

**IMPLEMENTASI ALGORITMA A* PADA GAME TANK
RAMPAGE DI LINGKUNGAN XNA**

Skripsi



oleh
PRIMAREKHA PANJI SATRIA
22084426

PROGRAM STUDI TEKNIK INFORMATIKA FAKULTAS TEKNOLOGI INFORMASI
UNIVERSITAS KRISTEN DUTA WACANA
2013

IMPLEMENTASI ALGORITMA A* PADA GAME TANK RAMPAGE DI LINGKUNGAN XNA

Skripsi



Diajukan kepada Program Studi Teknik Informatika Fakultas Teknologi Informasi
Universitas Kristen Duta Wacana
Sebagai Salah Satu Syarat dalam Memperoleh Gelar
Sarjana Komputer

Disusun oleh

PRIMAREKHA PANJI SATRIA
22084426

PROGRAM STUDI TEKNIK INFORMATIKA FAKULTAS TEKNOLOGI INFORMASI
UNIVERSITAS KRISTEN DUTA WACANA
2013

PERNYATAAN KEASLIAN SKRIPSI

Saya menyatakan dengan sesungguhnya bahwa skripsi dengan judul:

IMPLEMENTASI ALGORITMA A* PADA GAME TANK RAMPAGE DI LINGKUNGAN XNA

yang saya kerjakan untuk melengkapi sebagian persyaratan menjadi Sarjana Komputer pada pendidikan Sarjana Program Studi Teknik Informatika Fakultas Teknologi Informasi Universitas Kristen Duta Wacana, bukan merupakan tiruan atau duplikasi dari skripsi kesarjanaan di lingkungan Universitas Kristen Duta Wacana maupun di Perguruan Tinggi atau instansi manapun, kecuali bagian yang sumber informasinya dicantumkan sebagaimana mestinya.

Jika dikemudian hari didapati bahwa hasil skripsi ini adalah hasil plagiasi atau tiruan dari skripsi lain, saya bersedia dikenai sanksi yakni pencabutan gelar kesarjanaan saya.

Yogyakarta, 25 Mei 2013

PRIMAREKHA PANJI SATRIA
22084426

HALAMAN PERSETUJUAN

Judul Skripsi : IMPLEMENTASI ALGORITMA A* PADA GAME
TANK RAMPAGE DI LINGKUNGAN XNA
Nama Mahasiswa : PRIMAREKHA PANJI SATRIA
N I M : 22084426
Matakuliah : Skripsi (Tugas Akhir)
Kode : TIW276
Semester : Genap
Tahun Akademik : 2012/2013

Telah diperiksa dan disetujui di
Yogyakarta,
Pada tanggal 25 Mei 2013

Dosen Pembimbing I

Dosen Pembimbing II

Rosa Delima, S.Kom., M.Kom.

Joko Purwadi, M.Kom.

HALAMAN PENGESAHAN

IMPLEMENTASI ALGORITMA A* PADA GAME TANK RAMPAGE DI LINGKUNGAN XNA

Oleh: PRIMAREKHA PANJI SATRIA / 22084426

Dipertahankan di depan Dewan Penguji Skripsi
Program Studi Teknik Informatika Fakultas Teknologi Informasi
Universitas Kristen Duta Wacana - Yogyakarta
Dan dinyatakan diterima untuk memenuhi salah satu syarat memperoleh gelar
Sarjana Komputer
pada tanggal 21 Mei 2013

Yogyakarta, 25 Mei 2013
Mengesahkan,

Dewan Penguji:

1. Rosa Delima, S.Kom., M.Kom
2. Joko Purwadi, M.Kom
3. Hendro Setiadi, M.Eng
4. Ir. Sri Suwarno, M.Eng.

DU TA WACANA



Dekan

(Drs. Wimmie Handiwidjojo, M.IT.)

Ketua Program Studi

(Nugroho Agus Haryono, M.Si)

KATA PENGANTAR

Puji dan syukur kepada Tuhan Yang Maha Kuasa, karena dengan rahmat-Nya penulis dapat menyelesaikan Laporan Tugas Akhir ini dengan baik. Penulisan tugas akhir ini ditujukan untuk memenuhi salah satu syarat memperoleh gelar Sarjana Komputer Jurusan Teknik Informatika. Penulis juga mengucapkan terima kasih kepada pihak-pihak yang telah membantu dalam menyelesaikan penulisan tugas akhir ini baik dalam memberikan bimbingan antara lain:

1. Keluarga tercinta terutama orang tua dan kakak-kakak yang telah memberikan dukungan material dan moral.
2. Rosa Delima, S.Kom.,M.Kom. dan Joko Purwadi, M.Kom., selaku dosen pembimbing tugas akhir yang telah membimbing dan membantu kepada penulis.
3. Drs. Wimmie Handiwidjojo, MIT., selaku dekan Fakultas Teknologi Informasi Universitas Kristen Duta Wacana .
4. Nugroho Agus Haryono, M.Si, selaku Ketua Program Studi Teknik Informatika.
5. Seluruh dosen Universitas Kristen Duta Wacana , khususnya dosen Teknik Informatika yang telah mengajarkan dan mmeberi bekal ilmu kepada penulis.
6. Sekretariat Fakultas Teknologi Informasi yang telah membantu dalam mengurus administrasi perkuliahan.
7. Rekan-rekan Teknik Informatika, yang selama ini belajar bersama, saling membantu, mendukung, dan mendorong penulis dalam menyelesaikan studi.
8. Pihak-pihak yang tidak dapat disebutkan satu persatu, yang telah membantu penulis.

Semoga Tuhan Tuhan Yang Maha Kuasa membalas kebaikan dan ketulusan yang telah membantu penulis dengan melimpahkan rahmat dan karunia-Nya. Saran dan kritik yang membangun sangat diharapkan dan semoga hasil ini dapat bermanfaat bagi sesama.

Yogyakarta, 14 Mei 2013

Penulis,

Primarekha Panji Satria

©UKDIN

INTI SARI

IMPLEMENTASI ALGORITMA A* PADA GAME TANK RAMPAGE DI LINGKUNGAN XNA

Dewasa ini, di mana teknologi berkembang dengan pesat, membuat teknologi *game* juga berkembang dengan pesat, terutama perkembangan metode pembuatan *game* dan *framework* yang dipakai untuk membuat *game*. Salah satu algoritma yang sering dipakai dalam membuat *game* adalah algoritma A*. Algoritma ini digunakan untuk melakukan pencarian jarak terpendek dari satu titik ke titik lain.

Berdasarkan permasalahan diatas, penulis membuat sebuah rancangan *game*. *Game* tersebut bernama *Tank Rampage*. Dalam perancangan *game* tersebut, penulis menggunakan XNA *framework* buatan *Microsoft* dan algoritma A* yang akan diterapkan pada pergerakan musuh komputer sehingga musuh komputer akan bisa bergerak lebih cerdas dalam mengejar *player*. Algoritma A* menggunakan fungsi heuristik sehingga hasilnya lebih optimal daripada algoritma *pathfinding* yang tidak menggunakan fungsi heuristik, misalnya algoritma *greedy*.

XNA *framework* menyediakan fungsi-fungsi dan variabel untuk membantu proses pembuatan *game*, terutama dalam proses implementasi algoritma A* dan *greedy*. Hasil yang diamati dalam kasus ini adalah dalam *map* dengan tembok, Tembok berguna untuk membedakan jalur algoritma A* dan *greedy*, musuh dengan algoritma A* akan bisa mendeteksi tembok penghalang sehingga jarak terpendek bisa langsung ditemukan, berbeda dengan algoritma *greedy* yang harus menabrak tembok dulu, baru berbelok mencari jalan.

Keywords: *Artificial Intelligence, PathFinding Algorithm, heuristik, A*, game, XNA Framework*

DAFTAR ISI

HALAMAN JUDUL.....	
PERNYATAAN KEASLIAN SKRIPSI	iii
HALAMAN PERSETUJUAN.....	iv
HALAMAN PENGESAHAN	v
KATA PENGANTAR	vi
INTI SARI.....	vii
DAFTAR ISI.....	ix
DAFTAR TABEL.....	xi
DAFTAR GAMBAR.....	xii
DAFTAR LAMPIRAN.....	xv
BAB 1 PENDAHULUAN.....	1
1.1 Latar Belakang Masalah.....	1
1.2 Rumusan Masalah.....	1
1.3 Batasan Masalah	2
1.4 Hipotesis	2
1.5 Tujuan Penelitian.....	2
1.6 Metode	3
1.7 Sistematika Penulisan	4
BAB 2 TINJAUAN PUSTAKA	5
2.1 Tinjauan Pustaka.....	5
2.2 Landasan Teori.....	7
2.2.1 Algoritma A*	7
2.2.2 Heuristik.....	9
2.2.3 XNA Game Studio.....	12

BAB 3 ANALISIS DAN PERANCANGAN SISTEM.....	15
3.1 Kebutuhan Perangkat Keras dan Lunak	15
3.2 Perancangan Skenario Game	15
3.3 Perancangan Antarmuka Game.....	16
3.3.1 Perancangan <i>Game Map</i> dan Screen.....	16
3.3.2 <i>Spritesheet</i> yang dipakai	16
3.3.3 Perancangan Antarmuka Gameplay.....	18
3.4 Perancangan Peraturan Game	19
3.5. <i>Pseudocode</i> Algoritma A* Pada <i>Game</i>	20
3.6. <i>FlowChart</i> Algoritma A* pada <i>Game</i>	24
BAB 4 IMPLEMENTASI DAN ANALISIS.....	28
4.1 Implementasi.....	28
4.2 Implementasi Algoritma A* pada <i>Game</i>	32
4.3 Studi Kasus A* dengan Map dan Jumlah Markas Statis	36
4.3.1 Implementasi Nilai Heuristik pada Algoritma A*	39
4.3.2 Perbandingan Jalur Algoritma A* dengan Algoritma <i>Greedy</i>	42
4.3.3 Perbandingan Waktu Algoritma A* dengan Algoritma <i>Greedy</i>	46
4.3.4 Perbandingan Jumlah <i>Node</i> Algoritma A* dengan Algoritma <i>Greedy</i>	50
BAB 5 KESIMPULAN DAN SARAN	53
5.1 Kesimpulan.....	53
5.2 Saran	53
DAFTAR PUSTAKA	54

DAFTAR TABEL

Tabel 4.1 Tabel perbandingan waktu tempuh musuh mencapai lokasi player antara algoritma <i>greedy</i> dengan algoritma A*	49
Tabel 4.2 Tabel daftar <i>node</i> dan jumlah <i>node</i> pada masing masing contoh kasus	52

©UKDW

DAFTAR GAMBAR

Gambar 1.1 <i>Iterative Software Life Cycle</i>	3
Gambar 2.1 <i>Pseudocode Algoritma A*</i>	8
Gambar 2.2 <i>Manhattan Distance</i> antara dua titik	9
Gambar 2.3 <i>Euclidian Distance</i> antara dua titik	10
Gambar 2.4 <i>Diagonal Distance</i> antara dua titik	10
Gambar 2.5 Diagram alir pemanggilan fungsi <i>XNA framework</i>	14
Gambar 3.1 <i>Sprite Sheet</i> yang digunakan	17
Gambar 3.2 Perancangan tampilan state awal <i>game</i>	18
Gambar 3.3 Perancangan tampilan state <i>game-play</i>	18
Gambar 3.4 Perancangan tampilan state <i>game-over</i>	19
Gambar 3.5 <i>Pseudocode Algoritma A*</i>	20
Gambar 3.6 <i>Pseudocode method AddNodeKeOpenList()</i>	21
Gambar 3.7 <i>Pseudocode method FindSuccessorNodes()</i>	21
Gambar 3.8 <i>Pseudocode method FindPath()</i>	23
Gambar 3.9 <i>Flowchart method AddNodeKeOpenList()</i>	24
Gambar 3.10 <i>Flowchart method FindSuccessorNode()</i>	25
Gambar 3.11 <i>Flowchart method FindPath()</i>	27
Gambar 4.1 <i>Title screen game tank rampage</i>	28
Gambar 4.2 <i>Game play tank rampage</i>	29
Gambar 4.3 Skor bertambah 10 saat musuh tertembak dan meledak	30
Gambar 4.4 Markas komputer berkurang satu saat markas dilewati	30
Gambar 4.5 Kondisi saat satu level selesai	31

Gambar 4.6 Penggambaran Jalur A*	32
Gambar 4.7 Potongan <i>code method</i> FindPath()	32
Gambar 4.8 Potongan <i>code method</i> AddNodeKeOpenList()	34
Gambar 4.9 Potongan <i>code method</i> FindAdjacentNode()	34
Gambar 4.10 Penggambaran jalur A* mengikuti player	37
Gambar 4.11 Penggambaran jalur A* langsung bisa melewati tembok	37
Gambar 4.12 Potongan <i>code</i> pembuatan jalur baru	38
Gambar 4.13 Penggambaran jalur A* yang langsung bisa berubah	38
Gambar 4.14 Perbandingan pergerakan musuh dengan nilai diagonal sama dengan nilai vertikal (gambar kiri) dan nilai diagonal menggunakan nilai hitung jarak (kanan)	40
Gambar 4.15 Perbandingan pergerakan musuh dengan nilai diagonal sama dengan dua kali nilai vertikal (gambar kiri) dan nilai diagonal menggunakan nilai hitung jarak (kanan), hasil dari pergerakan sama	41
Gambar 4.16 Perbandingan pergerakan musuh dengan nilai diagonal kurang dari nilai vertikal (gambar kiri) dan nilai diagonal lebih dari dua kali nilai vertikal (kanan) hasilnya tidak efektif.	41
Gambar 4.17 Potongan method BiayaSebenarnya()	42
Gambar 4.18 Potongan method CariAdjacentNode() dengan A*	43
Gambar 4.19 Potongan method CariAdjacentNodeBiasa()	43
Gambar 4.20 Perbandingan Perbandingan Jalur A* dan <i>Greedy</i> (I)	44
Gambar 4.21 Perbandingan Perbandingan Jalur A* dan <i>Greedy</i> (II)	44

Gambar 4.22 Perbandingan Perbandingan Jalur A* dan <i>Greedy</i> (III)	45
Gambar 4.23 Pengukuran waktu tempuh algoritma A* (I)	46
Gambar 4.24 Pengukuran waktu tempuh algoritma <i>greedy</i> (I)	46
Gambar 4.25 Pengukuran waktu tempuh algoritma A* (II)	47
Gambar 4.26 Pengukuran waktu tempuh algoritma <i>greedy</i> (II)	47
Gambar 4.27 Pengukuran waktu tempuh algoritma A* (III)	48
Gambar 4.28 Pengukuran waktu tempuh algoritma <i>greedy</i> (III)	48
Gambar 4.29 Potongan <i>code</i> setter getter LokasiGrid dalam kelas PathNode.cs ..	50
Gambar 4.30 Potongan <i>code constructor</i> PathNode dalam kelas PathNode.cs	50

©UKYDIN

DAFTAR LAMPIRAN

LAMPIRAN A LISTING PROGRAM	A-1
A.1 Class Program.cs	A-1
A.2 Class Game1.cs	A-1
A.3 Class Kamera.cs	A-6
A.4 Class Sprite.cs	A-7
A.5 Class TileMap.cs	A-12
A.6 Class Player.cs.....	A-15
A.7 Class Partikel.cs	A-20
A.8 Class EfekManager.cs.....	A-21
A.9 Class SenjataManager.cs.....	A-24
A.10 Class Musuh.cs.....	A-27
A.11 Class PengaturanMusuh.cs.....	A-29
A.12 Class Markas.cs.....	A-30
A.13 Class MissionManager.cs.....	A-32
A.14 Class PathNode.cs	A-35
A.15 Class PathFinder.cs	A-36
A.16 Class PengaturanGame.cs	A-43

BAB I

PENDAHULUAN

1.1 Latar Belakang Masalah

Dewasa ini, di mana teknologi berkembang dengan pesat, membuat teknologi *game* juga berkembang dengan pesat. Perkembangan teknologi dalam dunia *game* telah meliputi banyak *platform*, mulai dari *game console*, *game PC*, dan *game mobile*. Teknik dan algoritma untuk mengembangkan *game* juga semakin berkembang. Perkembangan teknik dan algoritma itu meliputi bidang animasi, *game design*, *game engine*, *game graphics*, dan kecerdasan buatan (AI).

Salah satu algoritma dalam kecerdasan buatan adalah algoritma pencarian A* (A star). Algoritma ini membuat *computer-player* menjadi pintar karena bisa tahu jarak terdekat dalam mengejar pemain. Algoritma ini dapat di implementasikan menggunakan bahasa pemrograman dan *framework* apapun yang mendukung pembuatan *struct* atau *linked list*.

XNA adalah *framework / game engine* yang dibuat dan dikembangkan oleh Microsoft. *Framework* ini bisa dipakai untuk membuat *game* di PC, Xbox 360, dan Zune. *Framework* ini masih tergolong teknologi baru sehingga masih sedikit *game developer* yang memanfaatkan *framework* ini dalam membuat *game*. Maka dari itu, penulis akan mencoba mengimplementasikan algoritma A* di dalam *game tank 2D tile based* dengan menggunakan *framework XNA*.

1.2 Rumusan Masalah

Berdasarkan latar belakang masalah yang telah diuraikan, rumusan masalah dalam kasus ini adalah :

Apakah penerapan algoritma A* pada *game* tank-rampage dapat membuat *computer-player* menjadi lebih cerdas dan bisa menemukan jalur terpendek dengan tujuan yang selalu berubah-ubah?

1.3 Batasan Masalah

Penelitian ini dibatasi oleh parameter-parameter yang diuraikan sebagai berikut :

- a. Game 2 dimensi dengan animasi menggunakan *sprite image*.
- b. Game dijalankan pada komputer yang memiliki *environment .NET framework* dan *XNA framework*.
- c. Algoritma A* diterapkan pada *computer-player*.
- d. *Map* berupa kotak-kotak (*tile based*) dengan ukuran *pixel* yang ditentukan oleh programmer.

1.4 Hipotesis

Hipotesis dari penelitian ini adalah :

Implementasi algoritma A* pada *game tank-rampage* berhasil membuat *computer-player* menjadi lebih cerdas, yaitu dapat menghindari penghalang dan dapat mengetahui jarak terpendek dari posisinya ke posisi tank *player* walaupun posisi tank *player* selalu berubah-ubah.

1.5 Tujuan Penelitian

Penelitian ini memiliki tujuan untuk membuat *game* 2 dimensi *tile based* dengan *computer-player* yang cerdas dengan implementasi algoritma A*.

1.6 Metode

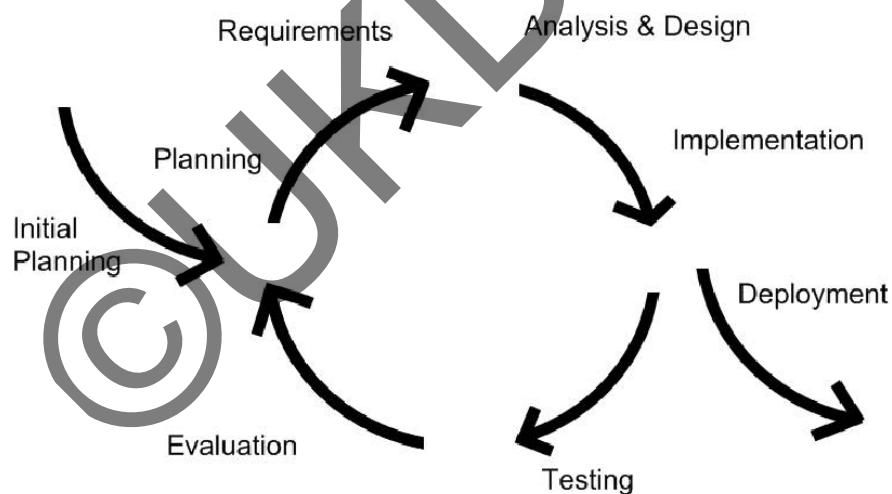
Metode Penelitian yang penulis gunakan dalam penelitian kali ini antara lain :

a. Metode Pengumpulan Data

- Penulis melakukan studi pustaka yang dilakukan dengan cara mempelajari teori-teori dan literatur-literatur yang mendukung penyelesaian penelitian ini, terutama yang berhubungan dengan algoritma A^* dan XNA Game Engine.
- Penulis melakukan observasi dengan mengamati dan memainkan game-game yang sejenis, yaitu game *tile-based 2D* yang menggunakan AI juga pada musuh, misalnya game Pacman, Mummy Maze, dan lain-lain.

b. Metode Pengembangan Sistem

Penulis menggunakan metode *Iterative Software Development*



Gambar 1.1 Iterative Software Life Cycle

Dikutip dari : Umair, Khan (2008). *Improved Iterative Software Development Method for Game Design*

- Fase Initial Planning adalah fase awal untuk mengonsep game yang akan dibuat.
- Fase Planning adalah merencanakan biaya proyek, dan deadline proyek.

- Fase Requirements adalah untuk menganalisa kebutuhan user untuk game, pada fase ini harus diperoleh informasi yang tepat agar game yang dibuat bisa benar-benar menghibur *user*.
- Fase Analisis dan Desain adalah menganalisis spesifikasi dari game yang sesuai dengan hasil dari proses requirement yaitu *game-play*, *game-rules*, dan *game-world*.
- Fase Implementasi adalah fase coding.
- Fase Deployment adalah fase saat game sudah dirilis untuk di tes.
- Fase Testing adalah untuk mencari bug yang ada pada game.
- Fase Evaluasi adalah fase untuk memperbaiki bug yang ada.

1.7 Sistematika Penulisan

Sistematika penulisan laporan tugas akhir ini dapat dijabarkan antara lain, Bab I tentang pendahuluan, berisi antara lain latar belakang masalah, perumusan masalah, batasan masalah, hipotesis, tujuan penelitian, metode atau pendekatan, serta sistematika penulisan.

Bab II tentang tinjauan pustaka, berisi tentang tinjauan pustaka yang menguraikan berbagai teori yang didapatkan dari sumber pustaka seperti jurnal ilmiah serta berisi landasan teori yang menguraikan berbagai teori pendukung penelitian.

Bab III tentang analisis dan perancangan sistem, berisi perancangan proses, perancangan basis data, perancangan antar muka, diagram alir sistem, penjelasan tentang algoritma serta implementasinya.

Bab IV tentang implementasi dari analisis dan perancangan sistem, berisi hasil dari penerapan perancangan basis data, perancangan antar muka, diagram alir sistem, penjelasan tentang algoritma serta implementasinya.

Dan yang terakhir adalah Bab V tentang kesimpulan dan saran, berisikan kesimpulan dan saran dari keseluruhan penelitian tugas akhir yang telah dilakukan.

BAB V

KESIMPULAN DAN SARAN

5.1 Kesimpulan

Kesimpulan yang dapat diambil dari penelitian ini adalah :

- a. Algoritma A* bisa membuat *computer-player* lebih cerdas jika dibandingkan dengan algoritma lain yang tidak memiliki nilai heuristik. Hal ini dibuktikan dengan penelitian dalam permainan *Tank Rampage* yaitu pergerakan musuh dengan algoritma A* yang bisa mengejar *tank player* dengan langsung bisa menghindari penghalang meskipun posisi *tank player* sebagai *goal* tidak selalu diam.
- b. XNA *framework* membantu *programmer* dalam menghitung biaya sebenarnya dengan menggunakan *function* dan variabel-variabel yang sudah ada dalam *library*-nya.

5.2 Saran

Saran untuk pengembangan penelitian ini adalah:

- a. Algoritma untuk pergerakan musuh bisa dikembangkan dengan algoritma *pathfinding* lain yang lebih baru, misalnya dengan menggunakan algoritma variasi A* (*Beam Search*, *IDA**, *dynamic weighting*, *bandwidth search*, *bidirectional search*, *dynamic A** dan *Lifelong Planning A**)
- b. Metode deteksi tumbukan antara satu objek musuh dengan objek musuh yang lain bisa diperbaiki agar posisi musuh tidak ada yang saling bertumpuk.
- c. Aturan dalam *game* bisa ditambah lagi, misalnya tiap musuh bisa menembak dan *player* bisa meng-*upgrade* senjatanya.
- d. *Game* bisa dikembangkan tidak hanya dalam *platform* PC, tetapi juga *Windows Phone*, *Xbox*, atau *Zune* karena XNA *Framework* mendukung pembuatan *game* pada platform tersebut.

DAFTAR PUSTAKA

- Jaegers, Kurt. (2010). *XNA 4.0 Game Development by Example*. Birmingham : PACKT Publishing. Retrieved 30 Desember 2011 from http://www.ebook3000.com/Programming/General/XNA-4-0-Game-Development-by-Example--Beginner-s-Guide_101583.html
- Miller, Tom dan Johnson. (2010). *XNA Game Studio 4.0 Programming : Developing for Windows Phone 7 and Xbox 360*. Boston : Pearson Education. Retrieved 31 Desember 2011 from <http://www.filestube.com/cvrlitcQ9O86izkLseQnoJ/Pearson-XNA-Game-Studio-4-0-Programming-2010-RETAiL-EBook-DiGiBook.html>
- Aprillia, Shieny dan Widhiyasa. (n/a). *Membuat Game dengan XNA Game Studio*. Bandung : Agate Studio. Retrived 4 Januari 2012 from <http://netindonesia.net/files/folders/ebook/entry186774.aspx>
- Ardhinata, Taufan Harsillo. (n/a). *Step by Step XNA Game Studio 4.0 : "Make Your Own Game"*. Retrived 4 Januari 2012 from <http://students.netindonesia.net/blogs/taufan/archive/2011/03/16/tutorial-e-book-xna-game-studio-4-0-make-your-own-game.aspx>
- Silva, Pedro Brandão dan Coelho, António. (2011). *Procedural Modeling for Realistic Virtual Worlds Development*. Journal of Virtual World Reserch Volume 4, Number 1, Metaverse Assembled 2.0 July 2011.
- Donnelly, Ryan. (2006). *Artificial Intelligence in Gaming*. Department of Computer Science University of Wisconsin Platteville
- Rahayu, Dewi Nurul. (2010). *Penerapan Algoritma A* (A Star) Dalam Pencarian Jalan Terpendek Pada Game Pathfinding*. Jurusan Teknik Informatika Fakultas Teknik Dan Ilmu Komputer Universitas Komputer Indonesia Bandung.
- Xiao Cui and Hao Shi. (2011). *A*-based Pathfinding in Modern Computer Games*. IJCSNS International Journal of Computer Science and Network Security, VOL.11 No.1, January 2011
- Sood, Abhishek. (2012). *Neevengine: Using Xna Game Studio For Serious Game Design And Development*. San Diego State University.
- Edoror, Michael. (2012). *Microsoft .NET and XNA Framework for the Robot Operating System*. University of Maryland, Baltimore County