana tidak butuh sebuah *marker*.

**Kata kunci** : *Shooting Game, Augmented Reality, Unity3D, SDK Vuforia, Markerless.*

---

### 1 PENDAHULUAN

*Augmented Reality* (AR) adalah penggabungan benda-benda nyata dan maya sehingga kita bisa melihat benda tiga dimensi (3D) muncul pada lingkungan nyata dalam AR. AR pertama kali dikembangkan di Sutherland pada tahun 1965, para peneliti memanfaatkan teknologi ini sebagai salah satu cara baru untuk meningkatkan produktifitas, efektifitas dan efisiensi serta sebagai media entertainment. Sampai saat ini AR telah banyak digunakan di dunia hiburan, pelatihan militer, medis, desain rekayasa, robotik dan telerobotik, manufaktur, pendidikan, dan lain-lain.

Beberapa tahun terakhir kemampuan perangkat *mobile* telah mengalami perkembangan yang cukup pesat dengan kehadiran beberapa sistem operasi baru yaitu Android, IOS, dan Windows Phone, dan tidak hanya software yang terus diperbaharui tetapi *hardware* pada *smartphone* pun selalu diperbaharui. Perangkat *mobile* pada saat ini memiliki kamera yang beresolusi tinggi dan didukung dengan prosesor yang cepat dengan memori yang besar. Dengan adanya *smartphone* yang mempunyai spesifikasi tinggi, memungkinkan pengimplementasian AR pada perangkat *mobile* yang lebih maksimal karena dapat memproses data lebih banyak dan lebih cepat.

Ada berbagai tipe *game* pada *smartphone*, salah satunya *game* dengan tipe *shooting game* seperti yang akan dibuat sebagai tugas akhir berikut, *shooting game* dimainkan dengan cara menembak target (hantu) yang bergerak. Pemain akan bermain sebagai seorang pemburu yang menembak target yang terbang melintas pada layar *smartphone*. Setiap tembakan yang tepat mengenai target, maka pemain akan mendapatkan poin. *Game* ini akan dibuat menggunakan *game engine* Unity.

## 2 STUDI LITERATUR

### 2.1 *Augmented Reality*

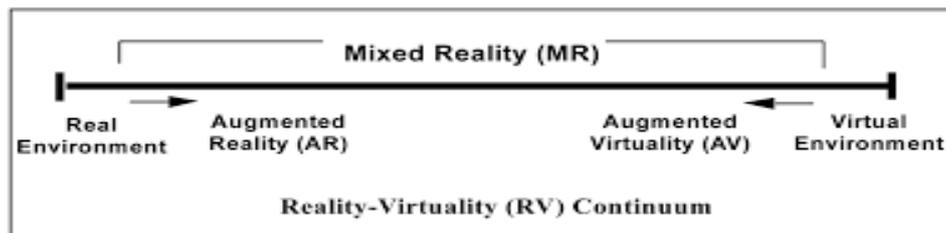
Menurut Greg Kipper dan Joseph Rampolla (Kipper, 2012 : 1), *Augmented Reality* adalah suatu sistem yang bertujuan untuk mengembangkan teknologi yang memperbolehkan penggabungan secara real-time terhadap digital *content* yang dibuat oleh komputer dengan dunia nyata. Sehingga *Augmented Reality* memungkinkan pengguna melihat dunia nyata dimana objek virtual dua dimensi maupun tiga dimensi diproyeksikan ke dalam dunia nyata.

Menurut Alan. B. Craig (Craig, 2013 : 1), *Augmented Reality* itu sendiri adalah ide yang berhubungan antara manusia dengan komputer dan sebaliknya antara komputer dengan manusia, yang tentunya dalam hal untuk mengimplementasikan *Augmented Reality* ini membutuhkan teknologi yang terkini.

*Augmented Reality* sebagai sistem yang memiliki karakteristik sebagai berikut :

- a. Menggabungkan lingkungan nyata dan virtual
- b. Berjalan secara interaktif dalam waktu nyata.
- c. Berintegrasi dalam tiga dimensi (3D)

Dalam buku Paul. A. Fishwick (Fishwick, 2008:123) menyebutkan bahwa pada tahun 1994 Paul Milgram dan Fumio Kishino mengenalkan *Mixed Reality* dapat dilihat pada Gambar 2.3. Paul Milgram dan Fumio Kishino mendeskripsikan bahwa terdapat celah yang menjadi pemisah antara lingkungan nyata dan lingkungan virtual. Diantara kedua lingkungan tersebut terdapat dua bagan yang menjadi jembatan yang memiliki kecenderungan yang berbeda. Dua bagan tersebut adalah *Augmented Reality* dan *Augmented virtuality*. Dapat dilihat dari gambar bahwa posisi kedua bagan tersebut berbeda, untuk *Augmented reality* lebih dekat dengan lingkungan nyata dan *Augmented virtuality* lebih dekat dengan lingkungan virtual.



Gambar 1 *Mixed Reality*  
(Sumber : Milgram dan Kishino, 1994 : 283)

Untuk sisi paling kiri adalah lingkungan nyata dimana maksud dari lingkungan nyata adalah dimana terdapat benda-benda nyata, sedangkan untuk sisi paling kanan adalah lingkungan maya dimana terdapat benda-benda maya atau tidak nyata seperti lingkungan yang terdapat pada animasi 3D maupun 2D. Pada bagian *Augmented Reality* lingkungan bersifat nyata dan benda bersifat maya. Pada *Augmented Virtuality* kebalikannya, dimana lingkungan bersifat maya dan benda bersifat nyata. Oleh karena itu pengelompokan AR dan AV seringkali disebut sebagai *Mixed Reality*.

Dalam pembuatan AR beberapa komponen penting yang diperlukan dalam pembuatan dan pengembangan aplikasi AR adalah sebagai berikut :

1. Komputer : berfungsi sebagai perangkat yang digunakan untuk membuat sebuah aplikasi atau yang memproses semua yang akan terjadi pada aplikasi. Kemudian hasil

yang telah dibuat / diproses melalui komputer akan ditampilkan di layar monitor atau layar ponsel.

2. *Marker*: berfungsi sebagai media gambar yang nantinya akan digunakan untuk proses tracking. Dimana sistem akan mengenali marker tersebut lalu akan menciptakan virtual objek 3D.
3. Kamera : berfungsi untuk memproses *image* yang ditangkap oleh kamera. Apabila kamera menangkap *image* yang mengandung marker, maka aplikasi yang ada di perangkat *mobile* akan mengenali marker. Lalu perangkat *mobile* akan menampilkan objek di atas marker tersebut.

Beberapa metode yang digunakan pada *Augmented Reality* diantaranya adalah *Marker Base Tracking* dan *Markerless Augmented Reality*

## 2.2 *Markerless*

Menurut Mario Vento (Vento, 2009 : 24) salah satu metode *Augmented Reality* yang saat ini sedang berkembang adalah metode *Markerless Augmented Reality*, dengan metode ini pengguna tidak perlu lagi menggunakan sebuah *marker* untuk menampilkan elemen-elemen digital. Seperti saat ini dikembangkan oleh perusahaan *Augmented Reality* terbesar di dunia *Total Immersion* dan *Qualcomm*, mereka telah membuat berbagai macam teknik *Markerless tracking* sebagai teknologi andalan mereka seperti :

1. *Face Tracking*  
Dengan menggunakan algoritma yang mereka kembangkan, komputer dapat mengenali wajah manusia secara umum dengan cara mengenali posisi mata, hidung, dan mulut manusia, kemudian akan mengabaikan objek-objek lain di sekitarnya seperti pohon, rumah, dan benda-benda lainnya.
2. *Motion Tracking*  
Pada teknik ini komputer dapat menangkap gerakan, *Motion tracking* telah mulai digunakan secara ekstensif untuk memproduksi film-film yang mencoba mensimulasikan gerakan.
3. *GPS Tracking*  
*GPS tracking* adalah teknik yang memanfaatkan fitur GPS dan kompas yang ada di dalam perangkat *mobile*. Aplikasi akan mengambil data dari GPS dan kompas kemudian akan menampilkan dalam bentuk arah yang kita inginkan secara *real time*.
4. *Device Tracking*  
Teknik ini memanfaatkan rotasi yang ada di perangkat *mobile*, maksud dari rotasi ini adalah objek berada di titik tertentu dan ketika perangkat *mobile* kita putar di sekitar lingkungan nyata, objek itu akan berada pada posisi yang di tetapkan.

## 2.3 *Vuforia*

Menurut Mario Fernando (Fernando, 2013 : 6) *Vuforia* adalah *Augmented Reality Software Development Kit (SDK)* yang digunakan pada perangkat *mobile* untuk pembuatan aplikasi *Augmented Reality*. *Software development Kit* ini menggunakan teknologi *computer vision* untuk mengenali dan melacak gambar target dan objek 3D yang sederhana secara *real time*. Kemampuan ini memungkinkan para *developer* untuk membuat posisi dan orientasi objek virtual seperti model 3D dan media-media lainnya yang berkaitan dengan dunia nyata dapat dilihat melalui kamera dari perangkat *mobile*. Objek virtual tersebut selanjutnya melacak posisi orientasi dan gambar secara *real time* sehingga pengguna dalam melihat objek tersebut melalui layar kamera perangkat *mobile* sehingga seakan virtual objek tersebut adalah bagian dari dunia nyata.

*Vuforia SDK* menggunakan beberapa tipe target, 2D dan 3D, termasuk pengaturan untuk target yang jumlahnya lebih dari satu (multi target), seperti target berbentuk silinder

untuk mendeteksi gambar yang berada pada permukaan silinder. Berikut ini beberapa kelebihan yang dimiliki Vuforia SDK :

1. Mendeteksi dengan cepat target local dengan kapasitas melacak lima target secara simultan
2. Pendeteksian dalam keadaan kurang cahaya dan bahkan ketika target tertutup sebagian
3. Kapasitas pelacakan yang tinggi yang membuat aplikasi terus melacak target dan membantu dari suatu objek yang atau bahkan ketika target tidak lagi terlihat pada kamera.

Vuforia menawarkan banyak fitur AR yang dapat dikombinasikan dalam pembuatan aplikasi antara lain :

1. *Image target*: Fitur yang menjadikan gambar sebagai target marker untuk menampilkan objek maya pada saat target marker dikenali oleh aplikasi.
2. *Cloud Recognition* : Fitur yang akan melakukan *scanning target* untuk menampilkan objek maya, dimana data dari *target marker*-nya diambil dari *cloud*.
3. *User defined target* : Fitur yang memperbolehkan pengguna mendefinisikan targetnya sendiri sesuai keinginan pengguna untuk menampilkan dimana objek mayanya akan ditampilkan.
4. *Virtual Button* : Fitur yang menjadikan objek pada dunia nyata sebagai tombol dengan pendefinisian pada *marker* dengan bentuk persegi
5. *Multi target* : Fitur yang dapat melakukan *scanning* lebih dari satu target untuk menampilkan objek maya
6. *Video playback* : Fitur yang dapat menjalankan video ketika suatu *marker* terdeteksi oleh kamera.
7. *Rotational Device* : Fitur ini memungkinkan para *developer* menciptakan aplikasi yang dapat melacak rotasi dari posisi perangkat itu berada.

Dengan banyaknya fitur, Vuforia menjadi salah satu *extension* yang banyak digunakan dalam pembuatan sebuah aplikasi *Augmented Reality*. Vuforia juga memiliki dokumentasi yang cukup lengkap serta dukungan dari forum yang aktif, sehingga memudahkan bagi para *developer* untuk mencari suatu informasi atau tutorial penggunaannya.

### 3 ANALISIS DAN PERANCANGAN PERANGKAT LUNAK

#### 3.1 Gambaran Umum Perangkat Lunak

Perangkat lunak *game* “*Ghost Hunter*” adalah perangkat lunak yang dibuat untuk bertujuan sebagai media hiburan bagi *player* yang ingin memainkannya. *Game* “*Ghost Hunter*” adalah *game first person shooter* dengan konsep menembaki objek – objek yang berada di sekitar lingkungan nyata yang berbentuk hantu.

Aplikasi ini dapat dimainkan menggunakan perangkat *smartphone* berbasis *android* yang menggunakan layar sentuh. Aplikasi “*Ghost Hunter*” ini juga menggunakan perangkat kamera pada *smartphone* itu sendiri sehingga kualitas dan resolusi gambar pada *game* tersebut tergantung oleh kamera pada *smartphone* yang dipakai.

Pada *game* “*Ghost Hunter*” terdapat beberapa fitur yaitu dapat melihat *urutan High Score* pada menu *High Score* dengan cara menekan *button High Score* pada tampilan menu utama. Selanjutnya pemain dapat melihat tutorial cara memainkan *game* dengan cara menekan menu *about* pada menu utama. Pemain juga bisa mengganti senjata pada saat *game play* dengan cara menekan *button* ganti senjata.

### 3.2 Spesifikasi Kebutuhan Perangkat Lunak

#### 3.2.1 Analisa Kebutuhan Fungsional

*Game* yang akan dibangun harus memiliki fitur seperti :

- Dapat melihat menu *score*
- Dapat melihat menu *about*
- Dapat mengganti senjata
- Dapat menempatkan posisi objek hantu secara acak,
- Dapat dimainkan pada sistem operasi Android
- Dapat mengakses kamera perangkat *mobile*

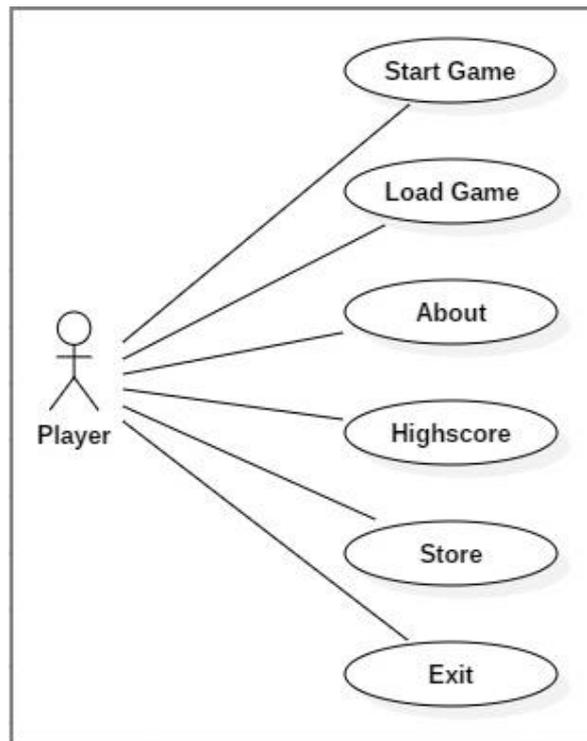
#### 3.2.2 Analisis Kebutuhan non Fungsional

Kebutuhan non fungsional yang dibutuhkan adalah sebagai berikut :

- Smartphone* dengan sistem operasi Android
- Minimum Sistem operasi *Nougat (7.0)*
- RAM 2GB
- Minimum *storage* adalah 1GB
- Kamera 3MP

### 3.3 Use Case Diagram

Diagram *use case* program aplikasi enkripsi dekripsi *file* data dapat dilihat pada gambar 2.



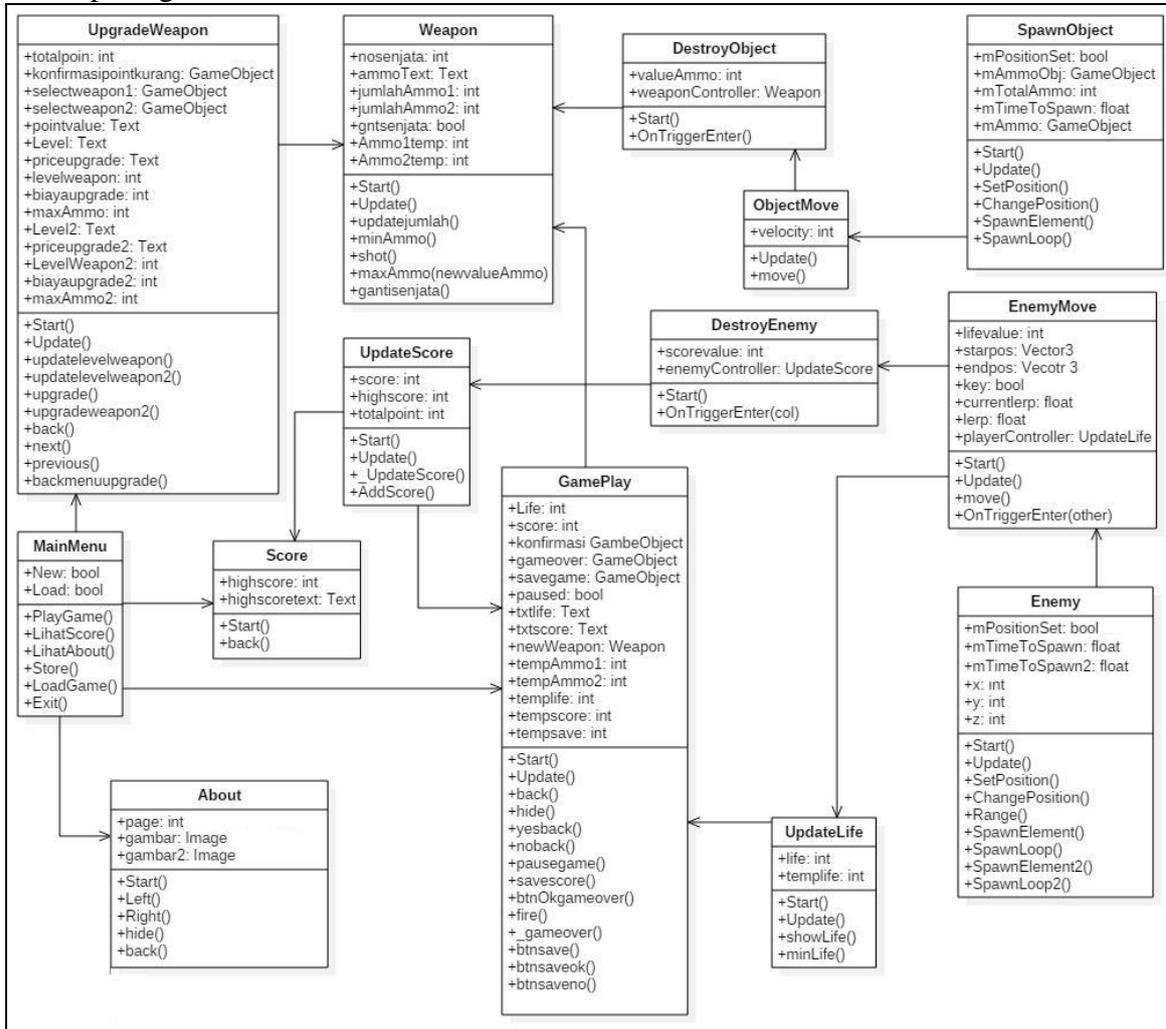
Gambar 2

*Use case diagram Augmented Reality Shooting Game*

Sistem aplikasi perangkat lunak yang dirancang memiliki enam fungsi utama (*key behaviour*) yaitu *start game*, *load game*, *about*, *highscore*, *store*, dan *exit* sehingga menjadi enam use case yang berinteraksi langsung dengan pengguna.

### 3.4 Class Diagram

Class diagram aplikasi perangkat lunak *Augmented Reality Shooting Game* dapat dilihat pada gambar 3.



Gambar 3

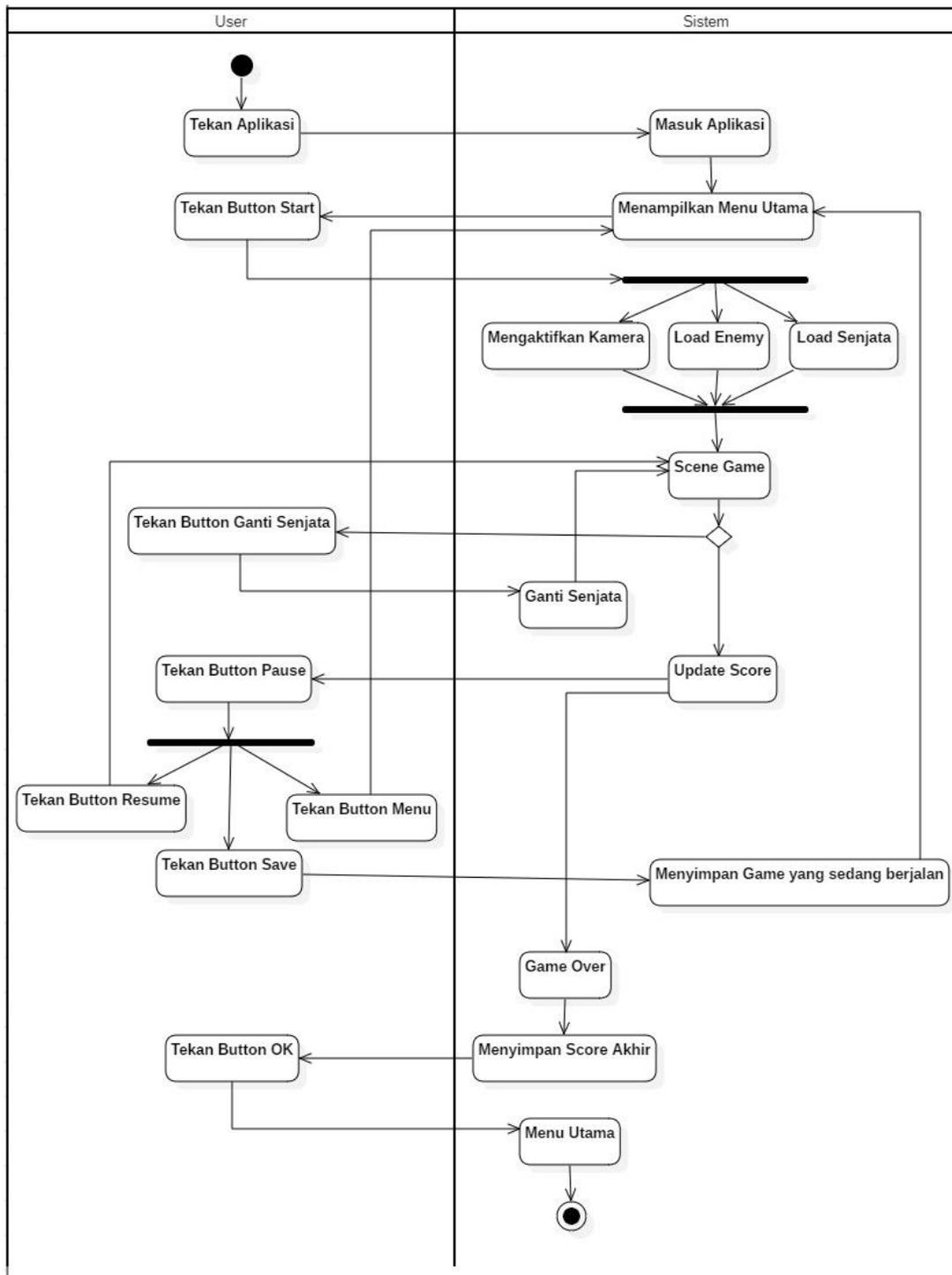
Class diagram program *Shooting Game*

### 3.5 Activity Diagram

#### 3.5.1 Activity Diagram Start Game

Gambar 4 merupakan aktivitas pada perancangan *game Ghost Hunter*, dimana *user* masuk ke aplikasi, lalu sistem akan menampilkan layar menu utama, *user* menekan tombol *start game* dan sistem akan mengaktifkan kamera perangkat android. Jika saat sistem mengaktifkan kamera terjadi error, maka sistem akan menampilkan pesan *error* lalu akan kembali ke menu utama. Apabila tidak terjadi error maka sistem akan melakukan *load enemy* dan *load senjata*, sistem akan masuk ke *scene* selanjutnya yaitu *scene game*, dimana *user* dapat memainkan permainan.

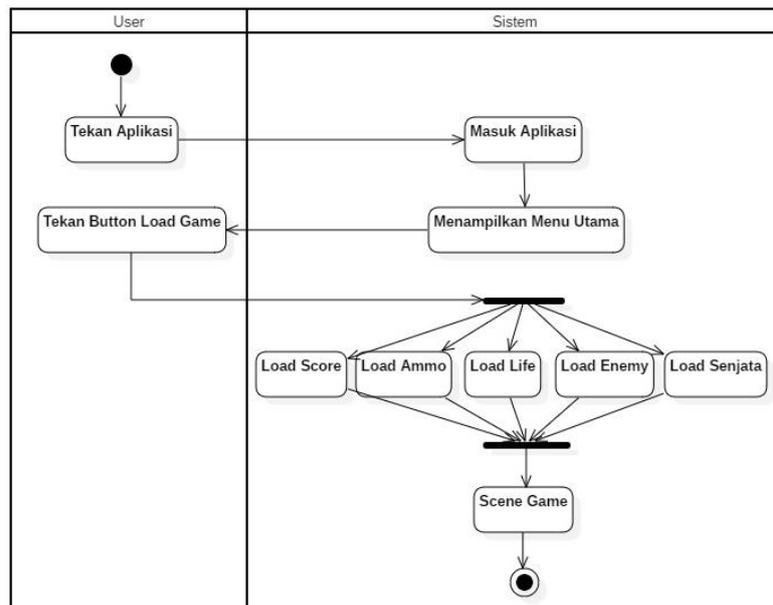
Permainan akan berakhir apabila *user* menekan *button exit*, dan kemudian ada dua pilihan jika *user* menekan “Yes” maka permainan akan berakhir dengan kembali ke menu utama dan *score* akhir akan disimpan. Jika *user* menekan “No” maka permainan dilanjutkan kembali. Permainan juga akan berakhir apabila *game over* atau *health user* habis, lalu *score* akan disimpan setelah kita mengisi nama dan menekan *button* “Yes” selanjutnya sistem akan mengembalikan tampilan ke menu utama.



Gambar 4  
Activity Diagram Start Game (Swimlane)

**3.5.2 Activity Diagram Load Game**

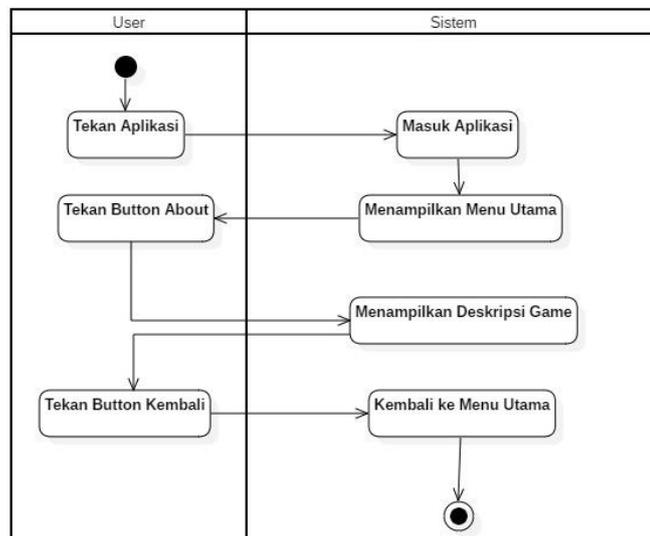
Gambar 5 merupakan diagram aktivitas untuk menampilkan menu *load game*, user menekan aplikasi lalu sistem akan masuk ke menu utama, di menu utama user menekan button *load game*, dimana menu *load game* akan memuat game yang terakhir kali kita *save*.



Gambar 5  
 Activity Diagram Load Game (Swimline)

### 3.5.3 Activity Diagram About

Gambar 6 merupakan diagram aktivitas untuk menampilkan menu *about*, *user* menekan aplikasi lalu sistem akan masuk ke menu utama, di menu utama *user* menekan *button about*, dimana *menu about* akan mendeskripsikan tentang cara memainkan *game* ini dan juga penjelasan tentang *enemy*. *User* dapat menekan *button* kembali dan sistem akan kembali ke menu utama.

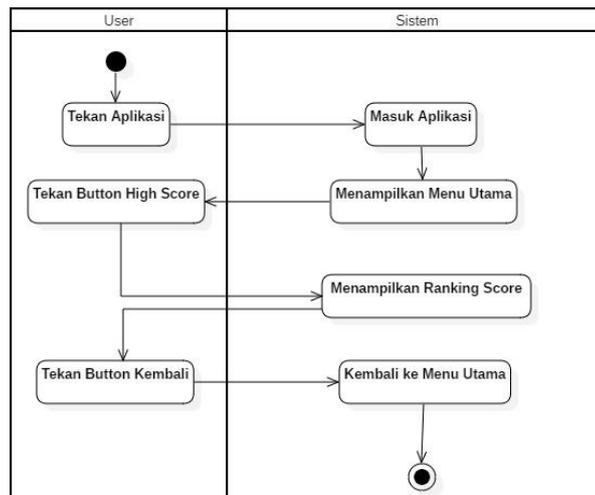


Gambar 6  
 Activity Diagram About (Swimline)

### 3.5.4 Activity Diagram HighScore

Gambar 7 merupakan diagram aktivitas untuk melihat peringkat *score*. *User* menekan aplikasi lalu sistem akan masuk ke menu utama. Di menu utama *user* dapat melihat peringkat *score* dengan cara menekan *button high score*, sistem akan pindah ke tampilan *score* dan menampilkan peringkat *score*. Ketika *user* selesai memainkan *game*,

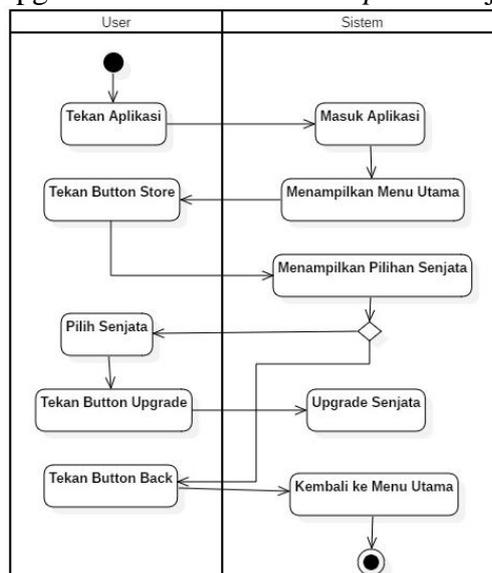
maka *score* akan disimpan, dan di menu inilah *user* dapat melihat *high score* dari permainan yang telah dimainkannya.



Gambar 7  
Activity Diagram High Score (Swimline)

### 3.5.5 Activity Diagram Store

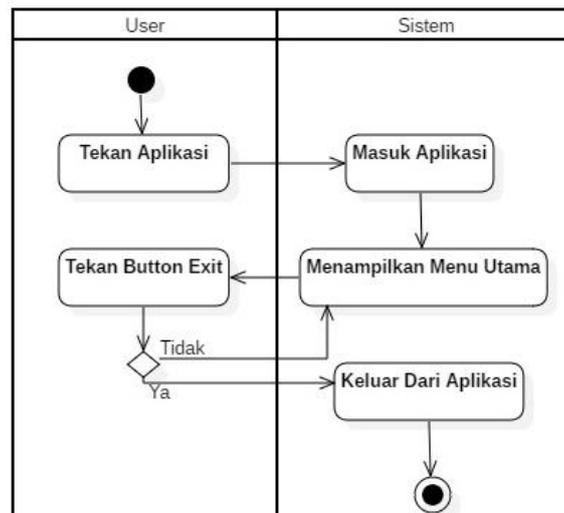
Diagram aktivitas selanjutnya adalah diagram aktivitas untuk melihat *atau membeli senjata* yang terdapat pada Gambar 8. *User* menekan aplikasi lalu sistem akan masuk ke menu utama. Di menu utama *user* dapat menekan *button store* untuk mengupgrade senjata, apabila senjata berhasil diupgrade maka sistem akan *update* senjata.



Gambar 8  
Activity Diagram Store (Swimline)

### 3.5.6 Activity Diagram Exit

Diagram aktivitas selanjutnya adalah diagram aktivitas untuk keluar dari aplikasi. Untuk keluar dari aplikasi kita hanya perlu menekan *button exit* yang ada pada menu utama. Lalu akan muncul form validasi, apabila *user* memilih ya maka aplikasi akan ditutup.



Gambar 9  
 Activity Diagram Exit (Swimlane)

## 4 IMPLEMENTASI DAN PENGUJIAN PERANGKAT LUNAK

### 4.1 Spesifikasi Kebutuhan Perangkat Keras

Spesifikasi perangkat lunak yang digunakan penulis untuk melakukan perancangan dan pengujian perangkat lunak program tertera pada tabel 1.

Tabel 1  
 Spesifikasi kebutuhan perangkat keras

Perangkat Keras	Spesifikasi
RAM	Minimum 2 GB
Memory	1 GB
Camera	Minimum 3 MP
Sistem Operasi	Android
Android Version	Nougat (7.0)
RAM	Minimum 2 GB

### 4.2 Pengujian Tampilan Antar Muka

Berikut ini adalah tampilan antar muka dalam game :

#### 1. Tampilan Menu Utama

Gambar 10 merupakan tampilan menu utama, dimana pemain dapat memilih berbagai menu yang tersedia seperti menu play game untuk memulai permainan, menu load game untuk melanjutkan permainan, dan menu exit untuk keluar dari game, menu about untuk melihat deskripsi singkat tentang game ini, menu score untuk melihat score tertinggi yang telah didapatkan, menu store untuk upgrade level senjata.



Gambar 10  
Tampilan Menu Utama

2. Tampilan *Game Play*

Gambar 11 merupakan tampilan *game play* dimana pemain dapat memulai memainkan permainan dengan mengarahkan kamera terhadap objek-objek yang terlihat, lalu menekan button fire untuk menembak. Setiap objek virus yang tertembak akan menambah jumlah score sebesar 10 point, dan juga terdapat jumlah amunisi yang akan berkurang setiap kali pemain melakukan tembakan, setiap score yang didapat akan di akumulasi dan akumulasi score tersebut akan menjadi point yang berfungsi untuk melakukan upgrade senjata.



Gambar 11  
Tampilan *Game Play*

3. Tampilan *Pause*

Gambar 12 merupakan tampilan apabila kita menekan tombol *pause* saat *game play*. Maka akan muncul tiga pilihan yaitu *resume* untuk melanjutkan *game*, *save* untuk menyimpan *game* untuk dilanjutkan kembali nantinya, dan terakhir *back to menu* untuk keluar dari permainan tanpa di *save*.



Gambar 12  
 Tampilan Menu *Pause*

4. Tampilan *Game Over*

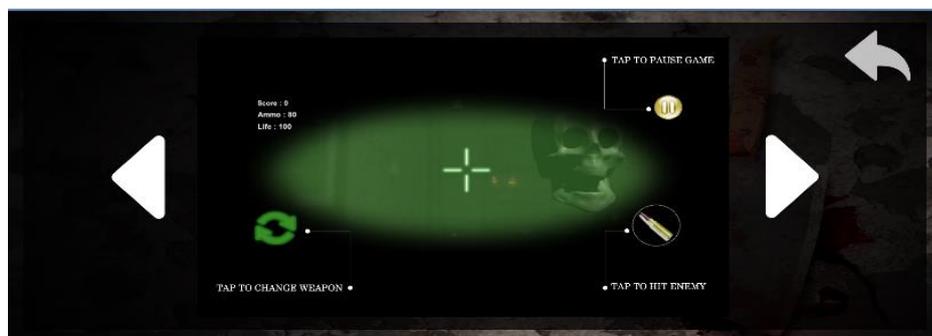
Gambar 13 merupakan tampilan game over yang muncul apabila nyawa pemain habis atau menunjukkan angka nol, dan pemain pun tidak dapat melanjutkan kembali permainan.



Gambar 13  
 Tampilan *Game Over*

5. Tampilan Menu *About*

Gambar 14 merupakan tampilan untuk menu about. Dalam menu ini terdapat penjelasan di setiap halamannya dimana pada halaman pertama akan menampilkan deskripsi singkat tentang cara bermain permainan ini, lalu pada halaman berikutnya terdapat deskripsi singkat tentang musuh yang harus kita hancurkan.



Gambar 14  
 Tampilan Menu *About*

6. Tampilan Menu *Highscore*

Gambar 15 merupakan tampilan pada menu *Highscore*, dimana score dari setiap permainan dengan nilai tertinggi akan disimpan dan akan ditampilkan pada menu *Highscore*.



Gambar 15  
Tampilan Menu *Highscore*

7. Tampilan Menu *Store*

Gambar 16 adalah tampilan menu store yang menampilkan beberapa senjata yang dapat di upgrade ke level yang lebih tinggi. Terdapat point yang dapat digunakan untuk upgrade senjata. Point diperoleh dari setiap score permainan yang dimainkan.



Gambar 16  
Tampilan Menu *Store*

#### 4.3 Pengujian Fungsi Perangkat Lunak

Pengujian perangkat lunak dilakukan dengan menggunakan metode *black box*, sehingga pengujian fungsi-fungsi perangkat lunak menjadi fokus utama. Pengujian dilakukan dengan maksud untuk melihat apakah fungsi-fungsi perangkat lunak sudah berjalan dengan yang diharapkan atau belum.

Tabel 2  
Pengujian Tombol-tombol Pada Menu Utama

No	Aksi	Hasil yang diharapkan	Output	Keterangan
1	KlikButton <i>Start Game</i>	Menampilkan <i>Scene Game Play</i>	Menampilkan <i>Scene Game Play</i>	Berhasil

No	Aksi	Hasil yang diharapkan	Output	Keterangan
2	Klik <i>Button Load Game</i>	Menampilkan <i>Scene Game Play</i> yang sudah di <i>save</i>	Menampilkan <i>Scene Game Play</i> yang sudah di <i>save</i>	Berhasil
3	Klik <i>Button About</i>	Menampilkan <i>Scene About</i>	Menampilkan <i>Scene About</i>	Berhasil
4	Klik <i>Button High score</i>	Menampilkan <i>Scene High score</i>	Menampilkan <i>Scene Highscore</i>	Berhasil
5	Klik <i>Button Store</i>	Menampilkan <i>Scene Store</i>	Menampilkan <i>Scene Store</i>	Berhasil
6	Klik <i>Button Exit</i>	Keluar dari aplikasi <i>Game</i>	Keluar dari aplikasi <i>Game</i>	Berhasil

Tabel 3  
 Pengujian Tombol-tombol Pada *Scene Game Play*

No	Aksi	Hasil yang diharapkan	Output	Keterangan
1	Klik <i>Button Shoot</i>	Menembakan peluru	Menembakan peluru	Berhasil
2	Klik <i>Button Change Weapon</i>	Mengganti senjata	Mengganti senjata	Berhasil
3	Klik <i>Button Pause</i>	Menghentikan <i>game</i> lalu memunculkan beberapa <i>button</i>	Menghentikan <i>game</i> lalu memunculkan beberapa <i>button</i>	Berhasil
4	Klik <i>Button Resume</i>	Melanjutkan <i>game</i> yang sedang di <i>pause</i>	Melanjutkan <i>game</i> yang sedang di <i>pause</i>	Berhasil
5	Klik <i>Button Save</i>	Menyimpan <i>game</i> yang sedang berjalan untuk dilanjutkan kembali nantinya	Menyimpan <i>game</i> yang sedang berjalan untuk dilanjutkan kembali nantinya	Berhasil
6	Klik <i>Button Back To Menu</i>	Kembali ke <i>Scene Menu Utama</i>	Kembali ke <i>Scene Menu Utama</i>	Berhasil

Tabel 4  
 Pengujian Aksi Pada *Game Play*

No	Aksi	Hasil yang diharapkan	Output	Keterangan
1	Peluru pada <i>Weapon 1</i> mengenai <i>object enemy</i>	<i>Enemy</i> hilang dan <i>score</i> bertambah	<i>Enemy</i> hilang dan <i>score</i> bertambah	Berhasil
2	Peluru pada <i>Weapon 2</i> mengenai	<i>Enemy</i> hilang dan <i>score</i> bertambah	<i>Enemy</i> hilang dan <i>score</i> bertambah	Berhasil

No	Aksi	Hasil yang diharapkan	Output	Keterangan
	<i>object enemy</i>			
3	Peluru pada <i>Weapon 1</i> mengenai <i>object cube</i>	<i>Object cube</i> hilang dan peluru <i>weapon 1</i> bertambah	<i>Object cube</i> hilang dan peluru <i>weapon 1</i> bertambah	Berhasil
4	Peluru pada <i>Weapon 2</i> mengenai <i>object Cube</i>	<i>Object cube</i> hilang dan peluru <i>weapon 2</i> bertambah	<i>Object cube</i> hilang dan peluru <i>weapon 1</i> bertambah	Berhasil
5	<i>Enemy</i> mendekati layar	<i>Enemy</i> hilang dan <i>life</i> berkurang	<i>Enemy</i> hilang dan <i>life</i> berkurang	Berhasil

Tabel 5  
Pengujian Tombol-tombol Pada Menu *About*

No	Aksi	Hasil yang diharapkan	Output	Keterangan
1	Klik <i>Button Next</i>	Berpindah ke halaman selanjutnya	Berpindah ke halaman selanjutnya	Berhasil
2	Klik <i>Button Previous</i>	Berpindah ke halaman sebelumnya	Berpindah ke halaman sebelumnya	Berhasil
3	Klik <i>Button Back</i>	Kembali ke tampilan menu utama	Kembali ke tampilan menu utama	Berhasil

Tabel 6  
Pengujian Tombol-tombol Pada Menu *Highscore*

No	Aksi	Hasil yang diharapkan	Output	Keterangan
1	Klik <i>Button Back</i>	Kembali ke tampilan menu utama	Kembali ke tampilan menu utama	Berhasil

Tabel 7  
Pengujian Tombol-tombol Pada Menu *Store*

No	Aksi	Hasil yang diharapkan	Output	Keterangan
1	Klik <i>Button Next</i>	Berpindah ke halaman senjata selanjutnya	Berpindah ke halaman senjata selanjutnya	Berhasil
2	Klik <i>Button Previous</i>	Berpindah ke halaman senjata sebelumnya	Berpindah ke halaman senjata sebelumnya	Berhasil
3	Klik <i>Button Upgrade</i>	Level senjata bertambah dan poin berkurang	Level senjata bertambah dan poin berkurang	Berhasil
4	Klik <i>Button Back</i>	Kembali ke tampilan menu utama	Kembali ke tampilan menu utama	Berhasil

Tabel 8  
Hasil Pengujian Terhadap Lingkungan

No	Lingkungan	Hasil
1	Lingkungan dengan pencahayaan terang	Semua animasi / objek muncul
2	Lingkungan dengan pencahayaan redup	Semua animasi / objek muncul
3	Lingkungan dengan pencahayaan gelap	Semua animasi / objek muncul

Tabel 9  
Hasil pengujian pada Smartphone

No	Tipe Smartphone	Hasil
1	Xiaomi Redmi 5 Plus	Semua animasi / objek muncul, dan <i>game</i> bisa dimainkan dengan baik.
2	Asus Zenfon Max Pro M1	Semua animasi / objek muncul, dan <i>game</i> bisa dimainkan dengan baik.
3	Iphone 8+	<i>File game</i> tidak terbaca.

Hasil dari pengujian fungsi dengan menggunakan metode *black box* telah dilakukan dan dapat dilihat pada tabel-tabel diatas, dimana program yang dibuat sudah berjalan dengan sebagaimana mestinya. Pembuatan *shooting game* dengan menggunakan *augmented reality* dengan sistem operasi android dapat dilakukan dan berjalan dengan baik menggunakan *game engine* Unity.

Lingkungan pencahayaan pun tidak menjadi masalah dikarenakan aplikasi “*Ghost Hunter*” tidak membutuhkan *marker* untuk dilacak oleh kamera. Metode yang digunakan adalah *device tracking* yang merupakan salah satu dari berbagai jenis *markerless*, sehingga animasi / objek akan tetap muncul dengan kondisi pencahayaan yang tidak menentu

## 5 KESIMPULAN

Kesimpulan dari rekayasa perangkat lunak perancangan dan pembuatan *augmented reality shooting game* yaitu Pembuatan *game* dengan tipe *shooting game* dipadukan dengan *Augmented Reality* yang diaplikasikan pada *platform android*, dapat dibuat menggunakan *game engine* Unity3D dengan bantuan dari Software Development Kit Vuforia.

## 6 DAFTAR PUSTAKA

- Clark, Dan, 2013, *Begining C# Object-Oriented Programming*, New York : Apress.
- Craig, B, A., 2013, *Understanding Augmented Reality : Concept and Application*, United State : Morgan Kaupmann.
- Creighton, Henson Ryan., 2011, *United 3D Game Development by Example Beginner Guide : Lite*, Birmingham : PACKT.
- Dregvite, Giedre., Robertas, Damasevicius, 2016, *Information and Software Technology*, Switzerland : Springer International.
- Enterprise, Jubilee, 2010, *Trik Bermain Game Android di PC*, Elex Media Komputindo.
- Fernando, Mario, 2013, *Membuat Aplikasi Android Augmented Reality Menggunakan Vuforia SDK dan Unity*, Solo : Buku AR Online.
- Fishwick. A. P, 2008, *Aesthetic Computing*, London : The MIT Press.
- Flavell, Lance, 2017, *Beginning Blender*, Apress.
- Furht, Borko, 2011, *Handbook of Augmented Reality*, Florida : Spinger.

- Harwani, B, M., 2012, *Android Programming Unleashed*, United State : Pearson Education, Inc.
- Kipper, Gred., Rampolla, Joseph, 2012, *Augmented Reality : An Emerging Technologies Guide to AR*, USA : Elsevier, Inc.
- Lethbrige, C, Timothy. Laganier, Robert , 2011, *Object-Oriented Software Engineering*, McGrawHill.
- Nugroho, Adi, 2009, *Rekayasa Perangkat Lunak menggunakan UML dan JAVA*, C.V ANDI OFFSET.
- Nugroho, Adi, 2010, *Rekayasa Perangkat Lunak Berorientasi Objek dengan Metode USDP*, C.V ANDI OFFSET.
- Pramudya, Puja, 2011, *Membuat Aplikasi Mobile dengan Qt SDK*, Yogyakarta : Andi.
- Pressman, Roger S, 2010, *Software Engineering : A Practitioner's Approach Seventh Edition*, New York : McGraw-Hill Companies, Inc.
- Rafrastara, A. F, dkk, 2009, *Membuat Game Fighting dengan Flash*, Jakarta : PT Elex Media Komputindo.
- Roedavan, Rickman, 2014, *Unity Tutorial Game Engine*, INFORMATIKA.
- Siberto, Ivan C, 2009, *Langkah Membuat Game 3D*, Yogyakarta : MediaKom.
- Simarmata, Janner, 2010, *Rekayasa Perangkat Lunak*, C.V. ANDI OFFSET.
- Sommerville, Ian, 2011, *Software Engineering 9th Edition*, Cloth.
- Thron, Alan, 2013, *Unity 4 Fundamentals : Get Started at Making Games with Unity*, United Kingdom : Focal Press.
- Vandome, Nick, 2014, *Android Tablet in Easy Steps : Fully Illustrated Using Google Nexus*, United Kingdom : In Easy Steps Limited.
- Vento, Mario, 2009, *Image Analysis and Processing – ICIAP 2009*, Vietri sul Mare : Springer.