

USULAN TUGAS AKHIR

1. IDENTITAS PENGUSUL

Nama : Rendi Budiman
NRP : 5108100020
Dosen Wali : Ahmad Saikhu, S.Si, MT.

2. JUDUL TUGAS AKHIR

“Integrasi Kinect dan Unreal Development Kit Menggunakan Framework OpenNI pada Studi Kasus Game Berbasis Interaksi Gerakan”

3. LATAR BELAKANG

Perkembangan teknologi sangat berpengaruh terhadap perkembangan game, dimana perubahan akhir-akhir ini makin pesat berkembang. Pada sekitar awal dekade 80-an, perusahaan game mulai bersaing dalam memasarkan produknya. Game-game populer pada masa itu hanya dapat dimainkan oleh satu atau dua orang pemain pada sebuah konsol game. Pesawat televisi dibutuhkan sebagai media tampilan. Beberapa dekade kemudian pemain dalam sebuah game tidak dibatasi oleh jumlah maksimal. Sebuah game dapat menampung puluhan bahkan ratusan pemain sekaligus. Media yang digunakan pun bervariasi mulai dari televisi, komputer, *handphone*, dan lain sebagainya.

Pada tahun 2010, Microsoft mengenalkan kepada publik sebuah teknologi baru yang akan memberikan sensasi berbeda dalam bermain game^[1]. Pemain tidak lagi menggunakan controller tertentu melainkan menggunakan gerakan tubuhnya sendiri untuk bermain game. Teknologi ini bernama Microsoft Kinect. Peluncuran Microsoft

Kinect disambut baik oleh publik, termasuk para pengembang game. Dengan memanfaatkan kontrol dengan gerakan manusia memacu pengembang game untuk membuat game yang selama ini dibatasi oleh kemampuan kontroler konvensional.

Pengembang game mengaplikasikan teknologi *3D motion capture*, *facial recognition*, dan *voice recognition* yang ada pada Microsoft Kinect kedalam berbagai macam genre seperti olahraga, *dance*, *racing*, dan *adventure*. Setiap genre game memiliki pola pengenalan gerakan yang berbeda sehingga dibutuhkan jenis kalkulasi/perhitungan yang berbeda dari setiap output Microsoft Kinect.

Saat ini *developer* dapat mencoba menggunakan fungsionalitas Microsoft Kinect secara penuh sejak dikeluarkannya Microsoft Kinect SDK pada Juni 2011. Namun sebelum dikeluarkannya SDK resmi tersebut, sebuah framework *open-source* telah tersedia lebih awal yaitu OpenNI. OpenNI adalah framework berbasis C++ yang digunakan sebagai interface untuk memproses data dari 3D sensor seperti sensor Microsoft Kinect. Fungsi utama OpenNI digunakan untuk mendefinisikan tipe data (seperti *depth map*, *color map*, *user pose*) dan modul-modul yang dapat melakukan translasi kedalam sensor ataupun *skeleton algorithm*. Dengan memanfaatkan fungsi – fungsi tersebut mengembangkan game dengan kontrol Microsoft Kinect dapat diwujudkan.

Salah satu teknologi yang digunakan oleh pengembang game terkenal di dunia adalah Unreal Engine. Unreal Engine adalah *game engine* yang dikembangkan oleh Epic Games. Game pertama yang dibuat adalah Unreal, game *first-person shooter* pada tahun 1998. Pada perkembangannya, Unreal Development Kit kini digunakan untuk berbagai macam jenis game, seperti *stealth*, *MMORPG*, *RPG*, *Racing*, dan digunakan untuk berbagai simulasi. Game engine ini menggunakan bahasa UnrealScript, bahasa scripting yang mendukung *object-oriented programming*. Epic Games merilis Unreal Development Kit bagi developer yang ingin mengeksplorasi lebih dalam mengenai UnrealScript, maupun membuat berbagai macam game tanpa perlu mempermasalahkan lisensi^[2].

Penulis telah mengamati berbagai contoh game yang dibuat menggunakan Unreal Development Kit masih menggunakan kontrol standart seperti keyboard,

mouse, ataupun joystick. Padahal, para pengembang game dapat mencoba menggunakan Microsoft Kinect sebagai alternatif kontroler. Dengan menggunakan bantuan framework OpenNI, input yang diterima oleh Microsoft Kinect dapat melacak *skeleton* pemain yang kemudian digunakan sebagai kontroler pada lingkungan Unreal Development Kit melalui UnrealScript. Oleh karena itu, penulis mengusulkan sebuah tugas akhir integrasi Kinect dan Unreal Development Kit menggunakan framework OpenNI pada studi kasus game berbasis interaksi gerakan untuk mengeksplorasi teknologi baru Microsoft Kinect. Juga diharapkan dengan adanya game ini pemanfaatan Microsoft Kinect dapat berkembang lebih jauh.

4. TUJUAN TUGAS AKHIR

Tujuan dari pembuatan tugas akhir ini antara lain:

1. Eksplorasi teknologi OpenNI.
2. Eksplorasi teknologi Unreal Development Kit dan UnrealScript.
3. Terbentuknya sebuah integrasi antara framework OpenNI dan Unreal Development Kit, sehingga dapat diaplikasikan untuk game.
4. Membuat gamemenggunakan Unreal Development Kit dengan kontroler utama Microsoft Kinect.
5. Mengaplikasikan hukum fisika (*physics*) pada game.

5. PERMASALAHAN

Permasalahan yang dibahas dalam tugas akhir ini meliputi :

1. Bagaimana melakukan konfigurasi Microsoft Kinect agar dikenali oleh perangkat komputer menggunakan perangkat khusus *Kinect to PC USB*.
2. Bagaimana mengambil data input yang diterima oleh Microsoft Kinect menggunakan framework OpenNI.
3. Bagaimana melakukan sinkronisasi data dari hasil yang didapatkan oleh framework OpenNI agar dikenali oleh UnrealScript.
4. Bagaimana mengimplementasikan gerakan karakter dalam Unreal Development Kit sesuai dengan gerakan pemain secara *real-time*.

5. Bagaimana membuat game yang mengimplementasikan fisika (physics) dalam Unreal Development Kit.

6. BATASAN MASALAH

Batasan permasalahan dalam tugas akhir ini adalah sebagai berikut:

1. Menggunakan sistem operasi minimal Windows XP SP3 atau Windows Vista.
2. Menggunakan minimal 2GB RAM.
3. Menggunakan Kartu grafis minimal Nvidia diatas seri 400, atau ATI Radeon diatas seri 5000.
4. Menggunakan Microsoft Kinect sebagai kontroler.
5. Menggunakan dasar framework OpenNI untuk pengolahan data dari Microsoft Kinect.
6. Mengenali secara keseluruhan dari *skeleton* yang ditangkap oleh Microsoft Kinect.
7. Menggunakan Unreal Development Kit diatas versi Oktober 2010.
8. Menggunakan bahasa pemrograman C++ dan Unreal Script dengan Microsoft Visual Studio dan nFringe.
9. Menggunakan aplikasi Adobe Flash, 3DS Max, Google SketchUp, SpeedTree, dan ActorX sebagai kakas untuk pembuatan setiap aset 3D dalam game.
10. Aplikasi Game 3D ini hanya mendukung satu pemain.

7. RINGKASAN TUGAS AKHIR

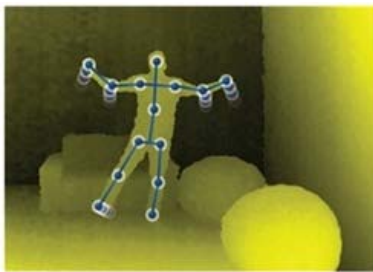
Dalam tugas akhir ini penulis mengusulkan untuk membuat integrasi antara Microsoft Kinect dan Unreal Development Kit menggunakan framework OpenNI yang diwujudkan kedalam aplikasi game 3D. Kontrol utama pada game ini tidak menggunakan keyboard ataupun mouse, melainkan menggunakan interaksi gerakan pemain (Microsoft Kinect).

Aplikasi game 3D ini memerlukan interaksi penuh dengan pemain menggunakan kontroler Microsoft Kinect. Pemain akan disimulasikan dengan karakter dalam game yang gerakannya akan sama dengan gerakan pemain aslinya. Dengan kontrol ini,

pemain akan diberi sejumlah waktu tertentu untuk menghalau serangan bola-bolayang melaju kearahnya. Apabila bola mengenai tubuh dari karakter maka poin akan bertambah, sebaliknya apabila bola meleset maka poin akan berkurang. Game ini menggunakan *physics* untuk melakukan kalkulasi natural seperti tumbukan, pantulan, dan kecepatan.

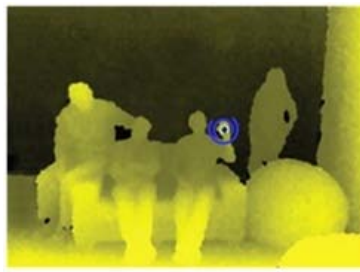
Agar game ini dapat dikontrol menggunakan Microsoft Kinect, diperlukan bantuan dari framework OpenNI. Framework ini dapat menangkap data dari Microsoft Kinect untuk kemudian diambil sesuai dengan spesifikasi yang diinginkan. Hasil output dari framework OpenNI ini dapat berupa analisa tubuh keseluruhan, analisa titik tangan (*hand point*), deteksi gerakan, generator map, dan lain sebagainya^[3].

Berikut adalah beberapa hasil deteksi menggunakan framework OpenNI.



Identifikasi figure

Output berupa lokasi saat ini dan orientasi dari sendi (*joint*) dari figure(*body data*)



Lokasi dari tangan user

Output dapat berupa telapak tangan (*hand point*)

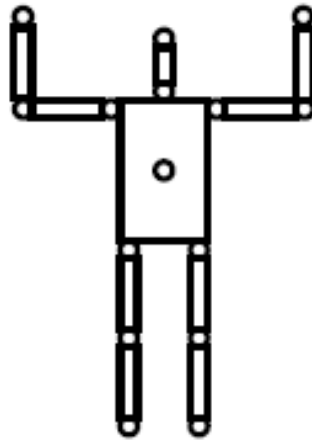


Identifikasi gerakan tangan

Output berupa *alert* ke aplikasi bahwa gerakan tangan telah terjadi.

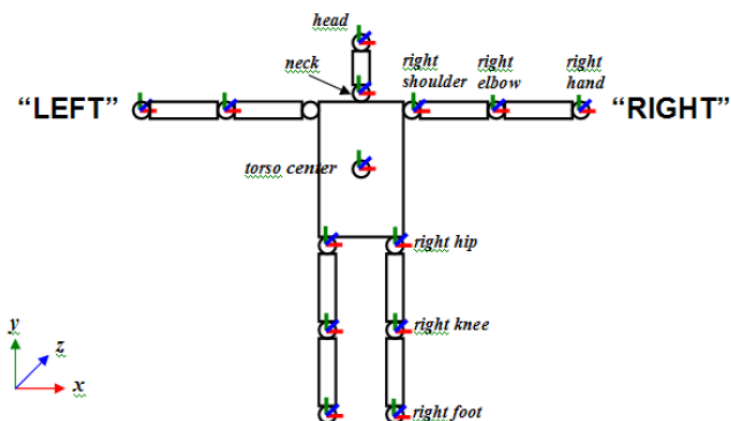
Gambar 1. Contoh deteksi menggunakan framework OpenNI

Sebelum gerakan dari pemain (pengguna) diterima, terlebih dahulu akan dilakukan proses kalibrasi, yaitu proses untuk mengatur *skeleton model* dari proporsi tubuh pengguna. Kalibrasi diperlukan sebelum proses *tracking* dimulai^[4]. Proses Kalibrasi akan berjalan ketika pengguna melakukan pose berikut.



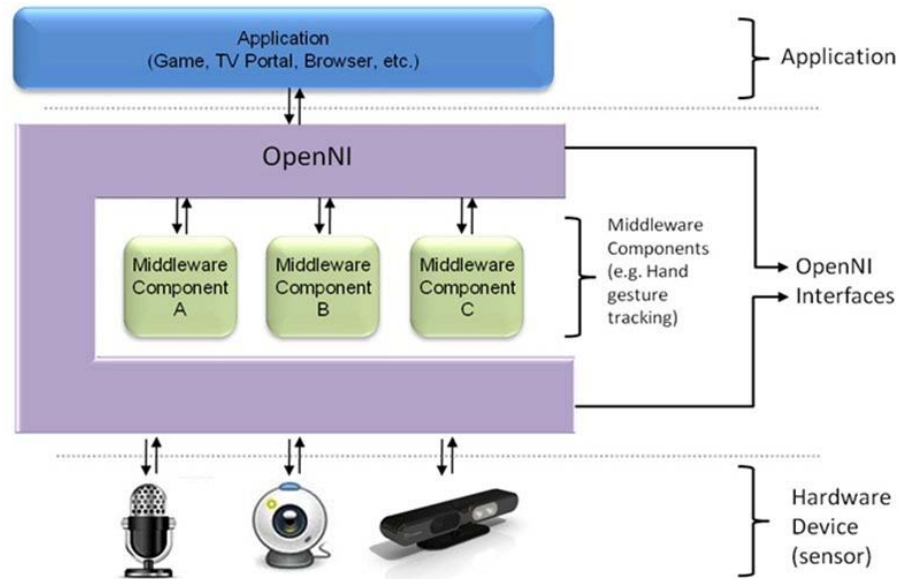
Gambar 2. Ilustrasi pose kalibrasi pada framework OpenNI

Framework ini kemudian akan mengembalikan (*return*) posisi dan orientasi dari sendi-sendi *skeleton*. Algoritma PrimeSense NITE 1.3 digunakan pada framework OpenNI^[5]. Data ini nantinya akan digunakan untuk proses selanjutnya sesuai dengan kebutuhan.



Gambar 3. Ilustrasi deteksi sendi

Secara garis besar, struktur dari framework OpenNI terbagi menjadi 3 *layer* sebagai berikut.



Gambar 4. Struktur framework OpenNI

Dari diagram diatas terbagi menjadi 3 *layer* yaitu :

a. *Hardware Device*

Merupakan *layer* terbawah pada susunan struktur framework OpenNI yang berisi hardware yang akan digunakan seperti mic, kamera, serta kamera sensor 3D (*Microsoft Kinect*).

b. *OpenNI Interfaces*

Layer tengah ini berisi sekumpulan fungsi yang dapat dijalankan untuk menerima input dari hardware untuk kemudian diolah.

c. *Application*

Merepresentasi software yang mengimplementasi interaksi natural dari framework OpenNI.

Pada tahapan integrasi, source code yang dibuat pada framework OpenNI akan dikenalkan pada *package files* dalam Unreal Development Kit. Kemudian sendi-sendi *skeleton* akan disinkronisasi dengan UnrealScript^[6]. Dengan metode diatas memungkinkan sistem Unreal Development Kit untuk menangkap data secara *real-time* dari Microsoft Kinect.

Diagram alur integrasi antara Microsoft Kinect dan Unreal Development Kit digambarkan dalam tahap input, tahap proses, dan tahap output.

Input integrasi ini berupa data mentah (*raw*) yang dihasilkan oleh Microsoft Kinect. Data ini berupa gambar RGB, gambar dengan informasi kedalaman (*depth image*), dan aliran audio.

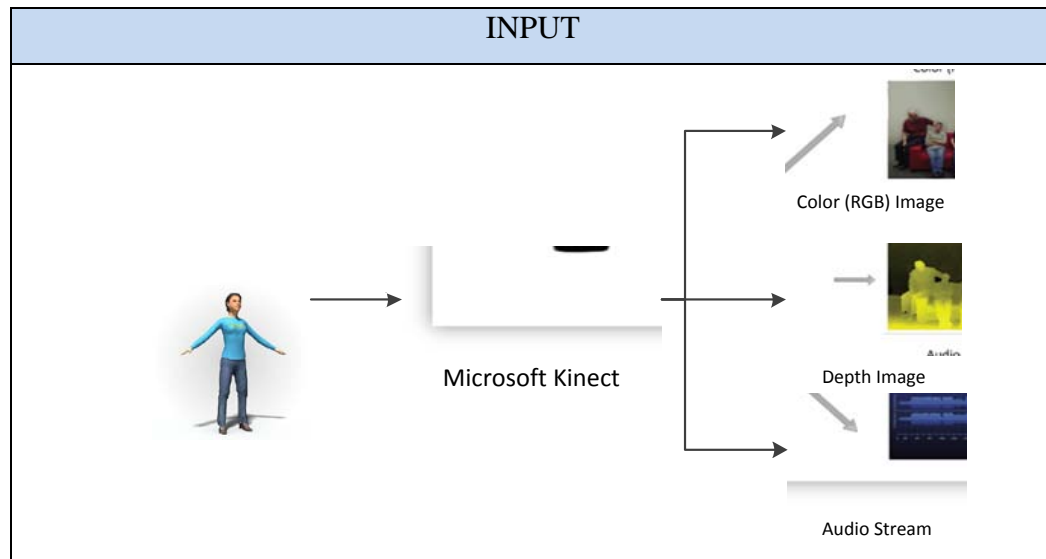


Diagram 1. Alur input integrasi

Proses integrasi berupa pengolahan data gambar RGB dan gambar dengan informasi kedalaman (*depth image*) dengan berbagai metode algoritma NITE yang ada dalam framework OpenNI. Data hasil output framework OpenNI digunakan untuk mengisi set AnimTree pada Unreal Development Kit. AnimTree adalah sebuah fungsi animasi yang dapat dikustomisasi oleh developer[7]. Data AnimTree ini akan dipasang pada objek karakter dalam game, yang biasa disebut dengan SkeletalMesh. SkeletalMesh adalah object 3D pada Unreal Development Kit yang memiliki struktur *skeleton* dan sendi-sendi layaknya manusia[8].

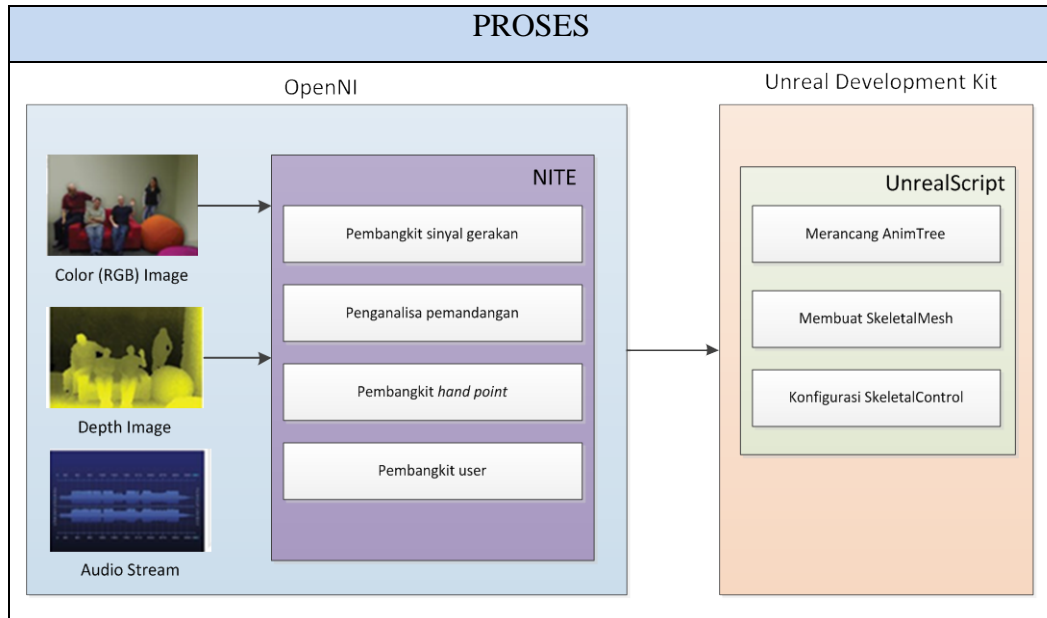


Diagram 2. Alur proses integrasi

Output integrasi ini berupa gerakan karakter dalam game secara *real-time* sesuai dengan gerakan pemain sebenarnya. Dengan terbentuknya sistem kontrol ini akan diterapkan pada aplikasi game 3D.

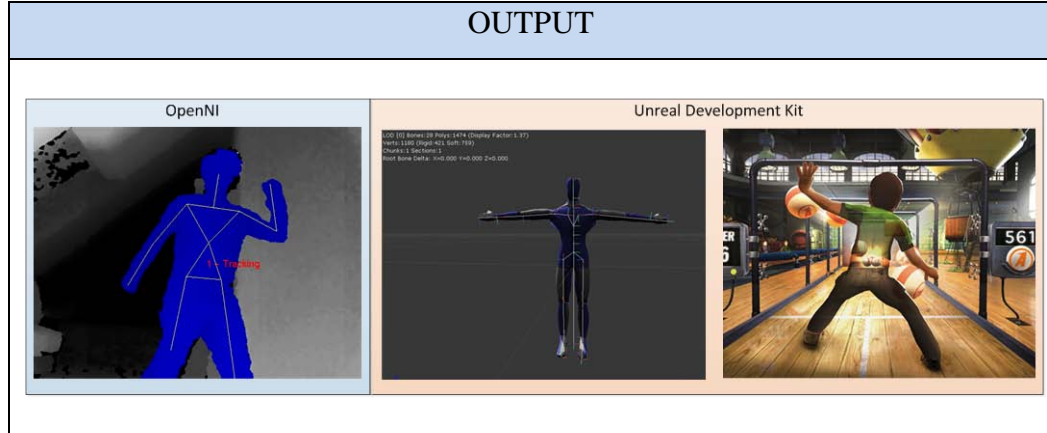


Diagram 3. Alur output integrasi

Pembuatan aset 3D dalam aplikasi game ini menggunakan kakas bantuan 3Ds MAX, SpeedTree, dan Google SketchUp. Kemudian aset ini akan diimpor ke dalam Unreal Development Kit ke dalam format .fbx maupun .jpg untuk tekstur aset. User Interface dalam game seperti timer, skor, splash screen, dibuat menggunakan ActionScript dalam Adobe Flash.

Gambaran mengenai rancangan arsitektur aplikasi game yang akan dibuat adalah sebagai berikut.



Gambar 4. Arsitektur Aplikasi Game

8. METODOLOGI

1. Analisis Kebutuhan dan Studi Literatur

Pada tahap ini dilakukan pengumpulan dan penggalian informasi dan literatur yang diperlukan dalam proses perancangan dan implementasi sistem yang akan dibangun. Literatur yang dicari meliputi cara mengintegrasikan Microsoft Kinect dengan framework OpenNI, dokumen tentang algoritma pada framework OpenNI, cara membuat karakter dan aset 3D dalam 3Ds MAX dan meng-*import* kedalam Unreal Development Kit, cara melakukan sinkronisasi antara framework OpenNI dengan UnrealScript, cara menggunakan Action Script pada Adobe Flash untuk merancang user interface dalam game, menerapkan *physics* (fisika) di dalam Unreal Development Kit.

2. Perancangan Sistem

Pada tahap ini dilakukan analisa awal dan pendefinisian kebutuhan sistem untuk mengetahui masalah yang sedang dihadapi. Dimulai dari merancang keseluruhan sistem, kelas-kelas beserta fungsi-fungsinya, koneksi, struktur data, sistem *rendering*, sehingga dapat memecahkan rumusan masalah yang diangkat.

3. Implementasi

Pada tahap ini dilakukan implementasi sistem secara menyeluruh berdasar pada desain sistem dan tampilan perangkat lunak. Bahasa pemrograman yang digunakan adalah C++ (framework OpenNI) dan UnrealScript (Unreal Development Kit). Implementasi aset 3D dalam game menggunakan aplikasi 3Ds MAX, SpeedTree, dan Google SketchUp. User Interface game menggunakan Adobe Flash.

4. Uji coba dan evaluasi

Pada tahap ini akan dilakukan pengujian terhadap integrasi yang telah dibuat antara Microsoft Kinect dan Unreal Development Kit menggunakan framework OpenNI. Uji coba dan evaluasi perangkat dilakukan melalui dua tahap. Tahap pertama integrasi akan diujicoba langsung menggunakan program sederhana pada framework OpenNI untuk menampilkan keseluruhan pemindaian *skeleton* maupun gerakan pada pemain (user). Tahap kedua adalah membandingkan

pindaian pada framework OpenNI dengan tampilan gerakan karakter pada game. Penulis menggunakan ujicoba menggunakan metode *black box testing*^[9]. Evaluasi keberhasilan integrasi ini adalah hasil pemindaian pada program OpenNI dan pemindaian karakter pada gamemenghasilkan gerakan yang sama.

5. Penyusunan Buku Tugas Akhir

Pada tahap ini melakukan pendokumentasian dan laporan dari seluruh konsep, dasar teori, implementasi, proses yang telah dilakukan, dan hasil-hasil yang telah didapatkan selama pengerjaan tugas akhir. Buku tugas akhir ini bertujuan untuk mendapatkan gambaran dari pengerjaan tugas akhir ini dan diharapkan dapat berguna untuk pembaca yang tertarik untuk melakukan pengembangan lebih lanjut.

Secara garis besar, buku tugas akhir nantinya terdiri atas beberapa bagian yaitu :

1. Pendahuluan
 - 1.1 Latar Belakang
 - 1.2 Permasalahan
 - 1.3 Batasan Tugas Akhir
 - 1.4 Tujuan
 - 1.5 Metodologi
 - 1.6 Sistematika Penulisan
2. Tinjauan Pustaka
3. Desain dan Implementasi
4. Uji Coba dan Evaluasi
5. Kesimpulan dan Saran
6. Daftar Pustaka

9. JADWAL Pengerjaan Tugas Akhir

Tugas akhir ini diharapkan bisa dikerjakan menurut jadwal sebagai berikut:

Tahapan	Bulan (Tahun 2012)															
	Maret				April				Mei				Juni			
Analisa kebutuhan dan studi literatur																
Perancangan Sistem																
Implementasi																
Ujicoba dan Evaluasi																
Penyusunan Buku Tugas Akhir																

10. DAFTAR PUSTAKA

- [1] Microsoft. (t.thn.). *Kinect for Windows*. Dipetik Februari 16, 2012, dari Microsoft: <http://www.microsoft.com/en-us/kinectforwindows/>
- [2] Games, E. (2009). Dipetik Februari 15, 2012, dari Unreal Development Kit: <http://udk.com/>
- [3] OpenNI. (2010). *OpenNI Documentation: Overview*. Dipetik Februari 20, 2012, dari OpenNI: <http://openni.org/Documentation/ProgrammerGuide.html>
- [4] OpenNI. (2010). Prime Sensor NITE 1.3 Algorithms notes. 14.
- [5] OpenNI. (2010). Prime Sensor NITE 1.3 Framework Programmer's Guide. 14.
- [6] Jared. (2011, Maret 08). *Kinect Motion Capture Animated*. Dipetik Februari 21, 2012, dari kinect.dashhacks.com: <http://kinect.dashhacks.com/kinect-news/2011/03/08/kinect-motion-capture-animated-series-under-hud>
- [7] Wraiyth. (2011, Januari 29). *AnimTree Crash Course*. Retrieved Februari 21, 2012, from UDK Central: http://udkc.info/index.php?title=Tutorials:Wraiyth%27s_AnimTree_crash_course
- [8] UDK Central. (2011, Maret 4). *Skeletal Mesh*. Retrieved Februari 20, 2012, from UDK Central: http://udkc.info/index.php?title=Skeletal_Mesh
- [9] Williams, L. (2006). Dipetik Februari 21, 2012, dari agile.csc.ncsu.edu: <http://agile.csc.ncsu.edu/SEMaterials/BlackBox.pdf>

LEMBAR PENGESAHAN

Surabaya,Maret 2012

Menyetujui,

Dosen Pembimbing I

Dosen Pembimbing II

(Imam Kuswardayan, S.Kom., M.T.)

(NIP. 1976 1215 2003 12 1 001)

(Dwi Sunaryono, S.Kom, M.Kom)

(NIP. 19720528199702 1 001)