
**IMPLEMENTASI KECERDASAN BUATAN UNTUK 3 TIPE
MONSTERS PADA PERMAINAN TOWER DEFENSE**

Skripsi



oleh
STEFANUS SETYA BUDI
71110035

PROGRAM STUDI TEKNIK INFORMATIKA FAKULTAS TEKNOLOGI INFORMASI
UNIVERSITAS KRISTEN DUTA WACANA
2015

**IMPLEMENTASI KECERDASAN BUATAN UNTUK 3 TIPE
MONSTERS PADA PERMAINAN TOWER DEFENSE**

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

STEFANUS SETYA BUDI
71110035

PROGRAM STUDI TEKNIK INFORMATIKA FAKULTAS TEKNOLOGI INFORMASI
UNIVERSITAS KRISTEN DUTA WACANA
2015

PERNYATAAN KEASLIAN SKRIPSI

Saya menyatakan dengan sesungguhnya bahwa skripsi dengan judul:

IMPLEMENTASI KECERDASAN BUATAN UNTUK 3 TIPE MONSTERS PADA PERMAINAN TOWER DEFENSE

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, 17 Juni 2015



STEFANUS SETYA BUDI
71110035

HALAMAN PERSETUJUAN

Judul Skripsi : IMPLEMENTASI KECERDASAN BUATAN
UNTUK 3 TIPE MONSTERS PADA PERMAINAN
TOWER DEFENSE

Nama Mahasiswa : STEFANUS SETYA BUDI

NIM : 71110035

Matakuliah : Skripsi (Tugas Akhir)

Kode : TIW276

Semester : Genap

Tahun Akademik : 2014/2015

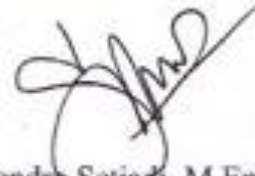
Telah diperiksa dan disetujui di
Yogyakarta,
Pada tanggal 17 Juni 2015

Dosen Pembimbing I



Gloria Virginia, S.Kom., MAI, Ph.D.

Dosen Pembimbing II



Hendro Setiadi, M.Eng

HALAMAN PENGESAHAN

IMPLEMENTASI KECERDASAN BUATAN UNTUK 3 TIPE MONSTERS PADA PERMAINAN TOWER DEFENSE

Oleh: STEFANUS SETYA BUDI / 71110035

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 11 Juni 2015

Yogyakarta, 18 Juni 2015

Mengesahkan,

Dewan Penguji:

1. Gloria Virginia, S.Kom., MAI, Ph.D.
2. Hendro Setiadi, M.Eng
3. Ignatia Dhian E K R, S.Kom, M.Eng
4. Rosa Delima, S.Kom., M.Kom.



DUTA WACANA

Dekan

Ketua Program Studi



(Budi Susanto, S.Kom., M.T.)



(Gloria Virginia, Ph.D.)

KATA PENGANTAR

Puji Syukur kepada Tuhan Yang Maha Esa atas segala berkat dan rahmat yang telah diberikan kepada penulis sehingga mampu menyelesaikan skripsi dengan baik dan benar. Skripsi yang dibuat penulis berjudul Implementasi Kecerdasan Buatan Untuk 3 Tipe *Monsters* Pada Permainan *Tower Defense*. Penulisan skripsi ini bertujuan sebagai pemenuhan dari salah satu syarat dalam memperoleh gelar Sarjana Strata Satu Program Studi Teknik Informatika Universitas Kristen Duta Wacana. Selain itu juga bertujuan sebagai sarana pembelajaran bagi penulis maupun pembaca.

Dalam menyelesaikan skripsi ini tidak semata – mata terselesaikan oleh kerja penulis sendiri, namun banyak pihak telah terlibat dalam membantu dan mendukung penulis dalam melewati setiap proses pembuatan. Skripsi hingga terselesaikan dengan baik. Oleh karena itu, pada kesempatan ini penulis ingin mengucapkan terimakasih kepada beberapa pihak :

1. Gloria Virginia, S.Kom., MAI, Ph.D. selaku Dosen Pembimbing I yang dengan sabar membimbing dan memberikan masukan-masukan yang membangun, sehingga skripsi dapat terselesaikan dengan baik dan tepat waktu.
2. Hendro Setiadi, M.Eng selaku Dosen Pembimbing II yang dengan sabar membimbing penulisan laporan skripsi, sehingga laporan skripsi dapat selesai dengan baik.
3. Ayah, Ibu, kakak dan keluarga penulis yang selalu memberikan dukungan doa, semangat, penuh kasih sayang kepada penulis dan menjadikan motivasi utama penulis untuk terus berusaha menyelesaikan skripsi.
4. Johannes Nindyo dan Dhanny Setiawan selaku rekan satu tim yang membantu dalam pembuatan *sprite* dan antarmuka dalam permainan *Tower Defense*.
5. Ega Aquilla sebagai sahabat yang terus memberikan motivasi sehingga skripsi dapat berjalan dengan lancar.
6. Semua pihak yang tidak dapat disebutkan satu persatu yang telah membantu penulis secara langsung atau tidak langsung.

Penulis menyadari bahwa Skripsi yang telah dibuat belum sempurna dan masih memiliki banyak kekurangan yang terjadi akibat keterbatasan penulis. Oleh karena itu, penulis terbuka terhadap setiap kritik dan saran yang membangun mengenai laporan Skripsi ini. Dengan demikian, penulis dapat memberikan karya yang lebih baik dan berguna bagi pembaca di masa mendatang.

Penulis memohon maaf apabila terdapat kesalahan selama proses penyusunan laporan dan pembuatan permainan. Penulis berharap pengetahuan yang ada dalam skripsi ini dapat bermanfaat bagi banyak pihak dan pembaca. Akhir kata, semoga karya ini dapat berguna bagi setiap pembaca maupun pihak lain.

Yogyakarta, 25 Mei 2015



Stefanus Setya Budi

INTISARI

IMPLEMENTASI KECERDASAN BUATAN UNTUK 3 TIPE *MONSTERS* PADA PERMAINAN *TOWER DEFENSE*

Algoritma A^* banyak dipergunakan oleh orang untuk menyelesaikan masalah pencarian jalur tercepat. Salah satu implementasi algoritma A^* adalah untuk pencarian jalur tercepat pada objek dalam *mobile game*. *Mobile game* yang terus berkembang dengan pesat membuat penulis memiliki sebuah ide untuk membuat sebuah *game tower defense* yang memadukan genre *action*, *puzzle*, dan *side-scrolling platformer*, yang didalamnya terdapat 3 macam *monsters* yang memiliki tingkat kecerdasan buatan yang berbeda. *Game*

Penulis menggunakan algoritma A^* untuk 3 tipe *monsters* yaitu tipe normal, tipe *fly*, dan tipe *boss*. *Monsters* tipe normal merupakan *monsters* dengan kecerdasan paling rendah, sedangkan *monsters* tipe *fly* dan *boss* memiliki kecerdasan yang lebih, karena 2 tipe *monsters* ini dapat mengubah *goal state* / targetnya sesuai dengan kondisi permainan, sehingga tingkat kesulitan dari permainan bertambah.

Dalam penelitian ini, implementasi algoritma A^* dapat diterapkan dengan baik untuk memberikan kecerdasan pada 3 tipe *monsters* pada permainan *Tower Defense*. Selain itu, permainan ini memberikan latihan soal matematika yaitu penjumlahan, pengurangan, dan perkalian untuk angka 0 sampai 10. Permainan ini juga bermanfaat untuk melatih berpikir cepat, karena pemain diberikan batasan waktu untuk menjawab setiap pertanyaan yang muncul.

Kata kunci : Kecerdasan Buatan, *Games*, A^*

DAFTAR ISI

PERNYATAAN KEASLIAN SKRIPSI.....	iii
HALAMAN PERSETUJUAN.....	iv
HALAMAN PENGESAHAN.....	v
KATA PENGANTAR	vi
INTISARI.....	viii
DAFTAR ISI.....	ix
DAFTAR GAMBAR	xi
DAFTAR TABEL.....	xiv
DAFTAR LAMPIRAN.....	xv
BAB 1 PENDAHULUAN	1
1.1 Latar Belakang Masalah.....	1
1.2 Rumusan Masalah	2
1.3 Batasan Masalah.....	2
1.4 Tujuan Penelitian.....	3
1.5 Metode Penelitian.....	3
1.6 Sistematika Penulisan.....	5
BAB 2 LANDASAN TEORI.....	7
2.1 Tinjauan Pustaka	7
2.1.1 Algoritma A* Untuk Menyelesaikan 8-Puzzle	7
2.1.2 Algoritma A* Untuk Permainan Komputer Modern	8
2.1.3 Efisiensi Algoritma <i>Pathfinding</i> Pada <i>Grid 2D</i>	9
2.1.4 Macam – Macam Genre <i>Game</i>	10
2.2 Landasan Teori	10
2.2.1 Algoritma A*.....	11
BAB 3 ANALISIS DAN PERANCANGAN SISTEM.....	17

3.1	Peralatan yang Digunakan Dalam Riset.....	17
3.2	Perancangan <i>Game</i>	17
	3.2. 1 Perancangan <i>Sprite</i>	17
	3.2.2 Perancangan Antarmuka	20
	3.2.3 Perancangan <i>Gameplay</i>	21
	3.2.4 Perancangan Algoritma <i>A*</i>	23
BAB 4 IMPLEMENTASI DAN ANALISIS SISTEM		30
4.1	Implementasi <i>Sprite</i> dan Antarmuka.....	30
	4.1.1 Implementasi <i>Sprite</i>	30
	4.1.2 Implementasi Antarmuka.....	32
4.2	Implementasi Algoritma <i>A*</i>	37
	4.2.1 Implementasi <i>A*</i> Untuk <i>Monsters</i> Tipe Normal	39
	4.2.2 Implementasi <i>A*</i> Untuk <i>Monsters</i> Tipe <i>Fly</i>	46
	4.2.2 Implementasi <i>A*</i> Untuk <i>Monsters</i> Tipe <i>Boss</i>	53
4.3	Uji Coba dan Analisis.....	59
BAB 5 KESIMPULAN DAN SARAN.....		64
5.1	Kesimpulan.....	64
5.2	Saran.....	66
DAFTAR PUSAKA.....		67
LAMPIRAN.....		68

DAFTAR GAMBAR

Gambar 1.1 Kuesioner Untuk Megecek Kecerdasan Buatan Pada Monsters.	5
Gambar 2.1 Contoh Implementasi A^* dan <i>Mahattan Distance</i> Pada <i>8-Puzzle</i>	7
Gambar 2.2 <i>Graph</i> yang Menggambarkan Cara Kerja Algoritma A^*	9
Gambar 2.3 Contoh Penerapan Algoritma A^* Untuk Menggerakkan Monsters Tipe Normal (<i>State Awal</i>).....	10
Gambar 2.4 Contoh Penerapan Algoritma A^* Untuk Menggerakkan Monsters Tipe Normal (<i>Langkah Pertama</i>)	11
Gambar 2.5 Contoh Penerapan Algoritma A^* Untuk Menggerakkan Monsters Tipe Normal (<i>Langkah Kedua</i>).....	10
Gambar 2.6 Contoh Penerapan Algoritma A^* Untuk Menggerakkan Monsters Tipe Normal (<i>Langkah Ketiga</i>).....	11
Gambar 2.7 Contoh Penerapan Algoritma A^* Untuk Menggerakkan Monsters Tipe <i>Fly</i> Tipe <i>Boss</i> (<i>State Awal</i>).....	12
Gambar 2.8 Contoh Penerapan Algoritma A^* Untuk Menggerakkan Monsters Tipe <i>Fly</i> atau Tipe <i>Boss</i> (<i>Langkah Pertama</i>)	13
Gambar 2.9 Contoh Penerapan Algoritma A^* Untuk Menggerakkan Monsters Tipe <i>Fly</i> atau Tipe <i>Boss</i> (<i>Langkah Kedua</i>)	14
Gambar 3.1 Hero <i>Sprite</i>	17
Gambar 3.2 <i>Monsters Sprite</i> (Tipe Normal)	18
Gambar 3.3 <i>Monsters Sprite</i> (Tipe <i>Fly</i>)	18
Gambar 3.4 <i>Monsters Sprite</i> (Tipe <i>Boss</i>).....	19
Gambar 3.5 <i>Tower Sprite</i>	19

Gambar 3.6 Kumpulan <i>Sprite</i> Antarmuka Untuk Permainan <i>Tower Defense</i> (Bagian 1).....	20
Gambar 3.7 Kumpulan <i>Sprite</i> Antarmuka Untuk Permainan <i>Tower Defense</i> (Bagian 2).....	21
Gambar 3.8 <i>Flowchart</i> yang Menggambarkan Alur Permainan <i>Tower Defense</i>	22
Gambar 3.9 <i>Flowchart</i> yang Menggambarkan Cara Kerja Algoritma A* Untuk Mengatur <i>Behavior Monsters</i> Tipe Normal (<i>Goal Node = Tower</i>).....	24
Gambar 3.10 <i>Flowchart</i> yang Menggambarkan Cara Kerja Algoritma A* Untuk Mengatur <i>Behavior Monsters</i> Tipe <i>Fly</i> (<i>Goal Node = Tower / Hero</i>)..	25
Gambar 3.11 <i>Flowchart</i> yang Menggambarkan Cara Kerja Algoritma A* Untuk Mengatur <i>Behavior Monsters</i> Tipe <i>Boss</i> (<i>Goal Node = Tower / Hero / Bom</i>).....	26
Gambar 4.1 Hasil Implementasi <i>Hero Sprite</i>	30
Gambar 4.2 Hasil Implementasi <i>Monsters Sprite</i> (Tipe Normal <i>Melee</i>)	30
Gambar 4.3 Hasil Implementasi <i>Monsters Sprite</i> (Tipe Normal <i>Ranger</i>)	31
Gambar 4.4 Hasil Implementasi <i>Monsters Sprite</i> (Tipe <i>Fly</i>).....	31
Gambar 4.5 Hasil Implementasi <i>Monsters Sprite</i> (Tipe <i>Boss</i>).....	31
Gambar 4.6 Hasil Implementasi <i>Tower Sprite</i>	32
Gambar 4.7 Hasil Implementasi Antarmuka <i>Splash Screen</i>	32
Gambar 4.8 Hasil Implementasi Antarmuka <i>Loading Screen</i>	33
Gambar 4.9 Hasil Implementasi Antarmuka Menu Utama (<i>Panel Upgrade</i>)	33
Gambar 4.10 Hasil Implementasi Antarmuka Menu Utama (<i>Panel Achievement</i>)	34
Gambar 4.11 Hasil Implementasi Antarmuka Menu Utama (<i>Panel Helps</i>)	35
Gambar 4.12 Hasil Implementasi Antarmuka Menu Utama (<i>Panel Power Up</i>).....	35
Gambar 4.13 Hasil Implementasi Antarmuka <i>In Game</i>	36

Gambar 4.14 Hasil Implementasi Antarmuka <i>Game Over</i>	37
Gambar 4.15 Objek yang Berisi <i>Script</i> Pathfinder A*	38
Gambar 4.16 Contoh Implementasi dari <i>Script</i> Pathfinder A* dengan Nilai Variabel Pada Gambar 4.15	39
Gambar 4.17 Contoh Kasus <i>Pathfinding</i> Untuk <i>Monsters</i> Tipe Normal (Bagian 1) ..	40
Gambar 4.18 Contoh Kasus <i>Pathfinding</i> Untuk <i>Monsters</i> Tipe Normal (Bagian 2) ..	42
Gambar 4.19 Contoh Kasus <i>Pathfinding</i> Untuk <i>Monsters</i> Tipe Normal (Bagian 3) ..	43
Gambar 4.20 Contoh Kasus <i>Pathfinding</i> Untuk <i>Monsters</i> Tipe Normal (Bagian 4) ..	44
Gambar 4.21 Contoh Kasus <i>Pathfinding</i> Untuk <i>Monsters</i> Tipe Normal (Bagian 5) ..	44
Gambar 4.22 Contoh Kasus <i>Pathfinding</i> Untuk <i>Monsters</i> Tipe Normal (Bagian 6) ..	45
Gambar 4.23 Contoh Kasus <i>Pathfinding</i> Untuk <i>Monsters</i> Tipe <i>Fly</i> (Bagian 1)	46
Gambar 4.24 Contoh Kasus <i>Pathfinding</i> Untuk <i>Monsters</i> Tipe <i>Fly</i> (Bagian 2)	49
Gambar 4.25 Contoh Kasus <i>Pathfinding</i> Untuk <i>Monsters</i> Tipe <i>Fly</i> (Bagian 3)	49
Gambar 4.26 Contoh Kasus <i>Pathfinding</i> Untuk <i>Monsters</i> Tipe <i>Fly</i> (Bagian 4)	50
Gambar 4.27 Contoh Kasus <i>Pathfinding</i> Untuk <i>Monsters</i> Tipe <i>Fly</i> (Bagian 5)	52
Gambar 4.28 Contoh Kasus <i>Pathfinding</i> Untuk <i>Monsters</i> Tipe <i>Fly</i> (Bagian 6)	52
Gambar 4.29 Contoh Kasus <i>Pathfinding</i> Untuk <i>Monsters</i> Tipe <i>Boss</i> (Bagian 1)	53
Gambar 4.30 Contoh Kasus <i>Pathfinding</i> Untuk <i>Monsters</i> Tipe <i>Boss</i> (Bagian 2)	56
Gambar 4.31 Contoh Kasus <i>Pathfinding</i> Untuk <i>Monsters</i> Tipe <i>Boss</i> (Bagian 3)	57
Gambar 4.32 Contoh Kasus <i>Pathfinding</i> Untuk <i>Monsters</i> Tipe <i>Boss</i> (Bagian 4)	57
Gambar 4.33 Contoh Kasus <i>Pathfinding</i> Untuk <i>Monsters</i> Tipe <i>Boss</i> (Bagian 5)	58
Gambar 4.34 Contoh Kasus <i>Pathfinding</i> Untuk <i>Monsters</i> Tipe <i>Boss</i> (Bagian 6)	58
Gambar 4.35 Lembar wawancara Untuk Uji Coba Permainan <i>Tower Defense</i>	61
Gambar 4.36 Data Hasil Uji Coba Untuk Membandingkan Tingkat Kesulitan 3 Tipe Monsters	62

DAFTAR TABEL

Tabel 4.1 Keterangan Variabel Dalam <i>Script</i> Pathfinder 2D.....	38
Tabel 4.2 Nilai x dan y dari <i>Node</i> Disekitar Posisi Awal <i>Monsters</i> Tipe Normal.....	40
Tabel 4.3 Nilai f_cost dari <i>Node</i> Disekitar Posisi Awal <i>Monsters</i> Tipe Normal....	41
Tabel 4.4 Nilai x dan y dari <i>Node</i> Disekitar Posisi Awal <i>Monsters</i> Tipe <i>Fly</i>	47
Tabel 4.5 Nilai f_cost dari <i>Node</i> Disekitar Posisi Awal <i>Monsters</i> Tipe <i>Fly</i>	48
Tabel 4.6 Nilai x dan y dari <i>Node</i> Disekitar Posisi Awal <i>Monsters</i> Tipe <i>Boss</i>	54
Tabel 4.7 Nilai f_cost dari <i>Node</i> Disekitar Posisi Awal <i>Monsters</i> Tipe <i>Boss</i>	55

DAFTAR LAMPIRAN

<i>Node Script</i>	A-1
<i>Pathfinder 2D Script</i>	A-1
<i>Pathfinding 2D Script</i>	A-11
<i>Normal Melee Script</i>	A-12
<i>Normal Ranger Script</i>	A-15
<i>Fly AI Script</i>	A-18
<i>Boss AI Script</i>	A-22
Lembar Wawancara Untuk Uji Coba Permainan Tower Defense	B-1

©UKYDIN

INTISARI

IMPLEMENTASI KECERDASAN BUATAN UNTUK 3 TIPE *MONSTERS* PADA PERMAINAN *TOWER DEFENSE*

Algoritma A^* banyak dipergunakan oleh orang untuk menyelesaikan masalah pencarian jalur tercepat. Salah satu implementasi algoritma A^* adalah untuk pencarian jalur tercepat pada objek dalam *mobile game*. *Mobile game* yang terus berkembang dengan pesat membuat penulis memiliki sebuah ide untuk membuat sebuah *game tower defense* yang memadukan genre *action*, *puzzle*, dan *side-scrolling platformer*, yang didalamnya terdapat 3 macam *monsters* yang memiliki tingkat kecerdasan buatan yang berbeda. *Game*

Penulis menggunakan algoritma A^* untuk 3 tipe *monsters* yaitu tipe normal, tipe *fly*, dan tipe *boss*. *Monsters* tipe normal merupakan *monsters* dengan kecerdasan paling rendah, sedangkan *monsters* tipe *fly* dan *boss* memiliki kecerdasan yang lebih, karena 2 tipe *monsters* ini dapat mengubah *goal state* / targetnya sesuai dengan kondisi permainan, sehingga tingkat kesulitan dari permainan bertambah.

Dalam penelitian ini, implementasi algoritma A^* dapat diterapkan dengan baik untuk memberikan kecerdasan pada 3 tipe *monsters* pada permainan *Tower Defense*. Selain itu, permainan ini memberikan latihan soal matematika yaitu penjumlahan, pengurangan, dan perkalian untuk angka 0 sampai 10. Permainan ini juga bermanfaat untuk melatih berpikir cepat, karena pemain diberikan batasan waktu untuk menjawab setiap pertanyaan yang muncul.

Kata kunci : Kecerdasan Buatan, *Games*, A^*

BAB 1

PENDAHULUAN

1. 1 Latar Belakang

Mobile game dapat diartikan sebagai sebuah permainan yang dimainkan di alat-alat elektronik seperti *smartphone*, *tablet*, *PDA*, dll, (Ola Davidsson. 2010). Sekarang ini, *mobile game* telah berkembang sangat pesat. Sejak tahun 2007 hingga tahun 2012, pasar *mobile game* telah tumbuh hampir 33%, dan menghasilkan laba diatas 9 milyar dollar (Nouch, 2013). Di tahun 2011, sebuah *game* berjudul *Angry Bird*, berhasil menjadi aplikasi perangkat lunak pertama yang dapat mencapai 1 milyar *download* (Heriksen, 2012).

Terdapat berbagai macam *mobile game* dengan genre yang berbeda-beda dapat diunduh dari *Google Playstore*. Sebagai contoh : *Zombie Diary*. (*game* dengan genre *action*), *Candy Crush* (*game* dengan genre *puzzle*), dsb. *Game Tower Defense* adalah sebuah *game* yang memadukan 3 macam genre yaitu : *action*, *puzzle*, dan *side-scrolling platformer*. Dalam *game* ini, pemain diminta untuk memainkan sebuah **Hero** yang bertugas melindungi *tower* dari serangan *monsters* jahat. Selain *monsters*, ada bom yang muncul secara acak di suatu tempat. Setiap bom memiliki pertanyaan matematika yang harus dijawab oleh **Hero**. **Hero** diminta untuk membunuh *monsters* yang bermunculan dari samping dan atas, untuk menemukan jawaban dari pertanyaan yang ada di bom. Jika pemain berhasil menjawab pertanyaan yang ada di bom, maka skor akan bertambah, namun jika pemain gagal menjawab pertanyaan dalam waktu tertentu, maka bom akan meledak dan skor

berkurang drastis. *Game* ini berakhir jika nyawa dari **Hero** atau nyawa *tower* habis.

Monsters dalam *game Tower Defense* yang dikembangkan adalah karakter yang bergerak secara otomatis dan bertujuan untuk menyerang **Hero/tower**. Saat ini pergerakannya hanya menuju satu titik saja, yaitu ke arah *tower*. Untuk meningkatkan tingkat kesulitan dari permainan *Tower Defense*, maka *monsters* perlu diberi kecerdasan, sehingga *monsters* tidak hanya bergerak pada satu titik saja, namun bisa berpatroli menyerang **Hero, tower**, dan bom. Hal ini dapat dilakukan dengan menggunakan pendekatan kecerdasan buatan. Salah satu pendekatan itu adalah dengan menggunakan algoritma A^* . Penelitian ini bertujuan untuk mengimplementasikan algoritma kecerdasan buatan, yaitu algoritma A^* , untuk memberikan kecerdasan pada 3 tipe *monsters* di permainan *Tower Defense*.

1.2 Rumusan Masalah

- a. Bagaimana cara membuat 3 tipe *monsters* di *game Tower Defense* agar ketiganya memiliki kecerdasan yang berbeda?
- b. Bagaimana respon *gamers* terhadap implementasi *AI* pada ketiga tipe *monsters* di *game Tower Defense*?

1.3 Batasan Masalah

- a. *Game* ini dikembangkan oleh *indie game developer* bernama *Myriad Dream* yang beranggotakan 4 orang. *Job Description* dari masing-masing anggota :
 - Stefanus Setya Budi : *AI Programming and Gameplay*
 - Johanes Nindyo W : *UI Programming and Animator*
 - Dhanny Setiawan : *Character and Environment Design*

Fokus dari penelitian ini adalah implementasi *AI* untuk 3 tipe *monsters*.

- b. Algoritma *A** akan diimplementasikan untuk 3 tipe *monster* di *game Tower Defense*. (tipe normal, tipe *fly*, tipe *boss*).
 - Tipe normal : target utama dari *monsters* adalah *tower*.
 - Tipe *fly* : target utama dari *monsters* adalah **Hero**.
 - Tipe *boss* : target dari *monsters* berubah-ubah sesuai dengan kondisi.
- c. *Game* ini dikembangkan untuk platform *mobile android*

1.4 Tujuan Penelitian

Tujuan utama dari penelitian ini adalah mengimplementasikan algoritma *A** untuk *memberikan* kecerdasan yang berbeda pada 3 tipe *monsters* dalam permainan *Tower Defense*. Kecerdasan yang berbeda yang dimaksud adalah *monsters* dapat mengetahui targetnya dan mampu bertindak sesuai dengan kondisi permainan.

1.5 Metode Penelitian

Tahap – tahap penelitian dalam pembuatan *game Tower Defense* adalah sebagai berikut:

1. Perancangan *Game*

Pada tahap ini dilakukan perancangan bagaimana alur cerita dan *gameplay* dari *game Tower Defense*.

2. Pengembangan *Game*

Pada tahap ini, *game* dikembangkan berdasarkan *resource* yang telah terkumpul. Tahap ini meliputi :

a. Pembuatan *Sprite*

Pada tahap ini dilakukan pembuatan untuk *character sprite*, *background sprite*, dan *foreground sprite*. Yang bertanggung jawab pada bagian ini adalah Dhanny S dan Hildaria Putri.

b. Perancangan Antarmuka

Pada tahap ini dilakukan perancangan/*prototyping* tampilan dari *game* secara keseluruhan. Tampilan meliputi : *Main Menu UI*, *Help Menu UI*, *Difficulty Selection UI*, *Gameplay UI*. Yang bertanggung jawab pada bagian ini adalah saya dan Johannes Nindyo W.

c. Pembuatan Animasi

Pada tahap ini, dilakukan pembuatan animasi untuk karakter, *background parallax*, dsb. Yang bertanggung jawab pada bagian ini adalah Johannes Nindyo W.

d. Pengkodean *UI*, *AI*, dan *Gameplay*

Pada tahap ini, dilakukan pengkodean untuk *UI*, *AI*, dan *gameplay*. Yang bertanggung jawab pada bagian ini adalah saya.

e. *Debugging*

Pada tahap ini, dilakukan pengecekan apakah ada kesalahan-kesalahan kode program yang membuat *game* tidak berjalan semestinya. Yang bertanggung jawab pada bagian ini adalah saya.

3. Evaluasi *Game*

a. Internal Validasi

Pada tahap ini dilakukan pengecekan dan validasi apa *game* yang dikembangkan sudah menerapkan algoritma A^* untuk membuat *monster* lebih cerdas.

b. Uji Coba

Pada tahap ini, dilakukan uji coba dengan *gamers* untuk mengetahui apakah kecerdasan buatan pada 3 tipe *monster* di *game Tower Defense* berjalan dengan baik. *Gamers* akan diminta memainkan *game* terlebih dahulu, kemudian akan dilakukan *informal interview* dengan *gamers*. Contoh dari *interview* ada pada Gambar 1.1.

No	Requirment	Result
1	Bertahan hidup lebih dari 30 detik	Ya
2	Bertahan hidup lebih dari 60 detik	Ya
3	Bertahan hidup lebih dari 90 detik	Tidak

- Tipe Monster Termudah : Tipe Normal
- Tipe Monster Tersulit : Tipe Boss

Gambar 1.1 Contoh *Interview* untuk megecek kecerdasan buatan pada *monsters*.

4. Analisis *Game*

Pada tahap ini, Dilakukan analisa dari hasil uji coba yang dilakukan dengan *gamers*. Pada tahap ini, dilakukan analisis apakah pengimplementasian algoritma A^* untuk *monsters* di *game Tower Defense* berhasil dengan baik.

1.6 Sistematika Penulisan

Bab 1 dari penelitian ini menjabarkan latar belakang dilakukannya penelitian. Pada bab 1, penulis memberikan penjelasan secara umum tentang bagaimana alur permainan dari permainan *Tower Defense*., dan mengapa *monsters* di dalam *game* ini perlu diberikan kecerdasan buatan. Penulis juga menjabarkan batasan masalah, tujuan penelitian, dan metode penelitian. Bab 2 menjabarkan tentang algoritma A^* secara umum dan bagaimana pengimplementasiannya di dalam permainan *Tower Defense*. Bab 3 adalah

tahap analisis terhadap algoritma A^* untuk dapat diimplementasikan ke dalam 3 tipe *monsters* yang ada di permainan *Tower Defense*. Pada tahap ini juga dilakukan perancangan algoritma, *flowchart*, dan perancangan arsitektur sistem.

Bab 4 mejabarkan hasil implementasi dari algoritma A^* yang dilakukan pada bab 3. Hasil implementasi disajikan dalam tabel dan gambar. Bab ini juga menganalisis hasil implementasi algoritma A^* pada 3 tipe *monsters* di permainan *Tower Defense*. Analisis dilakukan dengan membagikan Kuesioner untuk beberapa *gamers*. Contoh Kuesioner ada pada Gambar 1.1. Mereka akan diminta untuk mengisi Kuesioner tersebut kemudian penulis akan menanalisis hasilnya. Bab 5 menyimpulkan hasil dari analisis yang dilakukan pada bab 4. Bab 5 juga memuat saran untuk pengembangan sistem untuk ke depannya.

BAB 5

KESIMPULAN DAN SARAN

5.1 Kesimpulan

Berdasarkan hasil penelitian yang dilakukan untuk mengimplementasikan algoritma A* untuk 3 tipe *monsters* pada permainan *Tower Defense*, diperoleh beberapa hasil penelitian.

- a. Algoritma A* berhasil diimplementasikan untuk *monsters* tipe normal. *Monsters* tipe ini sudah dapat melakukan aksi utamanya, yaitu bergerak menuju *tower* dan menyerang *tower*.
- b. Algoritma A* berhasil diimplementasikan untuk *monsters* tipe *fly*. *Monsters* tipe ini sudah dapat melakukan aksi utamanya, yaitu bergerak menuju *tower* dan menyerang *tower*, serta dapat bergerak menuju **Hero** dan menyerang **Hero**. *Monsters* dapat mempertimbangkan targetnya dengan melihat skala prioritas berdasarkan *health poin*.
- c. Algoritma A* berhasil diimplementasikan untuk *monsters* tipe *boss*. *Monsters* tipe ini sudah dapat melakukan 3 aksi utamanya, yaitu 1) Bergerak menuju *tower* dan menyerang *tower*; 2) bergerak menuju **Hero** dan menyerang **Hero** jika diserang oleh **Hero**; dan 3) *Monsters* tipe *boss* juga sudah bisa bergerak menuju bom dan menyerangnya agar cepat meledak. *Monsters* dapat mempertimbangkan targetnya dengan melihat skala prioritas berdasarkan *health poin*.

Berdasarkan data hasil uji coba pada Gambar 4.36, 10 orang *gamers* (100%) mengatakan bahwa tipe *monsters* termudah adalah tipe normal, sedangkan 10 orang *gamers* (100%) juga sepakat bahwa tipe *monsters* tersulit adalah tipe *boss*.

Selain itu, pada 3 percobaan yang dilakukan, terlihat bahwa pada percobaan pertama, 10 orang berhasil dengan mudah mengalahkan *monsters* tipe normal, sedangkan tidak seorang pun berhasil mengalahkan *monsters* tipe *boss*. Pada percobaan kedua, 10 orang berhasil dengan mudah mengalahkan *monsters* tipe normal, dan hanya 2 orang berhasil mengalahkan *monsters* tipe *boss*. Pada percobaan terakhir dengan status maksimal, 7 orang berhasil mengalahkan *monsters* tipe *boss*, namun masih ada 3 orang yang gagal mengalahkannya. Dengan data ini, dapat ditarik kesimpulan bahwa kecerdasan buatan yang diimplementasikan pada 3 tipe *monsters* sudah berhasil dengan cukup baik, karena dapat dirasakan tingkat kesulitan yang berbeda dari ketiga tipe *monsters*.

Kecerdasan buatan pada *monsters* tipe normal, *fly*, dan *boss*, diterapkan berdasarkan 3 hal.

- a. *Goal state* yang berbeda dari setiap tipe *monsters*
 - *Monsters* tipe normal yang merupakan *monsters* terlemah, hanya memiliki 1 *goal state* saja, yaitu *tower*.
 - *Monsters* tipe *fly* memiliki 2 *goal state* yang dapat berubah-ubah setiap waktu yaitu *tower* atau **Hero**.
 - *Monsters* tipe *boss* yang merupakan *monsters* tercerdas, memiliki 3 *goal state* yaitu *tower*, bom, atau **Hero**.
- b. Perubahan *goal state* karena suatu kondisi
 - *Goal state* dari *Monsters* tipe *fly* dapat berganti menjadi **Hero** jika terkena serangan dari **Hero**.
 - *Goal state* dari *Monsters* tipe *boss* dapat berganti menjadi bom jika bom muncul. *Goal state* *monsters* tipe ini juga dapat berganti menjadi **Hero** jika terkena serangan dari **Hero**.
- c. Posisi dari *monsters* terhadap *goal state*

Pengecekan path dilakukan berdasarkan posisi dari x dan y dari *monsters*, dan posisi x dan y dari *goal state*.

5.2 Saran

Berdasarkan hasil penelitian yang dilakukan untuk mengimplementasikan algoritma A* untuk 3 tipe *monsters* pada permainan *Tower Defense*, ada beberapa fitur/fungsi yang dapat dikembangkan.

- a. Tingkat kesulitan dari permainan dari *Tower Defense*.

Tingkat kesulitan yang dimaksud adalah penambahan tantangan. Misal setelah *game* telah berjalan lebih dari 2 menit, muncul batu meteor dari langit, *monsters* yang muncul semakin banyak, dsb.

- b. Penambahan tipe *monsters* agar permainan semakin menantang

Pada permainan *Tower Defense*, saat ini hanya terdapat 3 tipe *monsters* yaitu normal, *fly*, dan *boss*. Adanya penambahan tipe *monsters* diharapkan dapat menambah variasi dari permainan.

- c. Implementasi *AI* untuk **Hero**

Pada permainan *Tower Defense*, **Hero** masih digerakan secara manual oleh pemain. Pemberian *AI* yang dimaksud adalah bahwa **Hero** dapat bergerak secara otomatis untuk melindungi *tower* dari serangan *monsters* dan dapat menjawab pertanyaan dengan tingkat akurasi 100%. Hal ini dapat digunakan untuk membandingkan skor yang diperoleh jika dimainkan oleh *AI*, dan skor yang diperoleh jika dimainkan secara manual oleh pemain.

DAFTAR PUSTAKA

- Adam, E. (2010). *Fundamentals of Game Design 2nd Edition*. California: New Riders Publishing.
- Heriksen, E. (2013). *Angry Birds will be bigger than Mickey Mouse and Mario. Is there a success formula for apps?* Diakses 17 September 2014, dari MIT: The Entrepreneurship Review : <http://miter.mit.edu/articleangry-birds-will-be-bigger-mickey-mouse-and-mario-there-success-formula-apps/>
- I.Zarembo, S.Kodors. (2013). *Pathfinding Algorithm Efficiency Analysis in 2D Grid*. RA Izdevniecība: Rēzekne.
- Millington, I. (2006). *Artificial Intelligence For Games*. San Francisco: Elsevier Inc.
- Nayak, D. (2014). *Analysis and Implementation of Admissible Heuristics in 8 Puzzle Problem*. India: Rourkela.
- Nouch, J. (2013, 02 14). Retrieved 09 25, 2014, from Pocket Gamer: <http://www.pocketgamer.biz/r/PG.Biz/Newzoo+news/news.asp?c=48541>
- Ola Davidson, J. P. (n.d.). *Game Design Patterns for Mobile Games*, 48.
- X.Cui, H.Shi. (2011). *A* Based Pathfinding In Modern COmputer Games*. Australia: Victoria University.