

**PEMBUATAN GAME PERSONAL COMPUTER “WIZAN FIGHTER”
MENGUNAKAN PEMOGRAMAN JAVA
DENGAN LIBRARY GTGE**

Tugas Akhir

untuk memenuhi sebagian persyaratan mencapai gelar Ahli Madya
pada jenjang Diploma III jurusan Teknik Informatika



Disusun oleh :

WINARTI

07.41.2453

YANU ARSAPTO YUWONO

07.41.2455

**JURUSAN TEKNIK KOMPUTER DAN JARINGAN
SEKOLAH TINGGI MANAJEMEN INFORMATIKA DAN KOMPUTER
AMIKOM
YOGYAKARTA
2010**

NASKAH PUBLIKASI

PEMBUATAN GAME PERSONAL COMPUTER
"WIZAN FIGHTER" MENGGUNAKAN
PEMOGRAMAN JAVA DENGAN
LIBRARY GTGE

disusun oleh

Winarti 07.41.2453
Yanu Arsapto Yuwono 07.41.2455

Dosen Pembimbing



Emha Taufiq Luthfi, S.T., M.Kom.
NIK. 190302125

Ketua Jurusan
Teknik Informatika




Sudarmawan, MT
NIK. 190302035

**MAKING PC GAME “WIZAN FIGHTER”
USING JAVA PROGRAMMING
WITH GTGE LIBRARY**

**PEMBUATAN GAME PERSONAL COMPUTER “WIZAN FIGHTER”
MENGUNAKAN PEMOGRAMAN JAVA
DENGAN LIBRARY GTGE**

**Winarti
Yanu Arsapto Yuwono**

**Jurusan Teknik Informatika
STMIK AMIKOM YOGYAKARTA**

ABSTRACT

Authors make this PC game because the game in Indonesia 2D is rare, one reason to buy the game so that not a few people who use pirated games. The author aims to reduce the proficiency level in these things because the game "Wizan Fighter" is free.

This is thesis of author discusses for Making Games PC "Wizan Fighter" using Java Programming Language with Library GTGE. The authors also use the Golden T Game Engine (GTGE) a cross-platform game programming library using java language. The author also uses Adobe Photoshop for creating images that are processed into sprites is a set of several images in one file. The image files are called in sequence when going into images as if moving.

Wizan Fighter game may be used for players skill and intelligence moreover this is one game that can be used for refreshing facilities. This game can be obtained free.

Keywords: game, GTGE, java, programming.

1. PENDAHULUAN

Game merupakan proses komunikasi interaktif antara pemain dengan game tersebut. Game dapat menumbuhkan ketangkasan dan kecekatan dalam mengatasi masalah yang dihadapi di dalam game. Game juga merupakan media melatih kecerdasan seseorang dalam memecahkan masalah dengan cepat dan tepat. Selain itu dalam bermain game dibutuhkan kesabaran dan keuletan untuk mencapai keberhasilan.

Game “Wizan Fighter” ini dirancang dengan Netbeans 6.8 dengan tambahan engine Golden T Game Engine (GTGE) yang merupakan 2D Game Library/ Game SDK (Software Development Kit) untuk membuat game yang berkualitas. GTGE merupakan suatu pustaka lapisan ingkat tinggi (High Level Interface), artinya programmer tidak perlu khawatir perintah tingkat rendah lagi.

GTGE ini menfokuskan untuk membuat suatu engine yang benar – benar mudah untuk digunakan namun tetap fleksibel dan tangguh untuk membuat game. Game ini dirancang untuk anak – anak mulai usia 8 tahun keatas. Game ini dirancang untuk melatih ketrampilan dalam bermain game dan untuk sarana refreshing.

2. LANDASAN TEORI

Dalam penyusunan tugas akhir ini penulis menggunakan beberapa metode, adapun metode tersebut sebagai berikut:

- **Metode Pengumpulan Data**

Metode Kepustakaan

Pengumpulan data yang diperoleh dengan cara membaca dan mempelajari permasalahan yang ada dari buku – buku yang berhubungan dengan masalah yang dihadapi.

Metode Pengambilan data

Pengumpulan gambar, sprites, musik dan suara dilakukan dengan cara mengambil data dari internet dan lain - lain.

- **Analisa**

Penentuan masalah

Dari latar belakang di atas dapat dirumuskan masalah sebagai berikut:
Bagaimana cara membangun game “Wizan Fighter” menggunakan pemograman java dengan library GTGE?

Penentuan ruang lingkup masalah

Dalam aplikasi game Wizan Fighter akan dilakukan beberapa batasan masalah sebagai berikut:

1. Game ini hanya dapat dimainkan paling banyak 2 player.
2. Game “Wizan Fighter ini terdiri dari 5 level.
3. Game ini dibangun dengan tambahan engine Golden T Game Engine dan hanya dengan sistem operasi Windows, Linux, Solaris, Mac OS.

Penentuan tujuan penelitian

Adapun tujuan penyusunan tugas akhir ini adalah untuk membuat aplikasi game yang dapat membantu melatih ketangkasan, kecerdasan dan imajinasi untuk mengalahkan musuh dalam game tersebut.

Penentuan manfaat penelitian

Penyusunan tugas akhir ini diharapkan dapat bermanfaat bagi beberapa pihak, antara lain:

1. Bagi penulis

Penulis dapat meningkatkan kemampuan dalam membuat program dan perancangan game.

2. Bagi pemain

Pemain dapat mengasah ketrampilan, kecerdasan dan keuletan dalam memecahkan masalah yang ada di dalam game.

3. HASIL PENELITIAN

Langkah – langkah membuat game

Awal mula membuat game dengan GTGE adalah mensubclass `Game` class (`Game` class berada di package `com.golden.gamedev`), atau dengan kata lain setiap game merupakan turunan dari `Game` class. Setiap turunan `Game` class wajib melaksanakan 3 tugas, yakni :

- Contoh kode sederhana pada java

```
public class Hello
{
    /**
     * My first java program
     */
    public static void main (String[] args){
        // menampilkan string "Hello world" pada layar
        System.out.println("Hello world!");
    }
}
```

- Inisialisasi Variabel Game

```
file :: YourGame.java

// GTGE API
import com.golden.gamedev.Game;

// Java Foundation Classes (JFC)
import java.awt.Graphics2D;

public class YourGame extends Game {

    public void initResources() {
        // initialization of game variables
    }

    public void update(long elapsedTime) {
        // updating the game variables
    }
}
```

```

    }

    public void render(Graphics2D g) {
        // rendering to the screen
    }
}

```

- Menampilkan Mode Windowed

```

public static void main(String[] args) {
    GameLoader game = new GameLoader();
    game.setup(new YourGame(), new Dimension(640,480), false);
    game.start();
}
}

```

- Menampilkan Mode Fullscreen

```

public static void main(String[] args) {
    GameLoader game = new GameLoader();
    game.setup(new YourGame(), new Dimension(640,480), true);
    game.start();
}
}

```

```

class :: GameLoader

```

Syntax:

```

    public void setup(Game game,
                      Dimension d,
                      boolean fullscreen);
    public void start();

```

dimana :

game = objek game (subclass dari Game class)

d = ukuran game

fullscreen = gunakan mode fullscreen atau tidak (windowed mode)

Misal:

menginisialisasi YourGame.java dengan ukuran 640x480, mode fullscreen

```
GameLoader game = new GameLoader();
game.setup(new YourGame(), new Dimension(640,480), true);
game.start();
```

- Menampilkan Mode Applet

file :: YourGameApplet.java

```
// GTGE
import com.golden.gamedev.Game;
import com.golden.gamedev.GameLoader;

public class YourGameApplet extends GameLoader {

    protected Game createAppletGame() {
        return new YourGame();
    }

}
```

Untuk melekatkan game ke halaman html Anda, tambahkan applet tag berikut

file :: webpage (.html,.php,...)

Syntax:

```
<applet code="YourGameApplet.class"
        archive="[your_game_lib],[GTGE_lib]"
        width="[width]" height="[height]">
</applet>
```

dimana :

```
your_game_lib    = jar dari game (.jar)
GTGE_lib         = pustaka GTGE (.jar)
width            = lebar game
```

height = tinggi game

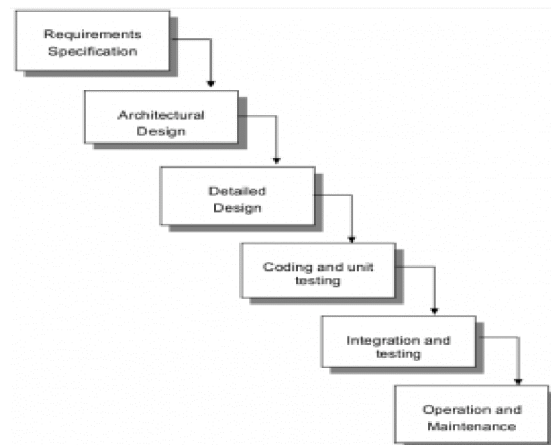
Misalnya menampilkan YourGame.java ke halaman html dengan ukuran 640x480 menggunakan GTGE v0.2.0

```
<html>
  <applet code="YourGameApplet.class"
          archive="yourgamearchive.jar, golden_0_2_0.jar"
          width="640" height="480">
  </applet>
</html>
```

4. DESIGN APLIKASI

4.1 Perancangan arsitektur system

Proses perancangan arsitektur system menggunakan metode waterfall, yaitu salah satu pendekatan bagaimana membuat sebuah perangkat lunak secara teratur, mulai dari mengumpulkan kebutuhan yang diperlukan, analisis, design, perancangan, coding, testing sampai maintenance.



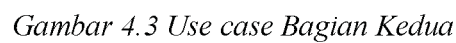
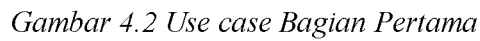
Gambar 4.1 Perancangan arsitektur system

4.2 Unified Modelling Language (UML)

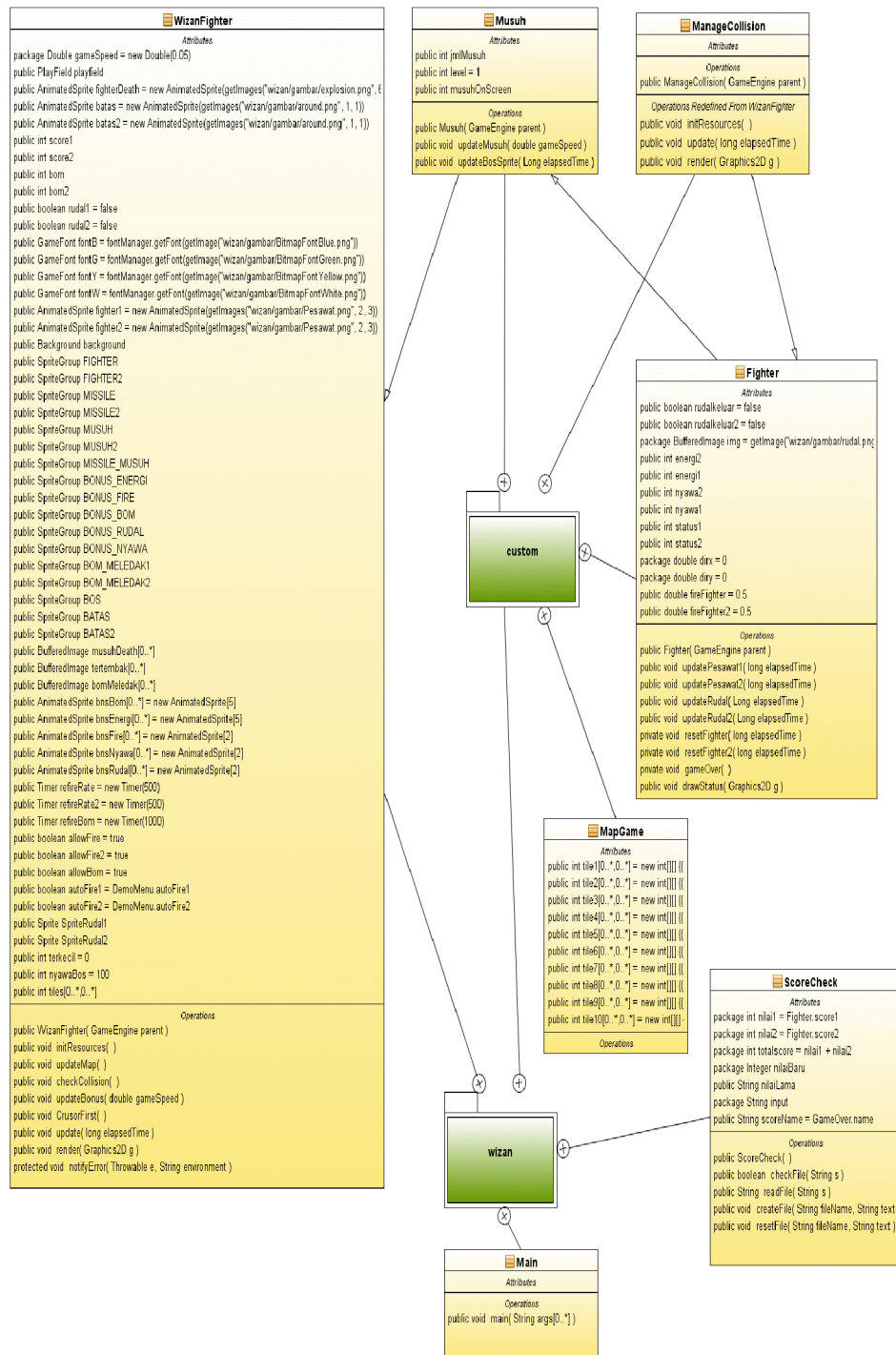
- **Usecase Diagram.**

Diagram Use menunjukkan fungsionalitas suatu sistem atau kelas dan bagaimana sistem tersebut berinteraksi dengan dunia luar dan menjelaskan sistem secara fungsional yang terlihat user.

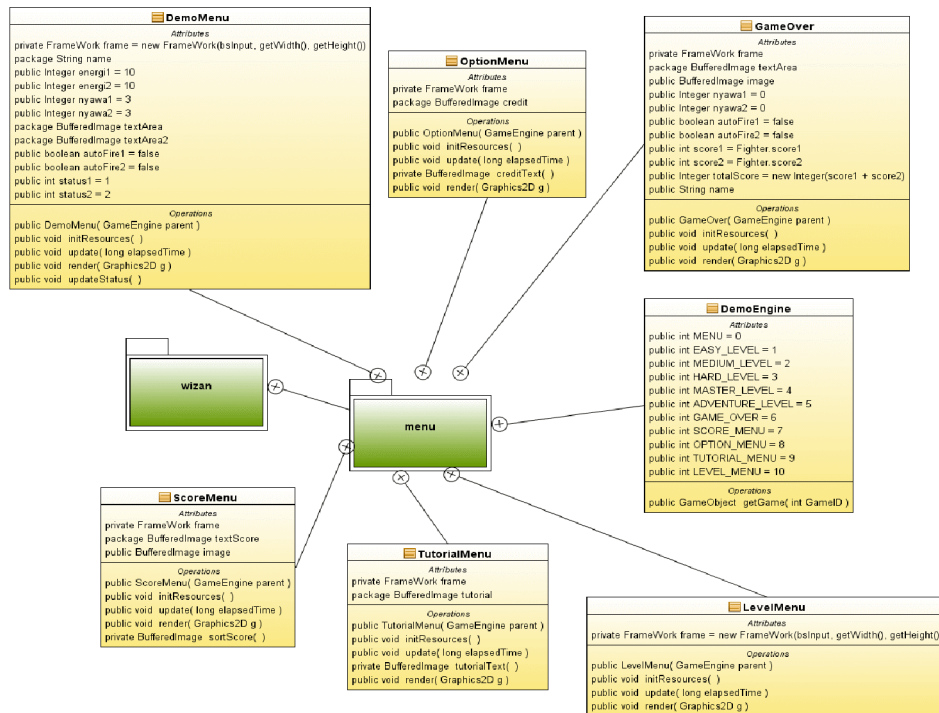
1. Memainkan game, actor mengubah karakter dari object game yang dimainkan, actor di haruskan memilih jumlah player dan level.
2. Menembak musuh, object dapat menembak musuh dan actor dapat memilih tembakan otomatis.
3. Mengeluarkan bom dan rudal, actor dapat membuat fighter mengeluarkan bom atau rudal jika sudah mendapat bonus bom atau rudal.
4. Mati, permainan selesai jika nyawa fighter mati.



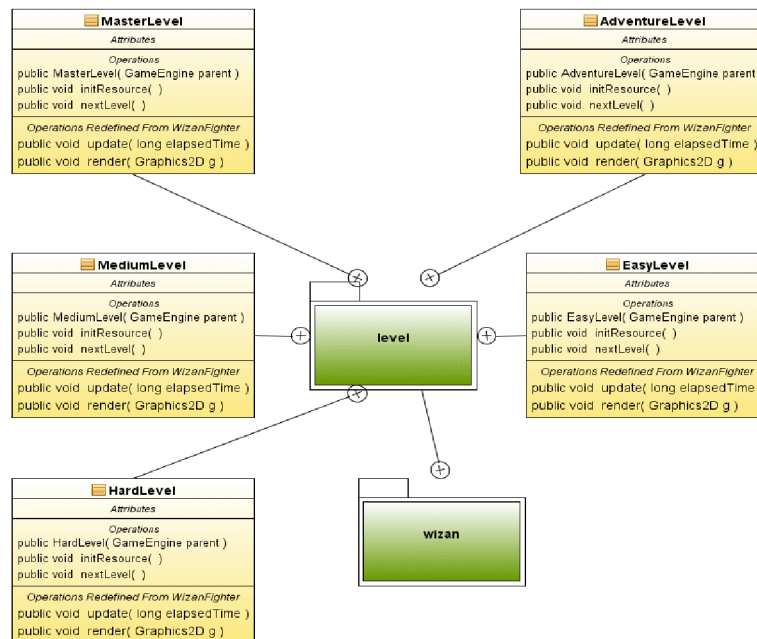
- **Class Diagram**



Gambar 4.4 Class diagram pertama



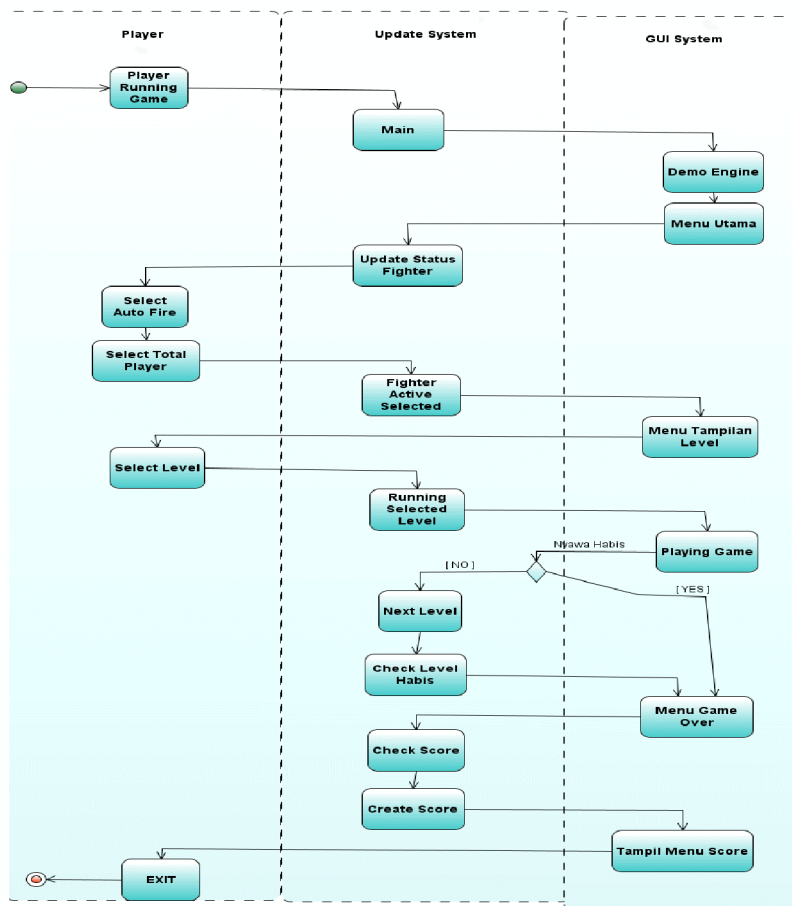
Gambar 4.5 Class diagram kedua



Gambar 4.6 Class diagram ketiga

Class diagram merupakan alat terbaik dalam perancangan perangkat lunak. Class diagram membantu pengembang mendapatkan struktur sistem dan menghasilkan rancangan sistem yang baik. Ada cukup banyak notasi dari Class diagram.

- **Activity Diagram**

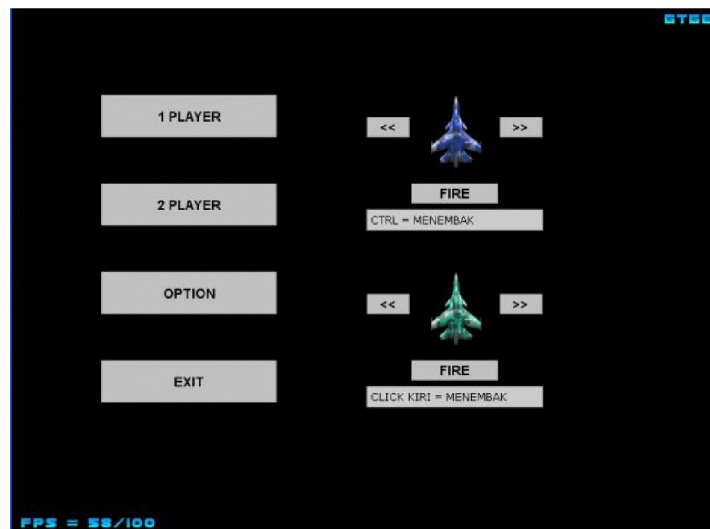


Gambar 4.7 Activity diagram pertama

Activity diagram diatas menerangkan alur dari sebuah system permainan, dimana saat player atau pengguna menjalankan sistem (game), sistem akan masuk ke Main yang mengatur mode tampilan atau dimensi layar kemudian masuk ke Demo Engine yang mengatur pemanggilan ke menu utama. Di menu utama ini ada tiga cabang update status untuk mengganti karakter dari object yang dimainkan, auto fire untuk menjalankan tembakan otomatis dan select total player untuk menentukan jumlah pemain. Setelah itu system akan masuk ke menu level untuk menentukan tingkatan level permainan, kemudian dalam permainan jika musuh habis sedangkan nyawa fighter masih ada akan naik ke level selanjutnya, tetapi jika nyawa atau level sudah habis system akan masuk ke menu game over untuk menampilkan score yang diperoleh, jika user memilih untuk menyimpan score system memeriksa file score, menyimpan score kemudian di tampilkan di menu score.

4.3 Antarmuka Aplikasi

Implementasi dan pembahasan merupakan tahapan yang bertujuan mengubah hasil dari analisis dan rancangan system menjadi bentuk nyata, dalam hal ini berupa aplikasi game yang berjalan pada piranti desktop computer. Pada saat pertama kali game dijalankan maka muncul sebuah window seperti gambar dibawah ini :



Gambar 4.9 Tampilan Menu Utama

Pada gambar menu utama diatas, button next dan prev untuk mengubah karakter pesawat sedangkan button fire digunakan untuk mengaktifkan Auto Fire (tembak otomatis). Untuk memulai permainan game pilih salah satu dari button 1 player untuk game dengan satu player atau 2 player untuk game dengan dua player.

Permainan yang hanya menggunakan 1 player. Pergerakan yang dilakukan pesawat menggunakan input dari keyboard. Bagian kiri menampilkan score, bom, nyawa dan energi yang dimiliki.

Permainan yang menggunakan 2 player. Pergerakan yang dilakukan pesawat menggunakan input dari keyboard dan mouse.

Setelah musuh dalam permainan habis akan keluar bos musuh yang harus di hancurkan terlebih dahulu sebelum masuk ke level selanjutnya.

Apabila permainan selesai atau game over maka akan muncul tampilan menu game over, di dalam menu ini pemain harus memasukkan nama untuk menyimpan score yang diperoleh. Score yang di simpan merupakan penjumlahan dari score player 1 dan score player 2.

4.4 Ujicoba Aplikasi

Setelah komponen-komponen diimplementasikan dan diuji secara individual. Pengujian lebih lanjut dilakukan untuk memastikan perilaku yang benar dan tidak ada konflik penggunaan sumber daya bersama. Berikut adalah hasil dari pengujian game Wizan Fighter :

Tabel 4.1 Pengujian Game

No	Fungsi	Hasil uji
1	Edit karakter fighter	Berhasil
2	Menembak otomatis	Berhasil
3	Mengeluarkan rudal	Berhasil
4	Rudal mengejar musuh terdekat	Berhasil
5	Nyawa bertambah atau berkurang	Berhasil
6	Energi bertambah atau berkurang	Berhasil
7	Score bertambah	Berhasil
8	Read, save dan reset score	Berhasil

9	Collision peluru fighter dengan peluru musuh	Berhasil
10	Collision peluru fighter dengan musuh	Berhasil
11	Collision fighter dengan bonus	Berhasil
12	Collision fighter dengan musuh	Berhasil
13	Collision fighter dengan peluru musuh	Berhasil
14	Collision bom dengan musuh	Berhasil
15	Next level	Berhasil
16	Game over	Berhasil

Tabel 4.2 Pengujian Game Khusus Fighter 1

No	Fungsi	Hasil uji
1	Bergerak ke kanan (tombol →)	Berhasil
2	Bergerak ke kiri (tombol ←)	Berhasil
3	Bergerak ke atas (tombol ↑)	Berhasil
4	Bergerak ke bawah (tombol ↓)	Berhasil
5	Menembak (tombol ctrl)	Berhasil
6	Mengeluarkan bom (tombol space)	Berhasil

Tabel 4.3 Pengujian Game Khusus Fighter 2

No	Fungsi	Hasil uji
1	Bergerak menggunakan mouse	Berhasil
2	Menembak (klik kiri)	Berhasil
3	Mengeluarkan bom (klik kanan)	Berhasil

KESIMPULAN

Berdasarkan analisis, perencanaan, implementasi dan pembahasan yang telah dilakukan dalam pembuatan game Wizan Fighter, dapat diambil kesimpulan sebagai berikut:

1. Game Wizan Fighter ini merupakan salah satu game yang dapat digunakan untuk sarana refreshing.
2. Game Wizan Fighter dapat digunakan untuk ajang adu ketrampilan dan kecerdasan antar pemain.

SARAN

Terdapat beberapa hal yang mungkin dapat dijadikan referensi untuk pengembangan aplikasi game antara lain:

1. Pengembangan game dengan touch screen yang trend saat ini. Tetapi membutuhkan biaya besar karena harus membeli layar touch screen.
2. Membuat game 3D agar hasil lebih memuaskan tetapi membutuhkan hardware super tinggi untuk melakukan rendering objek.
3. Kurangnya logika yang lebih optimal dan baik agar game terlihat nyata dan lebih hidup.

DAFTAR PUSTAKA

Jacson, J.R. & Mccellan A.L. (1996). *Java By Example*, Yogyakarta: Penerbit Andi

Tuerah Paulus (2004). "*Powerful Game Engine*".

From <http://goldenstudios.or.id/products/GTGE/index.php> (diakses tanggal 19 April 2010)

Para Kontributor Wikipedia(2010), "*Java*," Wikipedia,

From <http://id.wikipedia.org/w/index.php?title=Java&oldid=3284629> (diakses pada 21 April, 2010).

Prinz Eugn (2008)."*Free Airplane Sprite Pack*".

From http://www.gamedev.net/community/forums/topic.asp?topic_id=495808 (diakses pada 21 April 2010).

Para Kontributor Reinerstileset (2006). "*Tutorials Around Making Graphics and Games*".

From <http://reinerstileset.4players.de/tutorialsE.html> (diakses pada 21 April, 2010).

Leon (2010). "*Membuat Class Diagram*".

From <http://zanuar.com/2010/01/membuat-class-diagram> (diakses pada 22 April 2010).