

# Perancangan *Game* Berjenis *Third Person* Dengan Tema Pemadam Kebakaran *Platform* *Android*

Oleh:

**Joshua Roniek<sup>1</sup>**

*Sekolah Tinggi Manajemen Informatika Dan Komputer  
Masa Depan Tangerang*

**Prayogo<sup>2</sup>**

*Sekolah Tinggi Manajemen Informatika Dan Komputer  
Masa Depan Tangerang*

[Joshualast18@gmail.com<sup>1</sup>](mailto:Joshualast18@gmail.com)

## ABSTRAK

Game adalah salah satu hiburan karena dapat mengurangi tingkat kepenatan dari rutinitas pekerjaan sehari-hari. Game ini juga mampu meningkatkan kecerdasan seseorang saat bermain. Tujuan dari penelitian ini adalah merancang aplikasi Game yang digunakan sebagai media hiburan dan juga simulasi pemadam kebakaran bagi masyarakat yang membutuhkan.

Dalam penelitian ini menggunakan metode wawancara dan melihat simulasi yang ada dilakukan pada saat training pemadam kebakaran yang diselenggarakan PT Gajah Tunggal TBK. Pembuatan aplikasi game ini menggunakan game engine Unity seri 4 dengan didukung Asset store unity package yang memudahkan dalam pembuatan game ini metode blackbox dan whitebox dalam pengujiannya agar semua skrip dapat berjalan sesuai yang seharusnya agar tidak terjadi error saat aplikasi dijalankan.

Dari penelitian ini dihasilkan aplikasi game Berjenis *Third Person* Dengan Tema Pemadam Kebakaran, yang memiliki kesulitan di setiap levelnya yang sudah setting sesuai dengan keadaan yang hampir mirip dengan kejadian sebenarnya, agar user menjadi tertantang dalam memainkan game third person bertema pemadam.

**Kata kunci:** *Permainan, Sudut Pandang, Orang Ketiga.*

## ABSTRACT

*Game is one of the entertainment because it can reduce the level of fatigue of daily work routines. This game is also able to increase one's intelligence when playing. The purpose of this research is to design a game application that is used as a medium of entertainment and also a fire extinguisher simulation for people in need.*

*In this study using the interview method and see the simulation that was carried out during fire fighting training organized by PT Gajah Tunggal TBK. The making of this game application uses the Unity 4 series game engine supported by the Asset store unity package which makes it easy to make this game a Blackbox and Whitebox method in testing so that all scripts can run as they should so that no errors occur when the application is run.*

*This research resulted in the application of the Third Person Type Game with a Fire Extinguisher Theme, which has difficulties at each level that have been set according to circumstances that are almost similar to the actual events so that users become challenged in playing the third-person game-themed extinguisher.*

**Keyword:** *Game, Perspective, Third Person*

---

## A. PENDAHULUAN

*Game* merupakan salah satu hiburan karena mampu mengurangi tingkat kepenatan seseorang dari rutinitas pekerjaan setiap hari. *Game* juga mampu meningkatkan kecerdasan seseorang ketika *Game* tersebut memerlukan tingkat ketangkasan dari seorang pemain. Beberapa *game* yang beredar saat ini terdapat unsur mendidik, ketangkasan dan ada pula unsur kekerasan, maka ketika *game* itu diperjual belikan terdapat batas umur pemakainya.

Kemampuan sumber daya manusia ikut meningkat seiring perkembangan teknologi informasi. Dengan mengikuti perkembangan teknologi informasi, konsol *game* pun ikut berkembang seperti perangkat konsol *game* *Play Station 4* (rilis 2012), *X-BOX 360 Slim*, *Nintendo Wii* dan *personal computer* atau yang biasa disebut dengan *PC Games*. Selain dari konsol, *game* juga bias dijalankan dari *Handphone Android* yang memiliki grafis tidak kalah dari konsol diatas.

Dahulu untuk membuat *game* membutuhkan kemampuan lebih dalam pemograman. Pada saat ini para *developer* dapat membangun *game* lebih mudah dengan menggunakan *game engine*. *Unity 4* merupakan salah satu *game engine multi-platform* dan berbagai jenis *genre game* dapat dibuat dengan *Unity 4* salah satu *genre* yang dipakai dalam perancangan ini adalah *game Third Person* yang mengambil sudut pandang orang ketiga.

Perkembangan *game Third Person* dimulai dengan munculnya *game Wolfenstein 3D* yang merupakan nenek moyangnya *game* bergenre *Third Person*. Hadirnya *game Warcraft I* mulai mengawali penggunaan teknologi grafis *3D* dalam *genre Third Person* serta merupakan *game* yang bias dimainkan secara online. *Game Third Person* yang paling dikenal di Indonesia adalah *Ragnarok online* dimana pada saat itu merupakan *game* favorit para *gamers*. *Resident Evil* dan *Gear of Wars* yang saat ini hadir dengan teknologi grafis terbaru tingkat tinggi merupakan pionir *game Third Person 3D* selanjutnya. Perkembangan *game Third Person* saat ini adalah *Assasin Creed Unity* yang di sajikan semakin halus tingkat teknologi grafis dan narasi.

Berawal dan terinspirasi dari *game Third Person*, terdapat peluang mengenai membuat *game* dengan *genre Third Person* bertemakan pemadam kebakaran menggunakan software *Unity 4* dan beberapa *software* pendukung lain. *Game* ini bertujuan menjadi salah satu hiburan karena mampu mengurangi tingkat kepenatan seseorang dari rutinitas pekerjaan setiap hari dan mengasah kemampuan ketangkasan dan kecepatan dalam mematikan api di *game* tersebut. Karena sejalan dari tujuan dari *game* ini sebagai aplikasi *game Third Person* sebagai media hiburan dan simulasi kebakaran dalam bentuk permainan dengan keadaan yang hampir menyerupai kejadian sebenarnya.

## Rumusan Masalah

Bagaimana Desain *Game* Berjenis *Third Person* Dengan Tema Pemadam Kebakaran Platform *Android*

## **B. TINJAUAN PUSTAKA**

### Definisi Perancangan

Perancangan adalah suatu proses yang bertujuan untuk menganalisis, menilai memperbaiki dan menyusun suatu sistem, baik sistem fisik maupun non fisik yang optimum untuk waktu yang akan datang dengan memanfaatkan informasi yang ada (Santoso & Purnama, 2011).

### Definisi Third Person

*Third Person Shooter* (TPS) yaitu jenis game yang mengutamakan kecepatan gerakan dalam permainan. Dalam jenis ini akan lebih banyak baku tembak serta mengharuskan untuk bertahan hidup selama mungkin hingga permainan berakhir. dan jenis permainan dengan karakter pemain yang menggunakan sudut pandang orang ketiga yang dapat melihat keseluruhan karakter pemain dari belakang (punggung) (Gunawan et al, 2017).

### Definisi Game

*Game* atau permainan merupakan sesuatu yang dapat dimainkan dengan aturan tertentu sehingga ada yang menang dan ada yang kalah biasanya tidak dalam konteks serius atau dengan tujuan untuk *refreshing* (Gunadi & Al Fatta, 2012). permainan adalah jenis

kegiatan/aktivitas bermain yang dilakukan dalam konteks kenyataan berpura-pura (*pretended reality*) dimana pemain mencoba untuk mencapai sesuatu sesuai maunya mencapai tujuan yang dianggap berharga dengan bertindak sesuai aturan. Sedangkan, game Design adalah rancangan permainan adalah sebuah proses untuk menciptakan konten dan aturan permainan (Rouse, 2005). Rancangan permainan yang baik adalah proses menciptakan tujuan yang mana seorang pemain merasa dimotivasi untuk mencapai dan mengikuti aturan disaat ia mengambil keputusan tersebut (Brathwaite & Schreiber, 2009).

## **C. METODE**

### Metode Perancangan

Dalam perancangan membutuhkan metode sebagai acuan dalam pembuatan (proses desain). Metode perancangan dikenal sebagai suatu cara atau tahapan yang dilakukan dalam sebuah proses perancangan, metode ini dibutuhkan untuk memudahkan perancang dalam mengembangkan ide rancangan (Trisna, 2016). Metode yang dilakukan oleh seseorang berbeda-beda berdasarkan kebutuhannya. Dalam penerapannya, metode perancangan yang dilakukan memiliki beberapa tahap yaitu:

- 1) Riset dan Penyusunan Konsep Dasar Pada tahap ini ide dasar, objektif, tema, target audience, teknologi, media (platform), serta berbagai batasan lain dirumuskan. Tahapan

riset ini menjadi sebuah tahapan krusial, berbagai elemen dasar dari sebuah game disusun di sini.

2) Perumusan Gameplay Pada tahap ini para game designer merumuskan gameplay/game mechanic yang akan digunakan dalam sebuah game. Gameplay adalah pola, aturan, atau mekanisme yang mengatur bagaimana proses interaksi pemain dengan game yang diciptakan. Gameplay ini juga mengatur bagaimana seorang pemain bisa memenuhi objektif dari game dan mendapatkan pengalaman bermain yang menyenangkan.

3) Penyusunan Asset dan Level Desain Tahapan ini fokus pada penyusunan konsep dari semua karakter serta asset (termasuk suara/musik) yang diperlukan. Pada saat yang sama tim juga mulai melakukan Level Design atau pengelompokan tingkat kesulitan serta berbagai asset yang tepat pada tiap level (jika ada lebih dari 1 level) agar game tersebut bisa menghadirkan pengalaman bermain yang optimal.

4) Test Play Pada tahapan ini sebuah prototype/dummy dihadirkan untuk menguji gameplay serta berbagai konsep yang telah tersusun, baik dalam tiap level maupun secara keseluruhan, serta melakukan berbagai perbaikan yang diperlukan.

5) Development Pada tahap ini seluruh konsep (karakter dan asset) yang sebelumnya

telah tersusun mulai dikembangkan secara penuh, game engine mulai dikembangkan, dan semua elemen mulai dipadukan.

6) Alpha/close beta test Fokus utama pada tahap ini adalah untuk mengetahui apakah semua komponen utama dari game telah mampu memberikan user experience seperti yang diharapkan sekaligus juga untuk mendeteksi adanya masalah teknis yang belum terdeteksi pada tahapan sebelumnya (Blackman, 2011).

#### Metode Pengumpulan Data

Dalam pembuatan desain game perlu adanya data yang memiliki fungsi untuk penunjang perancangan. Data yang didapat, dikumpulkan melalui beberapa metode yang dilakukan. Metode pengumpulan data yang dilakukan dalam perancangan (desain) *game* ini terdiri dari:

1. Studi Pustaka yaitu Metode pengumpulan data dengan cara mempelajari buku-buku ilmu pengetahuan dan makalah ilmiah yang berhubungan dengan masalah yang dibahas pada desain game.
2. Wawancara yaitu Salah satu metode pengumpulan data yang di dapat melalui bertanya langsung kepada narasumber untuk mendapatkan informasi yang di butuhkan, pada perancangan *game third person* untuk

mengetahui gambaran simulasi kebakaran.

3. Studi Internet dalam Studi ini penulis ingin melihat website yang membahas tentang game *third Person* untuk mencari informasi yang berhubungan dengan masalah yang dibahas pada skripsi ini.
4. Observasi yaitu Suatu cara pengumpulan data dengan mengadakan pengamatan langsung terhadap suatu obyek dalam suatu periode tertentu dan mengadakan pencatatan secara sistematis tentang

hal-hal tertentu yang diamati, untuk mengetahui apa saja yang dibutuhkan dalam perancangan *game third person* ini (Jogiyanto. 1999).

#### Jadwal Implementasi

Dalam proses desain yang dilakukan, perlu diadakannya, jadwal sebagai acuan dalam membuat rancangan. Dikarenakan, dalam desain perlu adanya penyesuaian waktu pembuatan agar tidak memakan waktu yang lama dan sesuai dengan target yang ditentukan

Tabel 1 *Timeline* Perancangan

<i>Timeline</i> Desain									
No	Kegiatan	Bulan I	Bulan II	Bulan III	Bulan IV	Bulan V	Bulan VI	Bulan VII	Bulan VIII
1	Analisa aplikasi								
2	Penyiapan data								
3	Pembuatan Program								
4	Testing Aplikasi								
5	Implementasi								
6	Dokumentasi								

*Timeline* perancangan pertama di lakukan analisa sistem aplikasi sebagai rancangan dasar sebelum memulai perancangan, penyiapan data untuk proses pembuatan program setelah seluruh analisa perancangan selesai di buat, setelah proses analisa dan persiapan data di lakukan tahap pembuatan program, testing program di lakukan sejalan dengan tahap pembuatan program, implementasi di lakukan apabila

tahap testing sudah selesai, dokumentasi di lakukan sebagai pencatatan setiap tahap yang telah di lakukan.

#### Alat Bantu Perancangan

Perangkat lunak untuk multimedia sangat membantu dan mempermudah para penggunanya. Perangkat lunak seperti grafis, animasi, dan suara sangat berperan penting dalam pembuatan Game. Perangkat lunak

yang digunakan penulis sebagai alat bantu dalam perancangan game third person ini meliputi Adobe Photoshop sebagai pengolahan grafis dan Adobe flash sebagai pengolahan animasi. Unity 4.0 sebagai game engine dan software blender untuk permodelan 3 Dimensi.

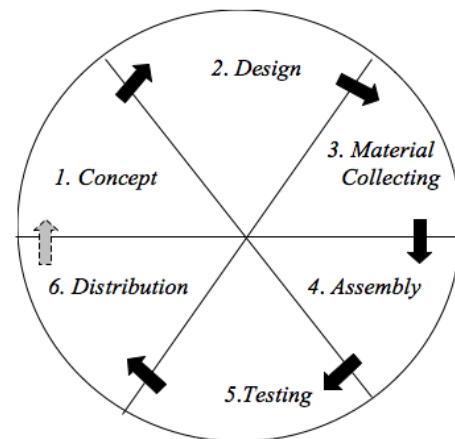
### Metode Pengembangan

Menurut (Luther, 1994), metodologi pengembangan terdiri dari enam tahap, yaitu:

1. Concept (pengonsepan)
2. Design (pendesainan)
3. Material collecting (pengumpulan materi)
4. Assembly (pembuatan)
5. Testing(penguujian)
6. Distribution(pendistribusian).

Keenam tahap ini tidak harus berurutan dalam praktiknya, tahap-tahap tersebut dapat saling bertukar posisi. Meskipun begitu, tahap concept memang harus menjadi hal yang pertama kali dikerjakan. (Sutopo, 2003) mengadopsi metodologi Luther dengan modifikasi, seperti yang terlihat pada gambar bagan dibawah ini.

1. Design (pendesainan)



Gambar 1. Siklus Pengembangan Aplikasi *Multimedia* (Sumber: Sutopo, 2003:32)

#### 1. *Concept*

Tahap *concept* (pengonsepan) adalah tahap untuk menentukan tujuan dan siapa pengguna program (identifikasi audiens). Tujuan dan pengguna akhir program berpengaruh pada nuansa *multimedia* sebagai pencerminan dari identitas organisasi yang menginginkan informasi sampa pada pengguna akhir. Karakteristik pengguna termasuk kemampuan pengguna juga perlu dipertimbangkan karena dapat memengaruhi pembuatan desain.

Selain itu, tahap ini juga akan menentukan jenis aplikasi (presentasi, interaktif dan lain-lain). Dasar aturan untuk perancangan juga ditentukan pada tahap ini, misalnya ukuran aplikasi, target, dan lain-lain. Output dari tahap ini biasanya berupa dokumen yang bersifat naratif untuk mengungkapkan tujuan proyek yang ingin dicapai.

#### 2. *Design*

*Design* (perancangan) adalah tahap pembuatan spesifikasi mengenai arsitektur program, gaya, tampilan, dan kebutuhan

material/bahan untuk program. Spesifikasi dibuat serinci mungkin sehingga pada tahap berikutnya, yaitu *material collecting* dan *assembly*, pengambilan keputusan baru tidak diperlukan lagi, cukup menggunakan keputusan yang sudah ditentukan pada tahap ini. Meskipun demikian, pada praktiknya, pengerjaan proyek pada tahap awal masih akan sering mengalami penambahan bahan atau pengurangan bagian aplikasi, atau perubahan-perubahan lain.

Tahap ini biasanya menggunakan *storyboard* untuk menggambarkan deskripsi tiap *scene*, dengan mencantumkan semua obyek *multimedia* dan tautan ke *scene* lain dan bagan alir (*flowchart*) untuk menggambarkan aliran dari satu *scene* ke *scene* lain.

### 3. *Material Collecting*

*Material Collecting* adalah tahap pengumpulan bahan yang sesuai dengan kebutuhan yang dikerjakan. Bahan-bahan tersebut, antara lain gambar *clip art*, foto, animasi, *video*, *audio*, dan lain-lain yang dapat diperoleh secara gratis atau dengan pemesanan kepada pihak lain sesuai dengan rancangannya. Tahap ini dapat dikerjakan secara paralel dengan tahap *assembly*. Namun, pada beberapa kasus, tahap *material collecting* dan tahap *assembly* akan dikerjakan secara *linear* dan tidak paralel.

## D. HASIL DAN PEMBAHASAN Perancangan Sistem

### Use Case Diagram

### 4. *Assembly*

Tahap *assembly* adalah tahap pembuatan semua obyek atau bahan *multimedia*. Pembuatan aplikasi didasarkan pada tahap *design*, seperti *storyboard*, bagan alir, dan/atau struktur navigasi.

Tahap ini biasanya menggunakan perangkat lunak *authoring*, seperti Macromedia Director. Selain itu, Macromedia Flash atau produk *open source* yang gratis, yaitu Sophie yang dapat berjalan di Linux maupun di Mac OS X juga dapat digunakan.

### 5. *Testing*

Tahap *Testing* (pengujian) dilakukan setelah menyelesaikan tahap pembuatan (*assembly*) dengan menjalankan aplikasi/program dan melihatnya apakah ada kesalahan atau tidak. Tahap pertama pada tahap ini disebut tahap pengujian alpha (*alpha test*) yang pengujiannya dilakukan oleh pembuat atau lingkungan pembuatnya sendiri. Setelah lolos dari pengujian *alpha*, pengujian *beta* yang melibatkan pengguna akhir akan dilakukan.

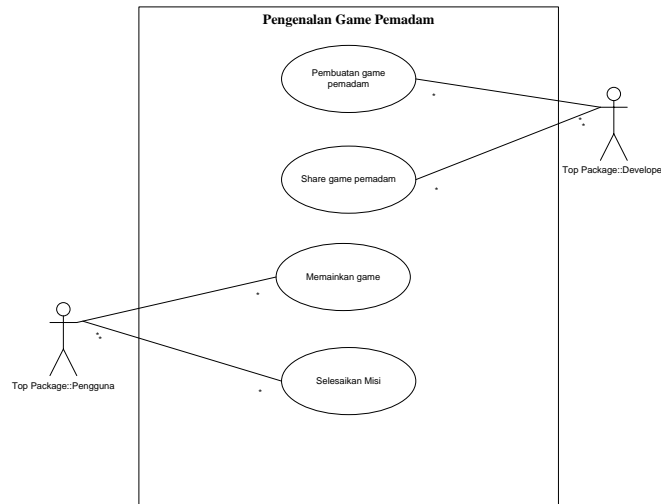
### 6. *Distribution*

Pada tahap ini, aplikasi akan disimpan dalam suatu media penyimpanan. Jika media penyimpanan tidak cukup untuk menampung aplikasinya, kompresi terhadap aplikasi tersebut akan dilakukan (Gunawan, 2017).

Diagram *use case* digunakan untuk menjelaskan dan menggambarkan sistem dan perilaku pengguna terhadap sistem. Use case diagram menekankan tentang apa yang akan

diperbuat oleh sistem dan bukan aktor, sedangkan perilakunya diwakili oleh *use case*.  
 menekankan bagaimana sistem tersebut bertindak. Pengguna sistem diwakili oleh

Diagram *Use Case* aplikasi *game third person* bertema pemadam

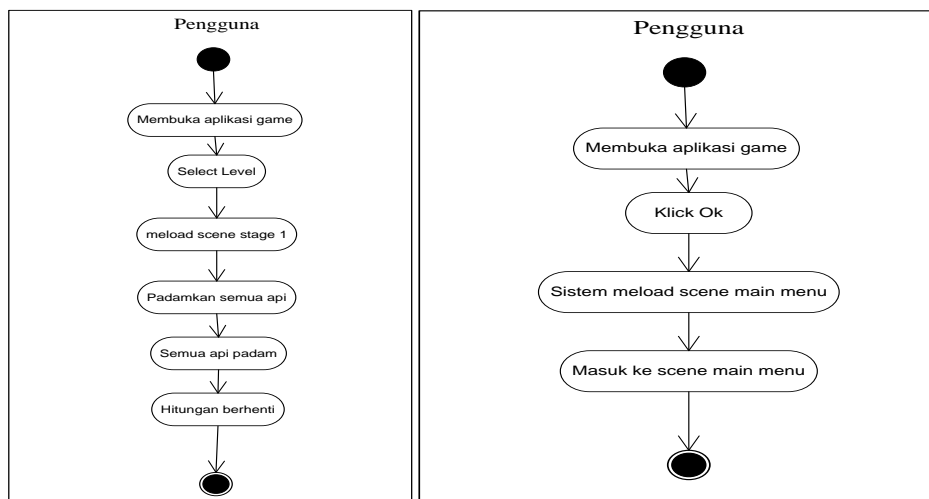


Gambar 2: *Use case* Pengenalan game pemadam

Diagram Activity

*Activity diagram* menjelaskan tentang berbagai aktivitas yang berlangsung selama pengguna

berinteraksi dengan sistem. *Activity diagram* akan membantu menjelaskan diagram *use case* lebih mendalam.



Gambar 3: Diagram *Activity Scene*

Diagram *Activity* menjelaskan interaksi antara pengguna dengan aplikasi game yang berbasis *third person*. Aktivitas game dimulai ketika pengguna memilih menu *Game*. Sistem akan

menampilkan *scene* Main menu. aktivitas game dimulai ketika pengguna membuka aplikasi dan masuk ke halaman menu dan pengguna memilih level yang diinginkan,

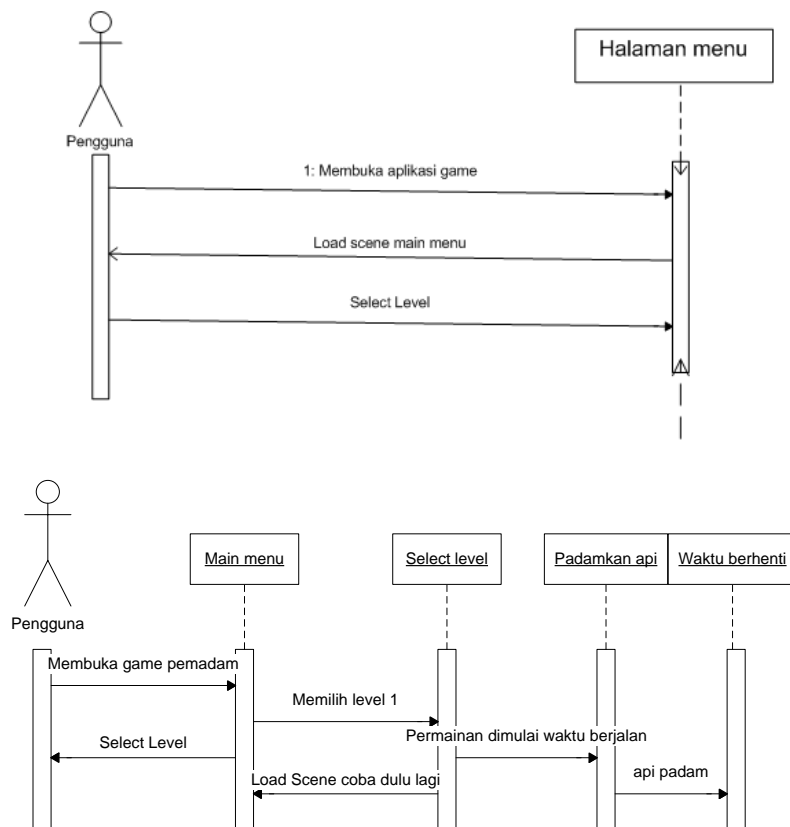


permainan otomatis memulai dengan sendirinya dengan dimulainya hitungan, pengguna akan mencari titik api yang ada dalam arena tersebut, setelah api semua padam maka hitungan akan berhenti dan hitungan itu akan menjadi score pengguna.

### Diagram Sequence

*Sequence diagram* menggambarkan interaksi antar objek di dalam dan di sekitar sistem

(termasuk *user*, *display* dan sebagainya). *Sequence diagram* menjelaskan tentang proses interaksi antar pengguna dengan sistem dalam *scene main menu*. Pengguna membuka aplikasi dan masuk ke *scene main menu* kemudian akan tampil halaman main menu dalam permainan *third person* bertema pemadam.



Gambar 4: Sequence Scene Load level scene main menu

Menjelaskan tentang proses interaksi antar pengguna dengan sistem dalam *scene main menu*. Pengguna membuka aplikasi dan masuk ke *scene main menu* kemudian akan tampil halaman main menu dalam permainan *third person* bertema pemadam.

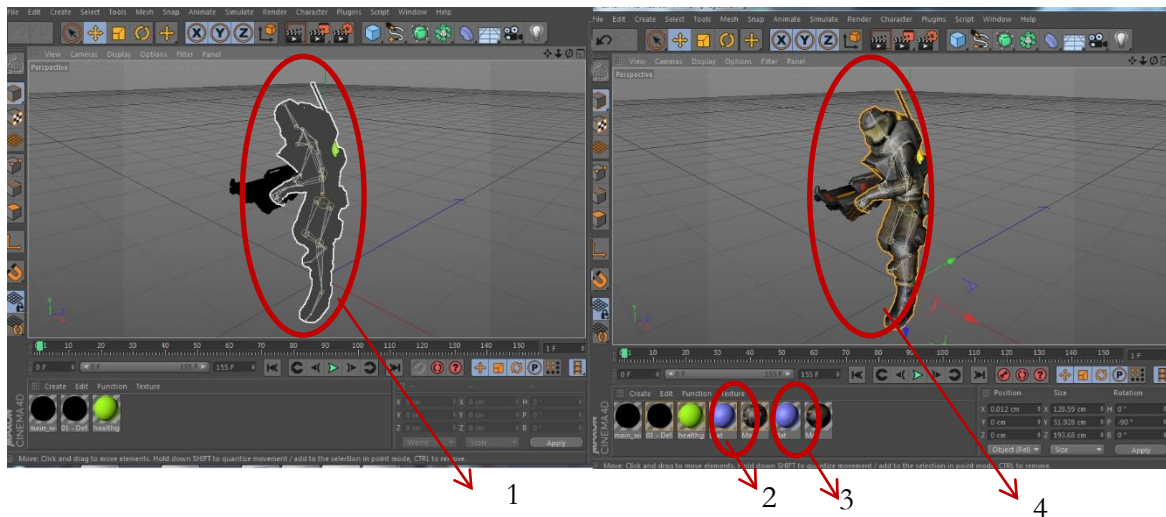
### Pembuatan Karakter (Modelling)

Model 3D mewakili objek 3D menggunakan koleksi poin dalam ruang 3D, dihubungkan dengan berbagai entitas geometris seperti segitiga, garis, permukaan lengkung, dll. Menjadi pengumpulan data (poin dan

informasi lainnya), model 3D dapat dibuat dengan tangan, *algorithmically* (model prosedural), atau scan. Model 3D banyak digunakan di mana saja di grafis 3D. Sebenarnya, mereka menggunakan luas mendahului penggunaan grafis 3D pada

a. Desain karakter

komputer pribadi. Banyak permainan komputer digunakan pra-gambar membuat model 3D seperti sprite sebelum komputer dapat membuat mereka secara *real-time*.



Gambar 5: Pembuatan karakter dalam *game third person*

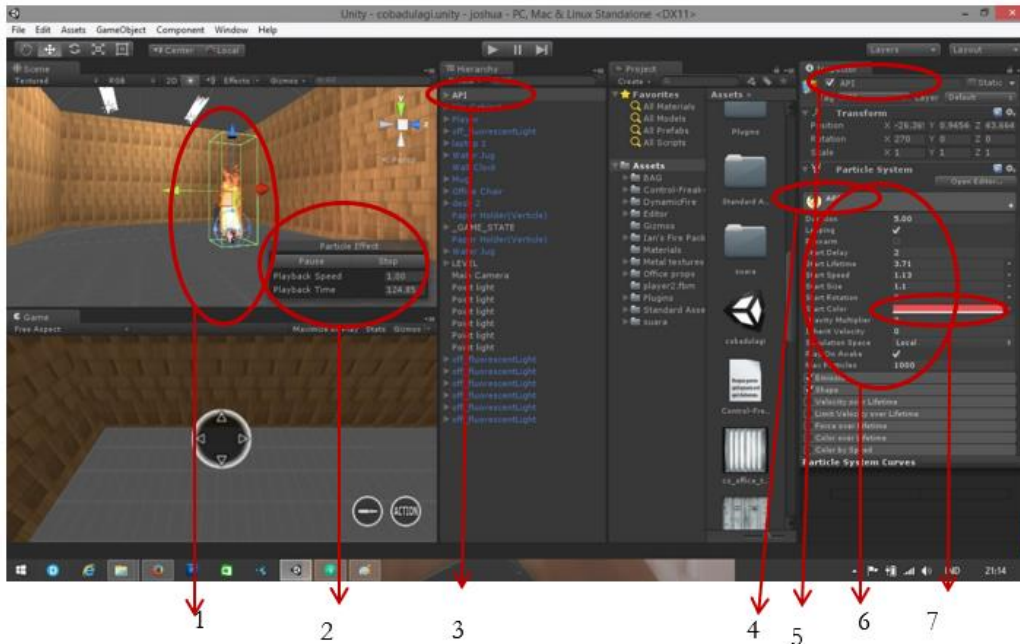
Keterangan:

1. Tahap pemberian kerangka (*Rigid body*)
2. *Material texture* untuk *body* karakter
3. *Material texture* untuk senjata
4. *Finishing texturing* karakter

Tahap pembuatan karakter utama dalam *game third person* ini, dari tahap membuat kerangka (*Rigid body*) dan tahap *texturing*, *rigid body* adalah tahapan pemberian kerangka yang tidak terlihat agar objek

animasi mempunyai sendi-sendi untuk bisa bergerak sesuai yang diinginkan animator. Sedangkan *Texturing* adalah proses pemberian karakteristik permukaan pada objek. Maksud dari karakteristik adalah termasuk pewarnaan, kilauan, dan lainnya. Pada umumnya teksturing adalah pemberian warna pada permukaan objek atau pengecatan, walaupun ada proses yang mengubah geometri objek.

b. Desain komponen (api)

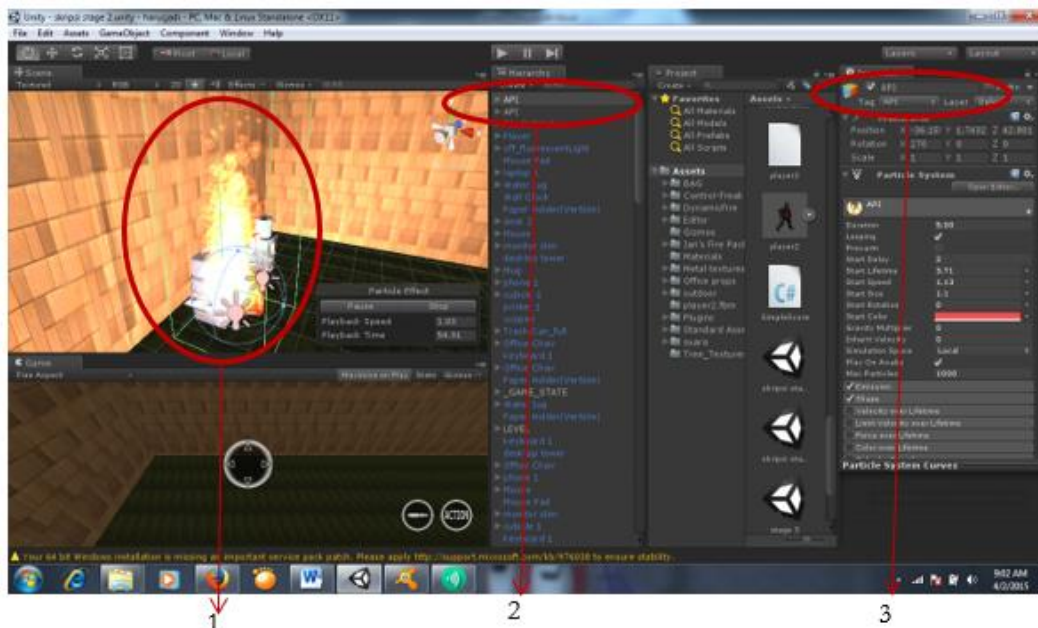


Gambar 6: Perancangan pembuatan api  
Keterangan :

1. Objek api dengan *box collider*
2. *Particle Effect time* untuk waktu penggulungan api
3. Hierarki objek api aktif
4. *Texture* api
5. Objek name api dan memiliki Tag API

6. Settingan Objek api
7. *Starting Colour*

Perancangan api menggunakan partikel system yang terdapat di Unity dan menambah *texture* api, kemudian render ulang agar api tidak berhenti bergerak.

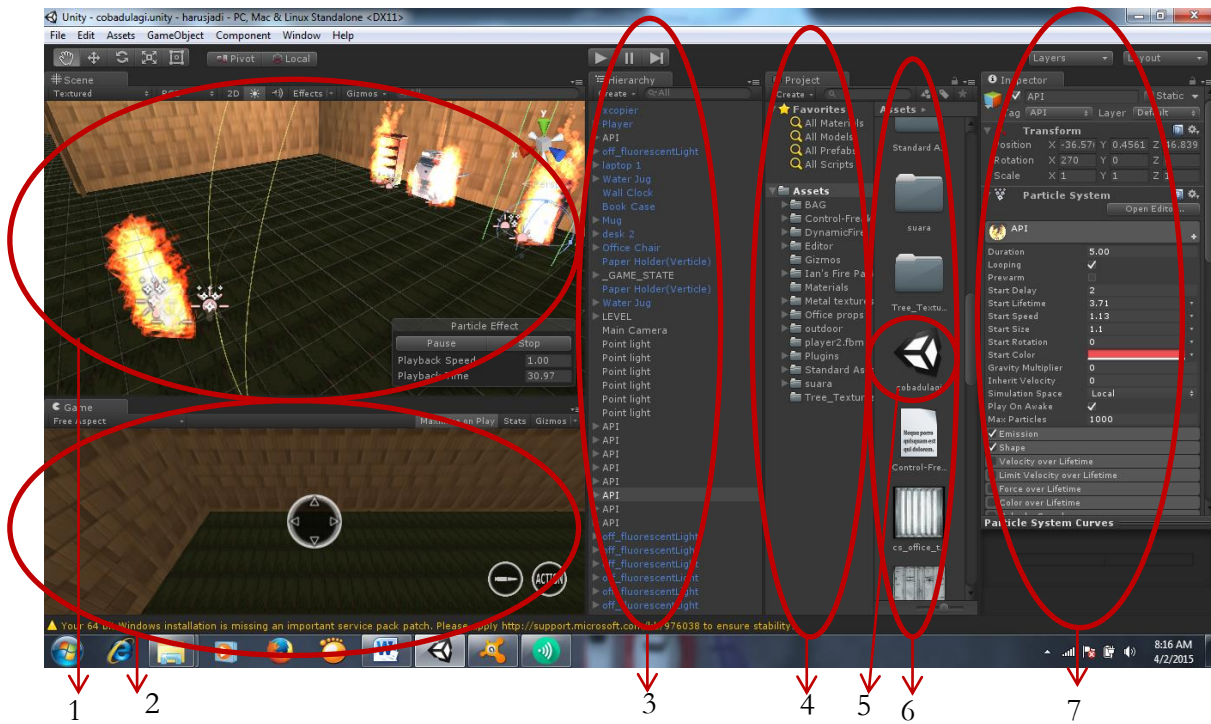


Gambar 7: Rendering Api dengan default CC Partikel  
Keterangan:

1. Partikel api dirender agar seperti aslinya dengan ditambah *box collider* sebagai box yang tidak terlihat untuk tabrakan dengan partikel air supaya api itu ke *destroy*.

2. Objek api dalam keadaan aktif di hierarki objek.
3. Objek api di berikan Tag API supaya *script* mengetahui objek mana yang akan *destroy*.

c. *Desain area (stage) game*



Gambar 8: Gambar *Stage* untuk latihan

Keterangan :

1. Menampilkan *scene* cobadululagi.unity untuk tempat latihan sebelum permainan dimulai.
2. *Tab free aspect* sebagai tempat untuk tes permainan *scene* cobadululagi.unity.
3. *Tab* menampilkan *Hierarki game objek*.

4. *Tab* menampilkan semua *Asset* yang ada dalam *project game*.
5. *Icon* untuk cobadulagi.unity
6. *Tab* untuk melihat secara detail dari isi *folder Asset*.
7. *Tap inspector* untuk setingan lebih rinci pada *game objek*.



Gambar 9: Stage 2 untuk latihan

### Coding

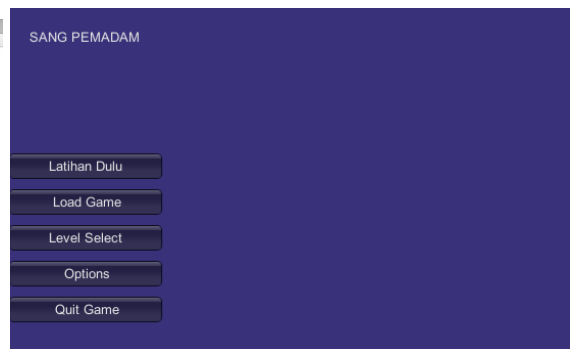
Source Code (atau disebut juga *source*) adalah kumpulan pernyataan atau deklarasi bahasa pemrograman komputer yang ditulis dan dapat di baca manusia. Source code memungkinkan programmer untuk berkomunikasi dengan komputer

menggunakan beberapa perintah yang telah terdefinisi. Disini penulis menggunakan bahasa program C# dalam semua Source code dalam permainan *Third Person Pemadam*, karena lebih mudah untuk pemula dalam pembuatan game.

```

MainMenuScripts
MainMenuScript - No selection
1 private bool _isFireMenu = true;
2 private bool _isLevelSelectMenu = false;
3 private bool _isLoadGameMenu = false;
4 private bool _isOptionsMenu = false;
5
6
7 // Use this for initialization
8 void Start ()
9 {
10 }
11
12 // Update is called once per frame
13 void Update ()
14 {
15 }
16
17 void OnGUI()
18 {
19     GUI.Label(new Rect(50, 75, 300, 25), "SANG PEMADAM");
20     FireMenu();
21     LoadGameMenu();
22     LevelSelectMenu();
23     OptionsMenu();
24 }
25
26 if (_isLevelSelectMenu == true || _isLoadGameMenu == true || _isOptionsMenu == true)
27 {
28     if (GUI.Button(new Rect(15, Screen.height - 35, 150, 25), "Back"))
29     {
30         _isLevelSelectMenu = false;
31         _isLoadGameMenu = false;
32     }
33 }
34

```



Gambar 10: Script dan tampilan main menu.cs

Script mainmenus.cs berfungsi untuk menampilkan menu pembuka, yang berisi menu latihan dan *select level*.

```
MainMenuScript.cs x SimpleScore.cs x BulletCS.cs x
No selection
24 // Destroy this bullet if it didn't hit anything...
25
26 if ((this.lifetime += Time.deltaTime) > this.maxLifetime)
27     Destroy(this.gameObject);
28 }
29
30
31 // -----
32 void OnTriggerEnter(Collider objectHit)
33 {
34     // TODO : explode, inflict damage, etc.
35
36     if (this.gun != null)
37     { Destroy(this.gameObject);
38       // ...
39       if (objectHit.tag == "API"){
40         objectHit.tag = "BUKANAPI";
41         objectHit.particleSystem.enableEmission = false;
42         SimpleScore.StopNilai();
43       }
44     }
45
46
47 #if UNITY_EDITOR
48     Debug.Log("Fr[" + Time.frameCount + "] bullet ["+this.name+"] hit [" + objectHit.name + "]:");
49 #endif
50
51     // Destroy on impact.
52
53     Destroy(this.gameObject);
54 }
55 }
56
```

Gambar 11: *Script* bulletCS.cs

```
MainMenuScript.cs x SimpleScore.cs x
SimpleScore ▶ Update ()
1 using UnityEngine;
2 using System.Collections;
3 public class SimpleScore : MonoBehaviour {
4     static bool Mulai = true;
5     static int Max = 9;
6     static int Current = 0;
7     int Nilai = 0;
8     // Use this for initialization
9     void Start () {
10         Current = 0;
11         InvokeRepeating("HitungScore",1,1);
12     }
13
14     // Update is called once per frame
15     void Update () {
16         if (!Mulai){
17             CancelInvoke("HitungScore");
18         }
19     }
20     public static void StopNilai(){
21         Current++;
22         if (Current == Max){
23             Mulai = false;
24         }
25     }
26     void HitungScore(){
27         Nilai = Nilai + 1;
28     }
29     void OnGUI(){
30         GUI.Label(new Rect(50,50,100,100), Current.ToString() + " "+ Nilai.ToString());
31     }
32 }
33
```

Gambar 12: *Script* Simple Score

Setelah permainan dimulai otomatis hitungan akan berjalan sampai nilai Max dari api yang tentukan telah *didestroy* (hilangkan) oleh partikel air yang ditembakkan, maka hitungan akan berhenti dan akan menjadi *score* dari permainan, semakin cepat mematikan api yang ada, maka *score* akan menjadi lebih baik.

## E. KESIMPULAN

### Kesimpulan

Selama proses implementasi dan pengujian yang dilakukan terhadap aplikasi *game third person* bertema pemadam untuk permainan dan simulasi pemadam kebakaran. Pembuatan *Game* berjenis *Third Person* Bertema Pemadam kebakaran dengan menggunakan *game engine* Unity seri keempat(4) dengan menggunakan *asset unity* yang bernama *control freak* untuk membuat *game* berjenis *shooter* yang memudahkan dalam perancangan *game* bertema pemadam ini, dengan mengimport semua *asset* kedalam *unity* dan merancang *game* tersebut dengan *setting* sudut pandang orang ketiga (*third person*), lalu untuk perancangan api menggunakan *CCpartikel* yang ada di *unity*, diberi efek suara api terbakar diberi *Box Collider* dan diberi tag API, untuk kodingan menggunakan bahasa program *C sharp*(#) yang dibuat di *plugin MonoDevelop* dengan nama *BulletCS* yang memiliki fungsi *mendestroy* api yang sudah di beri *Tag* API, semua *asset* di *compile* dan di *build* untuk device Android.

Semua objek yang ada di *game* ini dibuat dengan program *Cinema4D*, mengambil *asset unity* yang terdapat di *asset store* yang didapat secara gratis dan untuk *asset unity* yang berbayar didapat dari membeli secara langsung saat mengembangkan *game* melalui website *unity asset store*.

Unsur edukasi didapat dari bermain *game* disaat kita dilatih untuk menyelesaikan permainan tersebut yang memiliki tingkatan kesulitan. Semakin tinggi level maka semakin tinggi kesulitannya sehingga pemain belajar dari kesulitan tersebut untuk menyelesaikan *game*. Dalam *game third person bertema pemadam* ini pemain harus memadamkan semua api yang ada untuk menyelesaikan misi yang di berikan dan akan mendapatkan nilai dari hasil memadamkan api tersebut dan setiap level memiliki kesulitan yang berbeda. Unsur edukasi dalam *game third person* didapat saat mematikan semua api secara cepat, seperti dalam keadaan yang nyata api harus segera di padamkan sebelum membesar, semakin cepat kita mematikan api dalam *game* ini semakin baik nilai yang didapat. Sedangkan untuk level dalam *game* ini, setiap level terdapat kesulitan dari posisi api yang diletakan di tempat yang sulit di jangkau oleh pemain, sehingga pemain harus mencari jalan sendiri untuk mematikan api tersebut.

### Saran

Aplikasi *game* yang dibuat tentunya masih banyak kekurangan, adapun saran yang diberikan mengenai aplikasi ini agar di masa

yang akan datang dapat disempurnakan dan dikembangkan lebih lanjut, saran yang diberikan antara lain sebagai berikut.

Karakter utama dapat menggunakan berbagai peralatan dalam memadamkan api, karena dalam keadaan nyata banyak peralatan yang dibutuhkan untuk memadamkan api, seperti *APAR*, karung goni basah, *hydran*, sehingga dibutuhkan pengembangan dalam game ini dengan membuat objek game tersebut dan memasukan kedalam proyek.

Aplikasi *game third person* pemadam tidak terbatas pada *platform android* saja, melainkan dapat mendukung platform mobile lainnya seperti iOS, Blackberry dan Windows phone karena *software unity game engine* memiliki fitur untuk membuat aplikasi dengan berbagai platform, agar semua perangkat *handphone* maupun *desktop* dapat menginstalasi permainan ini.

Game dapat lebih interaktif dengan penambahan *healthbar* dan api yang dapat mengurangi *healthbar* jika terlalu dekat dengan api, dan juga *level pressure* untuk peralatan yang digunakan karena seperti amunisi yang bisa habis jika terlalu sering digunakan.

Penambahan level yang menyerupai kejadian nyata, seperti kebakaran di gedung bertingkat di lantai atas harus menggunakan tangga penyelamat pada mobil pemadam kebakaran untuk menjangkau lokasi kebaran, selain itu perlunya tempat yang berpotensi mengalami kebakara serta menggunakan *APAR* kapasitas kecil agar

mudah digunakan, seperti rumah harus menggunakan alat-alat yang ada di dapur untuk memadamkan api. Selain untuk perancangan game, perlunya pemahaman secara teoritis bagi masyarakat yang ingin lebih mengetahui cara pembuatan *game*, karena sebuah game dapat memberikan berbagai macam unsur yang saling berkaitan dengan sebuah lingkungan. Perancangan *game third person* dapat menjadi pertimbangan untuk dimanfaatkan bagi masyarakat yang membutuhkan contoh perancangan sebuah game, oleh karena itu sebuah *game* akan lebih baik jika dikembangkan lebih sesuai dengan yang di hadapi.

## F. DAFTAR PUSTAKA

- Anggra. (2008). Memahami Teknik dasar Pembuatan Game Berbasis Flash. Yogyakarta: Gava Media.
- Anisyah. (2000). Analisa dan Desain Sistem Informasi. Yogyakarta: Andi Offset.
- Beane, A. (2012). *3D Animation Essentials*. Hoboken: John Willey & Sons.
- Blackman, S. (2011). *Beginning 3D Game Development with Unity*. New York: Apress.
- Darmawan H., E. (2010). Pemrograman Dasar C-JAVA-C#. Bandung: Informatika Bandung.
- Hadi, P, A. (2018). Perancangan Aplikasi Informasi Multimedia Tanaman Buah Kebun Raya Bogor Dengan Augmented Reality. *Narada: Jurnal Desain dan Seni*. Vol 5(3).
- Fowler, M. (2003). *UML Distilled: A Breaaf Guide to the Standard Object Modelling*



- Language, Third Edition*. Boston: Addison-Wesley.
- Menggunakan Unity 3d Berbasis Desktop Dan Android. *Jurnal Informasi Vol. 8(1)*.
- Gunawan, D., Atthariq & Aswandi. (2017) Meningkatkan Behaviour Npc Pada Game 3d Survival Menggunakan Metode Markov. *Jurnal Infomedia. Vol. 2 (1)*
- Gunadi, A., & Al Fatta, H. (2012). Analisis Dan Pembuatan Game “Petualangan Si Argo” Berbasis Flash. *Jurnal DASI, Vol. 13(1)*.
- Jackson, W. (2012). *Android Apps for Absolute Beginners, Second Edition*. New York: Apress.
- Jogiyanto, H. (1999), Analisis & Desain Sistem Informasi, Yogyakarta: Andi Offset.
- Kipper, G & Joseph R. (2013). *Augmented Reality: An Emerging Technologies Guide to AR*. Waltham: Elsevier.
- Kusuma, R, M, P. (2015). Perancangan Game First Person Shooter 3D ‘Zombie Hunter’ dengan Menggunakan Metode A\*. *JIntech, Vol 3 (1)*.
- Munawar. (2005). *Pemodelan Visual dengan UML*. Yogyakarta: Graha Ilmu.
- Pressman, R.S. (2010). *Software Engineering: A Practitioner’s approach, Seventh Edition*. New York: McGraw-Hill Companies.
- Rizky, D. (2009). *Pengantar Ilmu Komputer*. Surabaya: INDAH.
- Roedavan, R. (2014). *Unity Tutorial Game Engine*. Bandung: Informatika.
- Santoso, B., & Purnama, B. E. (2011). Perancangan Studio Mini Berbasis Multimedia Universitas Surakarta. *Speed-Sentra Penelitian Engineering dan Edukasi, Vol.2(2)*.
- Trisna, P. (2016). Rancang Bangun Game 3d “Ena Burena” Dengan Algoritma A\* Dan Collision Detection

