

**PENCARIAN RUTE TERPENDEK *NON-PLAYER
CHARACTER (NPC)* DENGAN METODE
FLOYD WARSHALL PADA GAME
WISATA BATU**

SKRIPSI

Oleh:
DEWI ISNAINI
NIM. 14650055



**JURUSAN TEKNIK INFORMATIKA
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM
MALANG**

2019

PENCARIAN RUTE TERPENDEK *NON-PLAYER*

**CHARACTER (NPC) DENGAN METODE
FLOYD WARSHALL PADA GAME
WISATA BATU**

SKRIPSI

**Diajukan kepada:
Fakultas Sains dan Teknologi
Universitas Islam Negeri (UIN)
Maulana Malik Ibrahim Malang
Untuk Memenuhi Salah Satu Persyaratan Dalam
Memperoleh Gelar Sarjana Komputer (S.Kom)**

**Oleh:
DEWI ISNAINI
NIM. 14650055**

**JURUSAN TEKNIK INFORMATIKA
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM
MALANG
2019**

LEMBAR PERSETUJUAN
PENCARIAN RUTE TERPENDEK *NON-PLAYER*
***CHARACTER (NPC)* DENGAN METODE**
***FLOYD WARSHALL* PADA GAME**
WISATA BATU

SKRIPSI

Oleh :
DEWI ISNAINI
NIM. 14650055

Telah Diperiksa dan Disetujui untuk Diuji

Tanggal : Juni 2019

Dosen Pembimbing I



Yunifa Miftachal Arif, M.T
NIP. 19830616 201101 1 004

Dosen Pembimbing II




Fachrul Kurniawan, M. MT
NIP. 19771020 200912 1 001

Mengetahui,

Ketua Jurusan Teknik Informatika
Fakultas Sains dan Teknologi
Universitas Islam Negeri Maulana Malik Ibrahim Malang




Dwi Cahyo Crysdiyan
NIP. 19740424 200901 1 008

LEMBAR PENGESAHAN
PENCARIAN RUTE TERPENDEK *NON-PLAYER*
***CHARACTER (NPC)* DENGAN METODE**
***FLOYD WARSHALL* PADA GAME**
WISATA BATU

SKRIPSI

Oleh :

DEWI ISNAINI
NIM. 14650055

Telah Dipertahankan di Depan Dewan Penguji
dan Dinyatakan Diterima Sebagai Salah Satu Persyaratan
untuk Memperoleh Gelar Sarjana Komputer (S.Kom)
Pada Tanggal Mei 2019

Susunan Dewan Penguji

- | | | |
|-----------------------|---|---|
| 1. Penguji Utama | : | <u>Hani Nurhayati, M.T</u>
NIP. 19780625 200801 2 006 |
| 2. Ketua Penguji | : | <u>Dr. Cahyo Crys dian</u>
NIP. 19740424 200901 1 008 |
| 3. Sekretaris Penguji | : | <u>Yunifa Miftachul Arif, M.T</u>
NIP. 19830616 201101 1 004 |
| 4. Anggota Penguji | : | <u>Fachrul Kurniawan, M. MT</u>
NIP. 19771020 200912 1 001 |

Tanda tangan

()
()
()
()

Mengetahui,
Ketua Jurusan Teknik Informatika
Fakultas Sains dan Teknologi
Universitas Islam Negeri Maulana Malik Ibrahim Malang



Dr. Cahyo Crys dian
NIP. 19740424 200901 1 008

HALAMAN MOTTO

"Orang-orang hebat di bidang apapun bukan baru bekerja karena mereka terinspirasi, namun mereka menjadi terinspirasi karena mereka lebih suka bekerja. Mereka tidak menyia-nyiakan waktu untuk menunggu inspirasi." (Ernest Newman)



HALAMAN PERSEMBAHAN

الْحَمْدُ لِلَّهِ رَبِّ الْعَالَمِينَ

Saya persembahkan karya ini dengan segala kerendahan hati, semoga skripsi ini bermanfaat dunia dan akhirat. Aamiin.

Dengan segala hormat, saya persembahkan karya ini kepada :

Kedua orang tua saya Bapak Madsiri dan Ibu Sainab, yang telah mendidik dengan penuh kasih sayang dan memberikan dukungan serta do'a untuk menuntut ilmu setinggi-tingginya.

Guru diskusi saya Galang Luhur Pekerti yang telah meluangkan waktu, tenaga dan pikiran demi terselesainya skripsi ini.

Keluarga Teknik Informatika 2014, keluarga Biner (Teknik Informatika angkatan 2014), serta seluruh keluarga besar Teknik Informatika UIN Maulana Malik Ibrahim Malang

Orang-orang yang saya sayangi, yang tak bisa saya sebutkan satu per satu.

Saya ucapkan terimakasih yang luar biasa. Semoga ukhwah kita tetap terjaga dan selalu diridhoi Allah SWT. Allahumma Aamiin.

PERNYATAAN KEASLIAN TULISAN

Saya yang bertandatangan di bawah ini:

Nama : Dewi Isnaini
NIM : 14650055
Fakultas/ Jurusan : Sains dan Teknologi/ Teknik Informatika
Judul Skripsi : **Pencarian Rute Terpendek *Non-Player Character* (NPC) Dengan Metode *Floyd Warshall* pada Game Wisata Batu.**

Menyatakan dengan sebenarnya bahwa skripsi yang saya tulis ini benar-benar merupakan hasil karya saya sendiri, bukan merupakan pengambil alihan data, tulisan atau pikiran orang lain yang saya akui sebagai hasil tulisan atau pikiran saya sendiri, kecuali dengan mencantumkan sumber cuplikan pada daftar pustaka.

Apabila dikemudian hari terbukti atau dapat dibuktikan skripsi ini hasil jiplakan, maka saya bersedia menerima sanksi atas perbuatan tersebut.

Malang, Juni 2019

Penulis



Dewi Isnaini
NIM. 14650055

KATA PENGANTAR

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ

Segala puji bagi Allah SWT, karena atas rahmat, hidayah serta karunia-Nya, penulis dapat menyelesaikan skripsi yang berjudul “Rancang Bangun Sistem *e-Learning* Berbasis *Microservices* dan *Domain Driven Design* (Studi Kasus Probistek UIN Maulana Malik Ibrahim Malang)” sebagai salah satu syarat untuk memperoleh gelar sarjana pada Program Studi Teknik Informatika jenjang Strata-1 Universitas Islam Negeri Maulana Malik Ibrahim Malang.

Salawat serta salam senantiasa terlimpahkan kepada Nabi Muhammad SAW, keluarga dan para sahabat yang telah membimbing umat dari gelapnya alam jahiliyah menuju cahaya islam yang diridoi Allah SWT.

Penulis menyadari adanya banyak keterbatasan yang penulis miliki, sehingga ada banyak pihak yang telah memberikan bantuan baik moril, materiil, nasihat dan semangat dalam menyelesaikan penelitian ini. Maka dari itu dengan segenap kerendahan hati penulis mengucapkan terimakasih kepada:

1. Prof Dr H Abd. Haris, M.Ag selaku rektor UIN Maulana Malik Ibrahim Malang.
2. Dr. Sri Harini, M.Si. selaku Dekan Fakultas Sains dan Teknologi UIN Maulana Malik Ibrahim Malang.
3. Dr. Cahyo Crysdiان selaku Ketua Jurusan Teknik Informatika Fakultas Sains dan Teknologi UIN Maulana Malik Ibrahim Malang.

4. Yunifa Miftachul Arif, M.T dan Fachrul Kurniawan, M. MT, selaku pembimbing yang senantiasa meluangkan waktu untuk membimbing, mengarahkan penulis, dan memberi masukan.
5. Hani Nurhayati, M.T dan Dr. Cahyo Crysdiyan selaku penguji yang telah membimbing dan memberikan masukan pada skripsi ini.
6. Seluruh Dosen Jurusan Teknik Informatika Fakultas Sains dan Teknologi UIN Maulana Malik Ibrahim Malang yang telah memberikan ilmu dan pengetahuan serta pengalaman.
7. Segenap civitas akademik Jurusan Teknik Informatika Fakultas Sains dan Teknologi UIN Maulana Malik Ibrahim Malang.
8. Kedua orang tua serta seluruh keluarga besar penulis yang senantiasa mendukung.
9. Sahabat-sahabat seperjuangan Jurusan Teknik Informatika Fakultas Sains dan Teknologi UIN Maulana Malik Ibrahim Malang.

Penulis menyadari dalam karya ini masih banyak kekurangan. Oleh karena itu penulis selalu menerima segala kritik dan saran dari pembaca. Semoga karya ini bermanfaat bagi seluruh pihak.

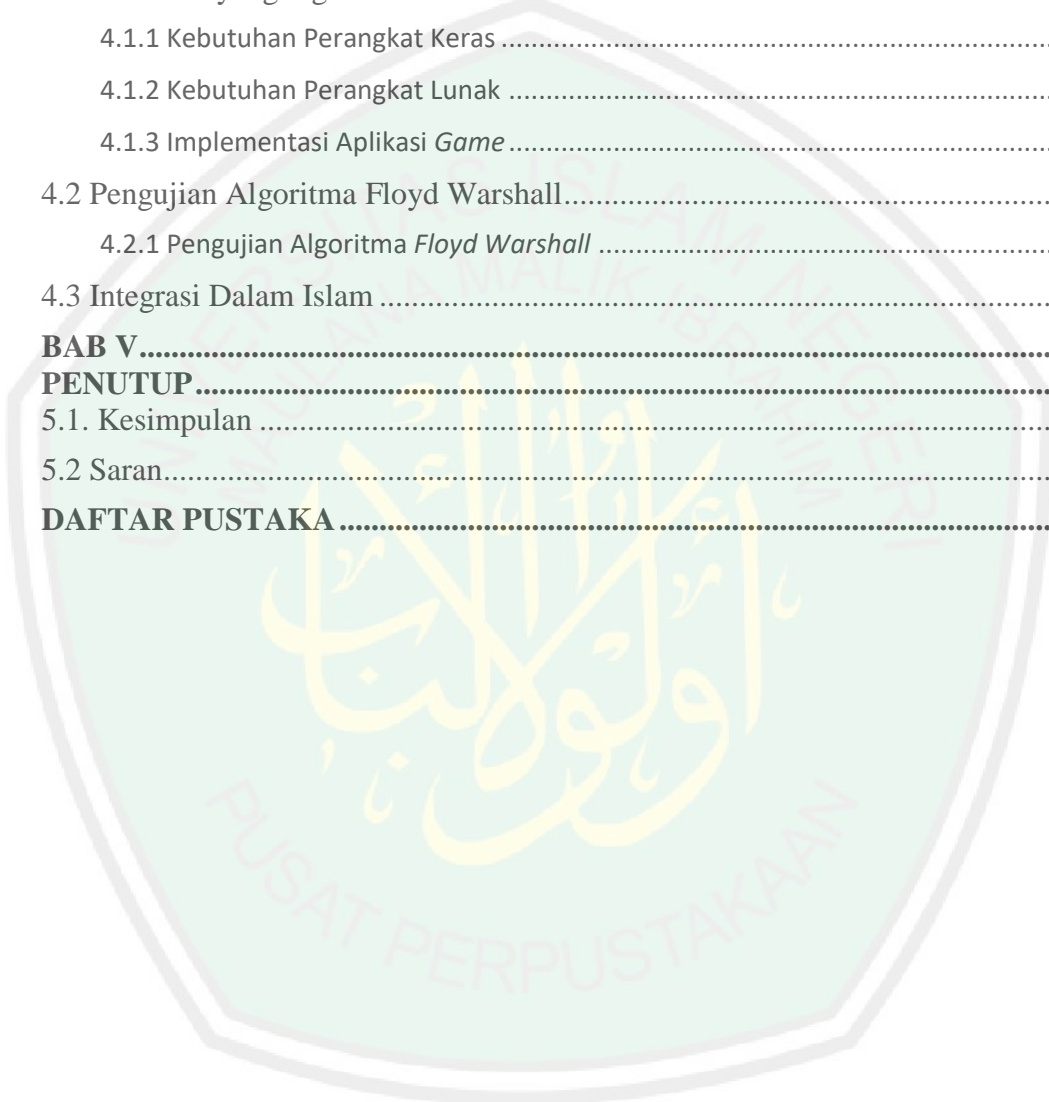
Malang, Juni 2019

Penulis

DAFTAR ISI

LEMBAR PERSETUJUAN	iii
LEMBAR PENGESAHAN	iv
HALAMAN MOTTO	v
HALAMAN PERSEMBAHAN.....	vi
PERNYATAAN KEASLIAN TULISAN	vii
KATA PENGANTAR.....	viii
DAFTAR ISI.....	x
DAFTAR GAMBAR.....	xii
DAFTAR TABEL.....	xiv
ABSTRAK	xv
ABSTRACT	xvi
المخلص.....	xvii
BAB I.....	1
PENDAHULUAN.....	1
1.1 Latar Belakang	1
1.2 Pernyataan Masalah	3
1.3 Tujuan Penelitian	3
1.4 Manfaat Penelitian	3
1.5 Batasan Penelitian	4
BAB II	5
TINJAUAN PUSTAKA	5
2.1 Kajian Pustaka	5
2.1.1 Game (Permainan)	5
2.1.2 Jenis-jenis <i>Game</i>	5
2.1.3 Genre Game	6
2.1.4 <i>Game</i> Pembelajaran	13
2.1.5 Non-Player Character (NPC)	14
2.1.6 Algoritma <i>Floyd Washall</i>	15
2.2 Unity	26
2.3 Penelitian Terkait	34
BAB III.....	36
DESAIN DAN PERANCANGAN SISTEM	36
3.1 Alur Cerita (<i>Storyline</i>)	36
3.2 Game Play.....	36
3.3 Konten Permainan.....	36
3.4 Layout Aplikasi.....	39
3.5 Perancangan Finite State Machine.....	41

3.6 Sumber Data.....	41
3.7 Flowchart	42
3.8 Proses Perhitungan Algoritma Floyd Warshall.....	43
BAB IV	48
HASIL DAN PEMBAHASAN	48
4.1 Peralatan yang digunakan	48
4.1.1 Kebutuhan Perangkat Keras	48
4.1.2 Kebutuhan Perangkat Lunak	49
4.1.3 Implementasi Aplikasi <i>Game</i>	49
4.2 Pengujian Algoritma Floyd Warshall.....	51
4.2.1 Pengujian Algoritma <i>Floyd Warshall</i>	51
4.3 Integrasi Dalam Islam	83
BAB V.....	89
PENUTUP.....	89
5.1. Kesimpulan	89
5.2 Saran.....	89
DAFTAR PUSTAKA.....	90



DAFTAR GAMBAR

Gambar 2.1 Gambar <i>Action</i>	7
Gambar 2.2 Gambar <i>Adventure</i>	8
Gambar 2.3 Gambar <i>Game Puzzle</i>	9
Gambar 2.4 Gambar <i>Game RPG</i>	10
Gambar 2.5 Gambar <i>Game Simulation</i>	11
Gambar 2.6 Gambar <i>Game Strategy</i>	12
Gambar 2.7 Gambar <i>Game Balapan</i>	12
Gambar 2.8 Gambar <i>Game Olahraga</i>	13
Gambar 2.9 Contoh Gambar Graf Berbobot	19
Gambar 2.10 Asset Store Unity	28
Gambar 2.11 Platform yang didukung Unity	30
Gambar 2.12 Physics	31
Gambar 2.13 Rendering.....	32
Gambar 2.14 Visual Studio 2017	33
Gambar 3.1 <i>Box</i>	38
Gambar 3.2 Peta	38
Gambar 3.3 Splash Screen.....	39
Gambar 3.4 Main menu	39
Gambar 3.5 Options	40
Gambar 3.6 Latar Permainan.....	41
Gambar 3.7 Finite State Machine	41
Gambar 3.8 Flowchart Algoritma <i>Floyd Warshall</i>	42
Gambar 4.1 Tampilan Splash Screen.....	49
Gambar 4.2 Tampilan Loading.....	50

Gambar 4.3 Tampilan Menu utama.....	50
Gambar 4.4 Tampilan Utama	51
Gambar 4.5 Jarak antar node	52
Gambar 4.6 Jarak node yang digambarkan dengan bentuk cube.....	52
Gambar 4.7 Posisi awal NPC dan player.....	53
Gambar 4.8 NPC menuju target	54
Gambar 4.9 Rute Terpendek yang berhasil ditemukan pada <i>Console Unity</i>	72
Gambar 4.10 <i>Timer/Stopwatch</i> pada <i>Console Unity</i>	73



DAFTAR TABEL

Tabel 2.1 Matriks D(0)	19
Tabel 2.2 Matriks D(1)	21
Tabel 2.3 Matriks D(2)	22
Tabel 2.4 Matriks D(3)	24
Tabel 2.5 Matriks D(4)	25
Tabel 3.1 Karakter	37
Tabel 3.2 Matriks D(0)	46
Tabel 3.3 Matriks D(18)	47
Tabel 4.1 Kebutuhan Perangkat Keras	48
Tabel 4.2 Kebutuhan Perangkat Lunak	49
Tabel 4.3 Keterangan Implementasi <i>Floyd Warshall</i>	54
Tabel 4.4 Pengujian di Jl. Trunojoyo menuju 3 tempat wisata	60
Tabel 4.5 Pengujian di Jl. Ir. Soekarno menuju 3 tempat wisata.....	66
Tabel 4.6 Pengujian di Jl. Ir. Semeru menuju 3 tempat wisata	73

ABSTRAK

Isnaini, Dewi. 2019. **Pencarian Rute Terpendek *Non Player Character* (NPC) Dengan Metode *Floyd Warshall* pada Game Wisata Batu**. Skripsi. Jurusan Teknik Informatika Fakultas Sains dan Teknologi Universitas Islam Negeri Maulana Malik Ibrahim Malang.

Pembimbing : (I) Yunifa Miftachul Arif, M.T

(II) Fachrul Kurniawan, M.MT.

Kata Kunci: *Pencarian Rute, Game Wisata Batu, Floyd Warshall*

Pariwisata sebagai penggerak sektor ekonomi dapat menjadi solusi bagi pemerintah dalam meningkatkan pembangunan ekonomi. Pengembangan kepariwisataan dapat membawa banyak manfaat dan keuntungan, sektor pariwisata tidak hanya menyentuh kelompok-kelompok ekonomi tertentu tetapi dapat menjangkau kalangan bawah. Karena masyarakat di sekitar obyek wisata dapat mendirikan berbagai kegiatan ekonomi misalnya tempat penginapan, rumah makan, tempat penukaran uang, layanan jasa transportasi dan lain-lain. Namun juga menimbulkan dampak negatif bagi masyarakat sekitar yakni kemacetan. *Game* ini untuk dijadikan sebuah media yang digunakan untuk mengetahui jarak atau rute terpendek menuju tempat wisata Kota Batu. *Game* Pencarian Rute Terpendek *Non Player Character* (NPC) Dengan Metode *Floyd Warshall* pada Game Wisata Batu adalah *game* berbasis desktop dengan menggunakan *floyd warshall* sebagai pathfinding pada *non player character* (npc) dan menggunakan engine unity 3D. Player bertugas, untuk mengikuti *non player character* (npc) untuk menuju tempat wisata. *Non Player Character* (NPC) mengimplementasikan sistem kecerdasan buatan yang akan mencari rute terpendek tempat wisata yang ingin dituju. Pada penelitian ini, memanfaatkan metode *Floyd Warshall* sebagai pencarian rute terdekat. Dalam penelitian ini dilakukan proses uji coba yang menghasilkan 94.44% sukses dari ketepatan dan kecepatan dalam menentukan jarak terpendek..

ABSTRACT

Isnaini, Dewi. 2019. **Searching for the Shortest Non Player Character (NPC) Routes with the Floyd Warshall Method on the Batu Tourism Game.** Skripsi. Jurusan Teknik Informatika Fakultas Sains dan Teknologi Universitas Islam Negeri Maulana Malik Ibrahim Malang.

Supervising Professor : (I) Yunifa Miftachul Arif, M.T,

(II) Fachrul Kurniawan, M.MT.

Key words: *Route Search, Wisata Kota Batu Game, Floyd Warshall*

Tourism, as a driver of the economic sector, can be a solution for the government in increasing economic development. The development of tourism can bring many benefits and profits. The tourism sector does not only touch certain economic groups but can reach the lower classes because people around tourism objects can establish various economic activities such as lodging, restaurants, money changers, transportation services and others. However, it also has a negative impact on the surrounding community, namely congestion. The purpose of this research and the development of this application are to accurately measure the use of *Floyd Warshall algorithm* in finding the shortest *Non Player Character* (NPC) route in the *Wisata Kota Batu* game and the benefits of this research to provide intelligence to the NPC to find the shortest route to tourist attractions. This game is used as a medium that is used to determine the distance or the shortest route to the tourist attractions of Batu city. The game of *Searching for the Shortest Route Non Player Character* (NPC) With the *Floyd Warshall Method* on Batu Tourism Games is a desktop-based game using *Floyd Warshall algorithm* as pathfinder on *Non Player Character* (NPC) and using a 3D unity engine. The player is in charge to follow the *Non Player Character* (NPC) to get to the place of travel. *Non Player Character* (NPC) implements an artificial intelligence system that will look for the shortest route to the destination you want to go. This study is utilizing the *Floyd Warshall algorithm* method as a search for the nearest route. In this study a trial process was carried out which resulted in a successful 94.44% of accuracy and speed in determining the shortest distance.

المخلص

إثنين دوي. 2019. البحث عن أقصر مسارات الشخصيات غير المشغلة (NPC) باستخدام طريقة Floyd Warshall على لعبة Stone للسياحة. المشرف : (1) يونيفة مفتاح العارف الماجستير. (2) فهورل كورنيوان الماجستير.

الكلمات الرئيسية: البحث عن طريق ، لعبة روك للسياحة ، فلويد ورشال

يمكن للسياحة كمحرك للقطاع الاقتصادي أن يكون حلاً للحكومة في زيادة التنمية الاقتصادية. إن تطوير السياحة يمكن أن يجلب العديد من الفوائد والفوائد ، لا يلمس قطاع السياحة مجموعات اقتصادية معينة فحسب ، بل يمكنه الوصول إلى الطبقات الدنيا. لأن الأشخاص المحيطين بالسياحة يمكنهم إنشاء أنشطة اقتصادية مختلفة مثل السكن والمطاعم ومحلات الصرافة وخدمات النقل وغيرها. لكن له أيضاً تأثير سلبي على المجتمع المحيط ، وهو الازدحام. تُستخدم هذه اللعبة كوسيلة تُستخدم لتحديد أقصر مسافة أو طريق إلى مناطق الجذب السياحي في مدينة باتو. ابحث عن لعبة أقصر شخصية غير لاعب (NPC) مع لعبة فلويد وارشال على روك توريسيت غيمز هي لعبة تعتمد على سطح المكتب باستخدام لعبة فلويد ورسهول كمسار على شخصية اللاعب (npc) واستخدام محرك الوحدة ثلاثية الأبعاد. اللاعب مسؤول ، لاتباع شخصية اللاعب (npc) للوصول إلى المكان الصحيح للسفر. يقوم برنامج (NPC) Non Player Character بتطبيق نظام ذكاء اصطناعي يبحث عن أقصر طريق إلى الوجهة التي تريد الذهاب إليها. في هذه الدراسة ، استخدم طريقة Floyd Warshall كبحث عن أقرب مسار. في هذه الدراسة تم إجراء تجربة تجريبية أدت إلى نجاح 94.44% من الدقة والسرعة في تحديد أقصر مسافة

BAB I PENDAHULUAN

1.1 Latar Belakang

Pariwisata sebagai penggerak sektor ekonomi dapat menjadi solusi bagi pemerintah dalam meningkatkan pembangunan ekonomi. Pengembangan kepariwisataan dapat membawa banyak manfaat dan keuntungan, sektor pariwisata tidak hanya menyentuh kelompok-kelompok ekonomi tertentu tetapi dapat menjangkau kalangan bawah. Karena masyarakat di sekitar obyek wisata dapat mendirikan berbagai kegiatan ekonomi misalnya tempat penginapan, rumah makan, tempat penukaran uang, layanan jasa transportasi dan lain-lain. Namun juga menimbulkan dampak negatif bagi masyarakat sekitar yakni kemacetan.

Kota Batu dikenal sebagai kota wisata di Indonesia karena potensi keindahan alam yang luar biasa. Pada masa Belanda kota Batu diberi julukan sebagai *De Kleine Zwitserland* atau Swiss kecil di Pulau Jawa. Dikarenakan sebagai salah satu tempat tujuan wisata Jawa Timur yang memiliki berbagai jenis wisata mulai dari wisata alam, wisata agro, wisata religi dan lain-lain.

Banyaknya objek wisata di Kota Batu membuat banyak pilihan dalam berwisata. Berwisata merupakan kebutuhan jasmani yang sangat penting, karena dengan berwisata dapat menghilangkan penat akibat aktivitas selama seharian. Sedangkan Pariwisata adalah suatu perjalanan yang dilakukan untuk sementara waktu, yang diselenggarakan dari suatu tempat ke tempat lain dengan maksud bukan usaha atau mencari nafkah di tempat yang dikunjungi, tetapi semata-mata untuk memenuhi keinginan yang beraneka ragam (Yoeti, 1980).

Di dalam ajaran islam juga dijelaskan bahwa untuk meninggikan pemahaman wisata harus mengaitkannya dengan tujuan-tujuan yang mulia, seperti dalam firman Allah dalam surat Al-Ankabut ayat 20 :

قُلْ سِيرُوا فِي الْأَرْضِ فَانظُرُوا كَيْفَ بَدَأَ الْخَلْقَ ثُمَّ اللَّهُ يُنشِئُ
النَّشْأَةَ الْآخِرَةَ إِنَّ اللَّهَ عَلَىٰ كُلِّ شَيْءٍ قَدِيرٌ ﴿٢٠﴾

Katakanlah: "Berjalanlah di (muka) bumi, maka perhatikanlah bagaimana Allah menciptakan (manusia) dari permulaannya, kemudian Allah menjadikannya sekali lagi. Sesungguhnya Allah Maha Kuasa atas segala sesuatu.

Ayat ini mengingatkan manusia tentang pemahaman wisata dalam Islam adalah safar untuk merenungi keindahan ciptaan Allah Ta'la, menikmati indahnya alam nan agung sebagai pendorong jiwa manusia untuk menguatkan keimanan terhadap keesaan Allah dan memotivasi menunaikan kewajiban hidup. Karena itu Rasulullah saw. Bersabda : “Hendaknya (wajib) bagi kalian bekerja atau beramal yang tidak memberatkan. Demi Allah, sesungguhnya Allah tidak akan pernah bosan sampai kalian sendiri merasa bosan.” HR. Muslim.

Untuk mempermudah kondisi ini, maka diperlukan suatu media pembelajaran untuk mengatasi atau mencari solusi jika terjadi kemacetan, maka dibuatlah sebuah media berbentuk game untuk mengurangi kemacetan agar mengetahui dan mencari jalur tercepat ke tempat-tempat wisata yang ada di Kota Batu. Dalam pencarian jalur menuju tempat yang ingin di kunjungi penulis akan menggunakan Algoritma *Floyd Warshall*. Algoritma *Floyd Warshall* merupakan jenis algoritma yang menggunakan pendekatan penyelesaian masalah dengan mencari nilai maksimum sementara pada setiap langkahnya. Nilai maksimum

sementara ini dikenal dengan istilah *local maximum*. Pada kebanyakan kasus, algoritma *Floyd Warshall* tidak akan menghasilkan solusi paling optimal, begitupun algoritma *Floyd Warshall* biasanya memberikan solusi yang mendekati nilai optimum dalam waktu yang cukup cepat.

1.2 Pernyataan Masalah

Berdasarkan latar belakang diatas identifikasi masalah yang telah dipaparkan, didapat masalah pokok yang memerlukan penyelesaian dan menjadi dasar dalam pengerjaan penelitian ini adalah Seberapa akurat algoritma *Floyd Warshall* dalam pencarian rute terpendek *Non Player Character* (NPC) pada game Wisata Kota Batu.

1.3 Tujuan Penelitian

Tujuan dari diadakannya penelitian ini dan pengembangan aplikasi ini adalah Untuk mengukur akurat penggunaan algoritma *Floyd Warshall* dalam pencarian rute terpendek *Non Player Character* (NPC) pada game Wisata Kota Batu.

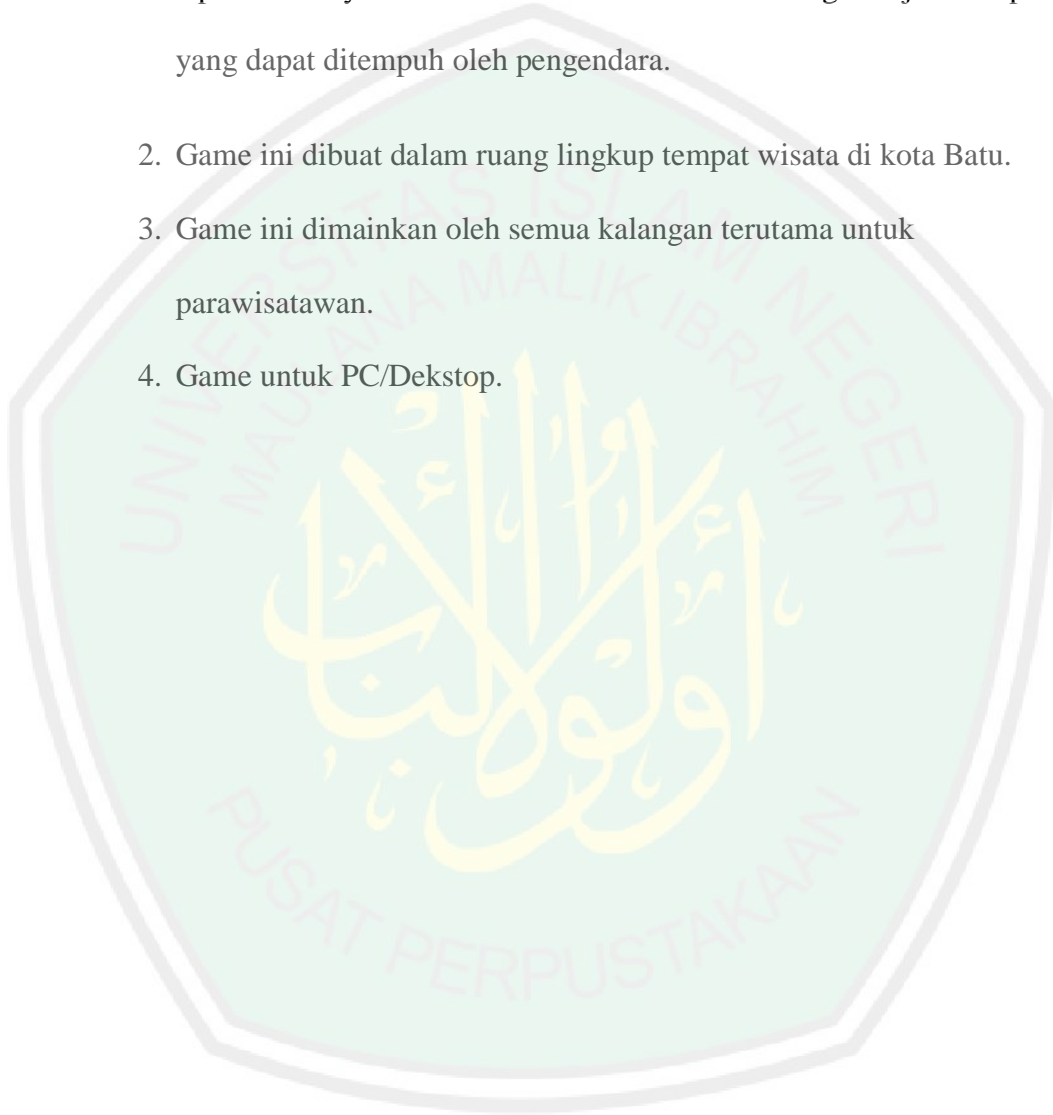
1.4 Manfaat Penelitian

Manfaat yang diharapkan dari penelitian ini adalah memberikan kecerdasan kepada NPC untuk menemukan jalur terpendek untuk menuju tempat wisata.

1.5 Batasan Penelitian

Batasan masalah yang penulis kemukakan dalam perancangan aplikasi ini adalah :

1. Aplikasi hanya akan memberikan informasi mengenai jalur terpendek yang dapat ditempuh oleh pengendara.
2. Game ini dibuat dalam ruang lingkup tempat wisata di kota Batu.
3. Game ini dimainkan oleh semua kalangan terutama untuk parawisatawan.
4. Game untuk PC/Dekstop.



BAB II TINJAUAN PUSTAKA

2.1 Kajian Pustaka

2.1.1 Game (Permainan)

Menurut Ernest Adams (2010), game adalah jenis kegiatan bermain yang dilakukan dalam konteks berpura-pura dalam dunia nyata, dimana para peserta berusaha untuk mencapai suatu tujuan yang dilakukan sesuai dengan aturan yang ada. Sedangkan Menurut Anggra (Zulfadli , 2010;6) game adalah sesuatu yang dapat dimainkan dengan aturan tertentu sehingga ada yang menang dan ada yang kalah, biasanya dalam konteks tidak serius atau dengan tujuan refreshing. Suatu cara belajar yang digunakan dalam menganalisa interaksi antara sejumlah pemain maupun perorangan yang menunjukkan strategi-strategi yang rasional.

Game terdiri atas sekumpulan peraturan yang membangun situasi bersaing dari dua sampai beberapa orang atau kelompok dengan memilih strategi yang dibangun untuk memaksimalkan kemenangan sendiri atau pun untuk meminimalkan kemenangan lawan. Peraturan-peraturan menentukan kemungkinan tindakan untuk setiap pemain, sejumlah keterangan diterima setiap pemain sebagai kemajuan bermain, dan sejumlah kemenangan atau kekalahan dalam situasi.

2.1.2 Jenis-jenis *Game*

Ada beberapa jenis platform di dunia game yang selalu dipilih oleh pengguna *games*, yaitu :

1. ***Console games***, yaitu video games yang dimainkan menggunakan console tertentu, seperti Playstation 2, Playstation 3, XBOX 360, dan Nintendo Wii.

2. **PC Games** , yaitu video game yang dimainkan menggunakan Personal Computers.
3. **Arcade games**, yaitu yang sering disebut ding-dong di Indonesia, biasanya berada di daerah / tempat khusus dan memiliki box atau mesin yang memang khusus di design untuk jenis video games tertentu dan tidak jarang bahkan memiliki fitur yang dapat membuat pemainnya lebih merasa “masuk” dan “menikmati”, seperti pistol, kursi khusus, sensor gerakan, sensor injakkan dan stir mobil (beserta transmisinya tentunya).
4. **Mobile games**, yaitu yang dapat dimainkan atau khusus untuk mobile phone atau PDA.
5. **Handheld games**, yaitu yang dimainkan di console khusus video game yang dapat dibawa kemana-mana, contoh Nintendo DS dan Sony PSP.

2.1.3 Genre Game

Game menurut Lindsay Grace (2005) dibagi atas beberapa genre, diantaranya yaitu:

1. *Action*

Action game yang mengutamakan gerak. Permainan jenis ini membutuhkan ketangkasan/respon yang cepat dari pemain. Permainan aksi (*Action game*) mengajak pemain untuk menggunakan refleks, akurasi serta waktu yang tepat untuk menyelesaikan sebuah tantangan. Merupakan genre dasar dari sebuah permainan dan digunakan di hampir semua permainan yang ada. Dalam permainan aksi, biasanya terdapat pertempuran. Terdapat banyak sub-genre dari

permainan aksi, seperti *Fighting games* dan *First-person shooter (FPS)*, *drive and shoot*, *shoot em up*, *beat em up* dan *light gun shooting*.

Gambar 2.1 Gambar Action

Sumber : <https://www.pricebook.co.id/article/review/2016/01/26/3593/ada-4-jenis-dan-11-genre->



game-yang-mana-favorit-kamu

2. *Adventure*

Adventure game adalah game yang menawarkan eksplorasi area dan memecahkan teka-teki sebagai daya tarik utama. Dilihat dari *game-game* yang sudah ada, *game* dengan *genre Adventure* menawarkan cerita yang paling menarik dibandingkan *game* dengan *genre* lain. Penalaran, kreatifitas, dan rasa ingin tahu adalah keterampilan yang paling dibutuhkan untuk memainkan *game* ini dengan baik. *Game* jenis ini contohnya, *Tomb rider* hingga *Grand Theft Auto*.



Gambar 2.2 Gambar Adventure

Sumber : <https://www.pricebook.co.id/article/review/2016/01/26/3593/ada-4-jenis-dan-11-genre-game-yang-mana-favorit-kamu>

3. *Puzzle*

Game jenis ini merupakan *genre game* yang menekankan pada pemecahan teka-teki oleh pemainnya. Tipe-tipe teka-teki yang tersedia pada *genre* ini adalah tantangan logis dimana pemain harus memutar otaknya dan berkejaran dengan waktu yang ditentukan. Pada *game* jenis ini biasanya memiliki peraturan yang tidak terlalu rumit begitu juga dengan cara mainnya. *Genre game* ini merupakan *genre game* yang sering ditemui dan bisa dimainkan secara online maupun offline karena berbagai perangkat. *Game* jenis ini biasanya dimainkan oleh pemain tunggal.



Gambar 2.3 Gambar Game Puzzle

Sumber : <https://pasberita.com/jenis-genre-game/>

4. *Role Playing Game*

Game jenis ini sesuai dengan terjemahannya, bermain peran, memiliki penekanan pada tokoh/peran perwakilan pemain didalam permainan, yang biasanya adalah tokoh utamanya, dimana seiring kita memainkannya, karakter tersebut dapat berubah dan berkembang ke arah yang diinginkan pemain.



Gambar 2.4 Game RPG

Sumber : <https://www.pricebook.co.id/article/review/2016/01/26/3593/ada-4-jenis-dan-11-genre-game-yang-mana-favorit-kamu>

5. *Simulation*

Game dengan *genre* ini hadir dengan konsep yang mirip kenyataan. Semua faktor dalam *game* akan dibuat semirip dunia nyata, ada yang menceritakan kehidupan manusia atau kehidupan menjadi seorang walikota membangun sebuah kota. Dalam *game* ini semua keputusanmu berpengaruh pada karakter yang dimainkan. Inti dari *game* ini membuat pemain harus berpikir untuk mendirikan, membangun hingga mengatasi masalah dengan dana terbatas. Contoh *game* ini ialah *The Sims* dan *Sim City*.



Gambar 2.5 Game Simulation

Sumber : <https://www.pricebook.co.id/article/review/2016/01/26/3593/ada-4-jenis-dan-11-genre-game-yang-mana-favorit-kamu>

6. *Strategy*

Game jenis strategi, layaknya bermain catur, lebih memerlukan keahlian berpikir dan memutuskan setiap gerakan secara tiba-tiba dan terencana. *Game* strategi biasanya memberikan pemain atas kendali tidak hanya satu orang tapi minimal sekelompok orang dengan berbagai jenis tipe kemampuan, sampai kendaraan, bahkan hingga pembangunan berbagai dari tema ceritanya.



Gambar 2.6 Gambar Game Strategy

Sumber : <https://www.pricebook.co.id/article/review/2016/01/26/3593/ada-4-jenis-dan-11-genre-game-yang-mana-favorit-kamu>

7. Game Balapan

Game jenis ini mensimulasikan balapan dalam dunia nyata kedalam game, baik itu balapan mobil, motor maupun balap alat transportasi lainnya.



Gambar 2.7 Gambar *Game* Balapan

Sumber : <https://www.pricebook.co.id/article/review/2016/01/26/3593/ada-4-jenis-dan-11-genre-game-yang-mana-favorit-kamu>

8. *Game* Olahraga

Game jenis ini biasanya mensimulasikan semua jenis olahraga menjadi sebuah game baik itu dimainkan di PC maupun *console* lainnya.



Gambar 2.8 Gambar *Game* Olahraga

2.1.4 *Game* Pembelajaran

Menurut Aldrich (2009), terdapat beberapa jenis game yang dapat dimanfaatkan dalam pendidikan yaitu :

1. *Educational Simulations* : game yang bertujuan untuk sebuah pembelajaran.

2. *Serious Games* : *Serious game* suatu konsep *game* dengan tujuan untuk kepentingan training, advertising, simulasi, edukasi. Atau dengan kata lain untuk kepentingan yang sifatnya non-entertainment. Implementasinya, *serious game* ini bisa diterapkan untuk berbagai tingkatan usia serta dengan berbagai genre dan teknologi *game*.
3. *Frame Games*
4. *Class Games*
5. *Virtual World*

Perbedaan dari setiap jenis kemudian dibedakan menjadi 4 (empat) aspek menurut :

- a. Tujuan pembelajaran yang dapat dicapai
- b. Kritea Suksesnya
- c. Kebutuhan teknologi untuk diimplementasikan
- d. Contoh dari jenis *game* tersebut

Game pembelajaran adalah *game* digital yang dirancang untuk pengayaan pendidikan (mendukung pengajaran dan pembelajaran), menggunakan teknologi multimedia interaktif.

2.1.5 Non-Player Character (NPC)

NPC atau Non-Player Character adalah jenis *otonomous agent* yang ditujukan untuk penggunaan komputer animasi dan media interaktif seperti *games* dan *virtual reality*. Agen ini mewakili tokoh dalam cerita atau permainan dan memiliki kemampuan untuk improvisasi tindakan mereka. Ini adalah kebalikan dari seseorang tokoh dalam sebuah film animasi, yang

tindakannya ditulis di muka, dan untuk “*avatar*” dalam sebuah permainan atau *virtual reality*, tindakan yang diarahkan real time oleh pemain. Dalam permainan, karakter *otonom* biasanya disebut *Non-Player Character* (NPC). (Yunifa Miftachul Arif, 2010).

Secara garis besar maka NPC dapat diartikan sebagai karakter *game* yang sepenuhnya dikendalikan oleh komputer dan tidak dapat dimainkan oleh *player*. Pengendalian NPC umumnya menggunakan bidang ilmu kecerdasan buatan.

Kecerdasan buatan dimasukkan dengan tujuan agar NPC dapat melakukan pekerjaan seperti yang dapat dilakukan manusia. Beberapa macam bidang yang menggunakan kecerdasan buatan antara lain sistem pakar, permainan komputer, logika fuzzy, jaringan syaraf tiruan dan robotika.

Kecerdasaan buatan menjadikan NPC memiliki gerak variatif, mampu memainkan perannya sebagai penyempurna permainan dan menjadikan *gameplay* lebih asik dan menantang.

2.1.6 Algoritma Floyd Washall

a. Pengertian Algoritma Floyd Warshall

Algoritma yang ditemukan oleh Warshall untuk mencari lintasan terpendek merupakan algoritma yang sederhana dan mudah implementasinya. Masukan Algoritma Warshall ada matriks hubung graf berlabel dan keluarannya adalah lintasan terpendek dari semua titik (Siang, 2004).

Menurut Novandi, sebagaimana dikutip oleh Nur & Setiawan, (2013:21), Algoritma Floyd-Warshall adalah sebuah algoritma analisis graf untuk mencari bobot minimum dari graf berarah. Dalam pengertian lain Algoritma Floyd-Warshall adalah suatu metode yang melakukan pemecahan masalah dengan memandang solusi yang akan diperoleh sebagai suatu keputusan yang saling terkait. Artinya solusi-solusi tersebut dibentuk dari solusi yang berasal dari tahap sebelumnya dan ada kemungkinan solusi lebih dari 21 dari satu. Algoritma Floyd-Warshall memiliki input graf berbobot (V,E) , yang berupa daftar titik (*node/verteks* V) dan daftar sisi (*edge* E). Jumlah bobot sisi-sisi pada sebuah lintasan adalah bobot sisi tersebut. Sisi pada E diperbolehkan memiliki bobot negatif, akan tetapi tidak diperbolehkan bagi graf ini untuk memiliki siklus dengan bobot negatif. Algoritma ini menghitung bobot terkecil dari semua sisi yang menghubungkan sebuah pasangan titik dan melakukannya sekaligus untuk semua pasangan titik. Prinsip yang dipegang oleh algoritma Floyd-Warshall adalah prinsip optimalitas, yaitu jika solusi total optimal, maka bagian solusi sampai suatu tahap (misalnya tahap ke- i) juga optimal.

Menurut Siang, sebagaimana dikutip oleh Sani *et al.* (2013:3), algoritma Floyd-Warshall untuk mencari lintasan terpendek, Dimisalkan $W(0)$ adalah matriks ketetanggaan awal graf berarah berbobot. W^* adalah matriks ketetanggaan berbobot kecil dengan sama dengan W_{ij} * lintasan terpendek dari titik V_i ke V_j .

- 1) $W = W_0$
- 2) Untuk $k = 1$ hingga, lakukan :

Untuk $i = 1$ hingga, lakukan :

Untuk $j = 1$ hingga, lakukan :

Jika $W [i, j] > W[i,k] + W [i,k]$ maka

Tukar $W [i,j]$ dengan $W [i, k] + W [k, j]$

3) $W^* = W$

Keterangan :

W = matriks

W_0 = adalah matriks ketetanggaan awal

k = iterasi 1 sampai ke- n

i = titik awal v_i

j = titik awal v_j

W^* = hasil matriks setelah perbandingan

b. Algoritma Floyd Warshall Untuk Graf Tidak Berarah

Algoritma Floyd-Warshall dikembangkan oleh R. W. Floyd sehingga Algoritma Floyd-Warshall dapat digunakan untuk mencari jarak antara semua titik dalam graf. Algoritma ini sangat efisien dari sudut pandang penyimpanan data karena dapat diimplementasikan dengan hanya pengubahan sebuah matriks jarak.

Algoritma Floyd-Warshall memiliki input grsftsk berarah dan berbobot (V,E) , yang berupa himpunan titik (titik V) dan himpunan sisi (sisi E). Bobot sisi e dapat diberi symbol $d(i,j)$

Diketahui n titik dalam graf tidak berarah adalah $V_1, V_2, V_3, \dots, V_n$ untuk menentukan lintasan terpendek di antara semua pasangan titik, dengan langkah sebagai berikut :

Langkah 1 = untuk $i \neq j$, jika $v_i v_j$ adalah sisi, ambil $d(i,j)$ sebagai bobot dari sisi tersebut. Jika tidak ada sisi yang menghubungkan langsung antara i dan j ditulis $d(i,j) = \infty$. Untuk $i=j$, maka ditulis $d(i,j)=0$.

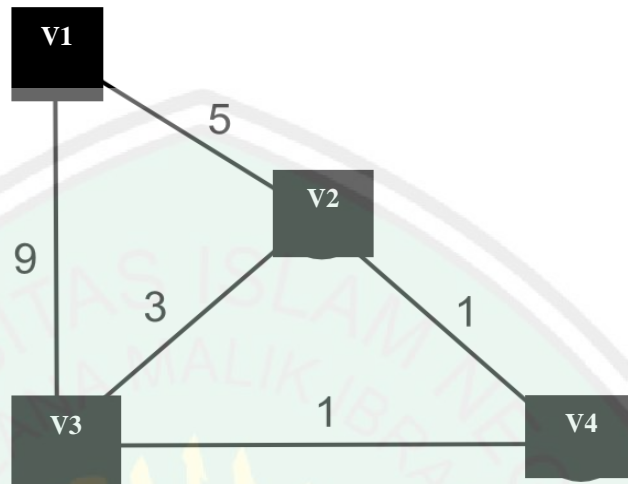
Langkah 2 : untuk $k=1$ sampai n

Untuk $i, j = 1$ sampai n Ditulis $d(i,j) = \min \{ +d(k,j) \}$

Nilai akhir dari $d(i,j)$ adalah jarak dari v_i ke v_j . (Goodaire & Parmeter, 1998:382)

Dari prosedur di atas dapat dilihat bahwa pada iterasi ke- k ($1 \leq k \leq n$) Mula-mula algoritma untuk jarak dari v_i ke v_j adalah panjang dari sisi v_i ke v_j adalah panjang sisi $v_i v_j$. Setelah iterasi pertama pada langkah 2 ($k=1$), jarak yang diperoleh digantikan dengan panjang dari lintasan $v_i v_l v_j$. Setelah iterasi k pada algoritma ini dapat ditentukan jarak terpendek dari v_i ke v_j pada titik-

titik, v_1, v_2, \dots, v_k . Jarak adalah setelah iterasi ke $k=n$ dengan mengambil $d(i,j)$ sebagai lintasan terpendek dari i ke j .



Gambar 2.9 Contoh Gambar Graf Berbobot

Bentuk Matriksnya sebagai berikut:

Tabel 2.1 Matrik D(0)

	V1	V2	V3	V4		V1	V2	V3	V4
V1	V1,V1	V1,V2	V1,V3	V1,V4	V1	0	5	9	∞
V2	V2,V1	V2,V2	V2,V3	V2,V4	V2	5	0	3	1
V3	V3,V1	V3,V2	V3,V3	V3,V4	V3	9	3	0	1
V4	V4,V1	V4,V2	V4,V3	V4,V4	V4	∞	1	1	0

K= 1

$$d(1,1) = \min \{d(1,1), d(1,1) + d(1,1)\} = \min \{0, 0 + 0\} = 0$$

$$d(1,2) = \min \{d(1,2), d(1,1) + d(1,2)\} = \min \{5, 0 + 5\} = 5$$

$$d(1,3) = \min \{d(1,3), d(1,1) + d(1,3)\} = \min \{9, 0 + 9\} = 9$$

$$d(1,4) = \min \{d(1,4), d(1,1) + d(1,4)\} = \min \{\infty, 0 + \infty\} = \infty$$

$$d(2,1) = \min \{d(2,1), d(2,1) + d(1,1)\} = \min \{5, 5 + 0\} = 5$$

$$d(2,2) = \min \{d(2,2), d(2,1) + d(1,2)\} = \min \{0, 5 + 5\} = 0$$

$$d(2,3) = \min \{d(2,3), d(2,1) + d(1,3)\} = \min \{3, 5 + 9\} = 3$$

$$d(2,4) = \min \{d(2,4), d(2,1) + d(1,4)\} = \min \{1, 5 + \infty\} = 1$$

$$d(3,1) = \min \{d(3,1), d(3,1) + d(1,3)\} = \min \{9, 9 + 0\} = 9$$

$$d(3,2) = \min \{d(3,2), d(3,1) + d(1,2)\} = \min \{3, 9 + 5\} = 3$$

$$d(3,3) = \min \{d(3,3), d(3,1) + d(1,3)\} = \min \{0, 9 + 9\} = 0$$

$$d(3,4) = \min \{d(3,4), d(3,1) + d(1,4)\} = \min \{1, 9 + \infty\} = 1$$

$$d(4,1) = \min \{d(4,1), d(4,1) + d(1,1)\} = \min \{\infty, \infty + 0\} = \infty$$

$$d(4,2) = \min \{d(4,2), d(4,1) + d(1,2)\} = \min \{1, \infty + 5\} = 1$$

$$d(4,3) = \min \{d(4,3), d(4,1) + d(1,3)\} = \min \{1, \infty + 9\} = 1$$

$$d(4,4) = \min \{d(4,4), d(4,1) + d(1,4)\} = \min \{0, \infty + \infty\} = 0$$

Sehingga diperoleh matriksnya sebagai berikut:

Tabel 2.2 Matrik D(1)

	V1	V2	V3	V4
V1	0	5	9	∞
V2	5	0	3	1
V3	9	3	0	1
V4	∞	1	1	0

K=2

$$d(1,1) = \min \{d(1,1), d(1,2) + d(2,1)\} = \min \{0, 5 + 5\} = 0$$

$$d(1,2) = \min \{d(1,2), d(1,2) + d(2,2)\} = \min \{5, 5 + 0\} = 5$$

$$d(1,3) = \min \{d(1,3), d(1,2) + d(2,3)\} = \min \{9, 5 + 3\} = 8$$

$$d(1,4) = \min \{d(1,4), d(1,2) + d(2,4)\} = \min \{\infty, 5 + 1\} = 6$$

$$d(2,1) = \min \{d(2,1), d(2,2) + d(2,1)\} = \min \{0, 0 + 0\} = 0$$

$$d(2,2) = \min \{d(2,2), d(2,2) + d(2,2)\} = \min \{0, 0 + 0\} = 0$$

$$d(2,3) = \min \{d(2,3), d(2,2) + d(2,3)\} = \min \{3, 0 + 3\} = 3$$

$$d(2,4) = \min \{d(2,4), d(2,2) + d(2,4)\} = \min \{1, 0 + 1\} = 1$$

$$d(3,1) = \min \{d(3,1), d(3,2) + d(2,3)\} = \min \{9, 3 + 5\} = 8$$

$$d(3,2) = \min \{d(3,2), d(3,2) + d(2,2)\} = \min \{3, 3 + 0\} = 3$$

$$d(3,3) = \min \{d(3,3), d(3,2) + d(2,3)\} = \min \{0, 3 + 3\} = 0$$

$$d(3,4) = \min \{d(3,4), d(3,2) + d(2,4)\} = \min \{1, 3 + 1\} = 1$$

$$d(4,1) = \min \{d(4,1), d(4,2) + d(2,1)\} = \min \{\infty, 1 + 5\} = 6$$

$$d(4,2) = \min \{d(4,2), d(4,2) + d(2,2)\} = \min \{1, 1 + 0\} = 1$$

$$d(4,3) = \min \{d(4,3), d(4,2) + d(2,3)\} = \min \{1, 1 + 3\} = 1$$

$$d(4,4) = \min \{d(4,4), d(4,2) + d(2,4)\} = \min \{0, 1 + 1\} = 0$$

Sehingga diperoleh matriknya sebagai berikut :

Tabel 2.3 Matrik D(2)

	V1	V2	V3	V4
V1	0	5	8	6
V2	5	0	3	1
V3	8	3	0	1
V4	6	1	1	0

K = 3

$$d(1,1) = \min \{d(1,1), d(1,3) + d(3,1)\} = \min \{0, 8 + 8\} = 0$$

$$d(1,2) = \min \{d(1,2), d(1,3) + d(3,2)\} = \min \{5, 8 + 3\} = 5$$

$$d(1,3) = \min \{d(1,3), d(1,3) + d(3,3)\} = \min \{8, 8 + 0\} = 8$$

$$d(1,4) = \min \{d(1,4), d(1,3) + d(3,4)\} = \min \{9, 8 + 1\} = 9$$

$$d(2,1) = \min \{d(2,1), d(2,3) + d(3,1)\} = \min \{5, 3 + 8\} = 5$$

$$d(2,2) = \min \{d(2,2), d(2,3) + d(3,2)\} = \min \{0, 3 + 3\} = 0$$

$$d(2,3) = \min \{d(2,3), d(2,3) + d(3,3)\} = \min \{3, 3 + 0\} = 3$$

$$d(2,4) = \min \{d(2,4), d(2,3) + d(3,4)\} = \min \{1, 3 + 1\} = 1$$

$$d(3,1) = \min \{d(3,1), d(3,3) + d(3,3)\} = \min \{8, 0 + 8\} = 8$$

$$d(3,2) = \min \{d(3,2), d(3,3) + d(3,2)\} = \min \{3, 0 + 3\} = 3$$

$$d(3,3) = \min \{d(3,3), d(3,3) + d(3,3)\} = \min \{0, 0 + 0\} = 0$$

$$d(3,4) = \min \{d(3,4), d(3,3) + d(3,4)\} = \min \{1, 0 + 1\} = 1$$

$$d(4,1) = \min \{d(4,1), d(4,3) + d(3,1)\} = \min \{6, 1 + 8\} = 6$$

$$d(4,2) = \min \{d(4,2), d(4,3) + d(3,2)\} = \min \{1, 1 + 3\} = 1$$

$$d(4,3) = \min \{d(4,3), d(4,3) + d(3,3)\} = \min \{1, 1 + 0\} = 1$$

$$d(4,4) = \min \{d(4,4), d(4,3) + d(3,4)\} = \min \{0, 1 + 1\} = 0$$

Sehingga diperoleh matriknya sebagai berikut :

Tabel 2.4 Matrik D(3)

	V1	V2	V3	V4
V1	0	5	8	6
V2	5	0	3	1
V3	8	3	0	1
V4	6	1	1	0

$K = 4$

$$d(1,1) = \min \{d(1,1), d(1,4) + W(4,1)\} = \min \{0, 6 + 5\} = 0$$

$$d(1,2) = \min \{d(1,2), d(1,4) + W(4,2)\} = \min \{5, 6 + 1\} = 5$$

$$d(1,3) = \min \{d(1,3), d(1,4) + d(4,3)\} = \min \{8, 6 + 1\} = 7$$

$$d(1,4) = \min \{d(1,4), d(1,4) + d(4,4)\} = \min \{6, 6 + 0\} = 6$$

$$d(2,1) = \min \{d(2,1), d(2,4) + d(4,1)\} = \min \{5, 1 + 6\} = 5$$

$$d(2,2) = \min \{d(2,2), d(2,4) + d(4,2)\} = \min \{0, 1 + 1\} = 0$$

$$d(2,3) = \min \{d(2,3), d(2,4) + d(4,3)\} = \min \{3, 3 + 1\} = 2$$

$$d(2,4) = \min \{d(2,4), d(2,4) + d(4,4)\} = \min \{1, 1 + 0\} = 1$$

$$d(3,1) = \min \{d(3,1), d(3,4) + d(4,3)\} = \min \{8, 1 + 6\} = 7$$

$$d(3,2) = \min \{d(3,2), d(3,4) + d(4,2)\} = \min \{3, 1 + 1\} = 2$$

$$d(3,3) = \min \{d(3,3), d(3,4) + d(4,3)\} = \min \{0, 1 + 1\} = 0$$

$$d(3,4) = \min \{d(3,4), d(3,4) + d(4,4)\} = \min \{1, 1 + 0\} = 1$$

$$d(4,1) = \min \{d(4,1), d(4,4) + d(4,1)\} = \min \{6, 0 + 6\} = 6$$

$$d(4,2) = \min \{d(4,2), d(4,4) + d(4,2)\} = \min \{1, 0 + 1\} = 1$$

$$d(4,3) = \min \{d(4,3), d(4,4) + d(4,3)\} = \min \{1, 0 + 1\} = 1$$

$$d(4,4) = \min \{d(4,4), d(4,4) + d(4,4)\} = \min \{0, 0 + 0\} = 0$$

Sehingga diperoleh matriknya sebagai berikut :

Tabel 2.5 Matrik D(4)

	V1	V2	V3	V4
V1	0	5	7	6
V2	5	0	2	1
V3	7	2	0	1
V4	6	1	1	0

Sehingga diperoleh suatu lintasan terpendek pada setiap titiknya. Dari matriks di atas dapat ditarik sebuah kesimpulan bawa jarak V_1 titik ke V_1 adalah 0, jarak dari titik V_1 ke V_2 adalah 5, V_1 ke V_3 adalah 7, V_1 ke V_4 adalah 6 dan sebagainya.

a. *Pseudocode* Algoritma Floyd Warshall

```

Function fw (int[1..n, 1..n] graph) {
  // Inisialisasi
  var int [1..n,1..n] jarak := graph
  var int [1..n,1..n] sebelum
  for i from 1 to n
    for j from 1 to n
      if jarak [i,j] < Tak-hingga
        sebelum[i,j] := i
  // Perulangan utama pada algoritma
  for k from 1 to n
    for i from 1 to n
      for j from 1 to n
        if jarak [i,j] > jarak [i,k] + jarak [k,j]
          jarak [i,j] = jarak [i,k] + jarak
[k,j]
          sebelum[i,j] = sebelum [k,j]
  return jarak

```

2.2 Unity

Unity merupakan perangkat lunak untuk pengembangan game *multiplatform*. Editor pada Unity dibuat dengan *user interface* yang sederhana dan mudah dipahami oleh *game developer* pemula sekalipun. Grafis pada unity dibuat dengan grafis tingkat tinggi untuk OpenGL dan DirectX. Unity mendukung semua format file, terutamanya format umum. Unity cocok dengan versi 64-bit. Dapat beroperasi pada Mac OS X dan Windows dan dapat menghasilkan *game* untuk Mac, Windows, Wii, iPhone, iPad maupun Android.

Unity merupakan salah satu *game engine* paling terkemuka dewasa ini. Unity adalah sebuah *software developer* yang terintegrasi untuk menciptakan *video game* atau konten lainnya seperti visualisasi arsitektur atau real-time animasi baik yang bernuansa 2D maupun 3D. Unity dapat digunakan pada microsoft Windows dan Mac OS X. Permainan yang dihasilkan dapat dijalankan secara *multiplatform*. Unity juga dapat menghasilkan permainan untuk browser dengan menggunakan *plugin Unity Web Player*.

Unity secara lebih rinci dapat dimanfaatkan untuk pengembangan 3D *video game*, *real time* animasi 3D dan visualisasi arsitektur maupun konten interaktif serupa lainnya. Editor Unity dapat menggunakan plugin untuk *web player* dan menghasilkan *game browser* yang didukung oleh Windows maupun Mac. Plugin web player dapat juga dipakai untuk widget Mac. Unity juga mendukung *console* terbaru seperti PlayStation 3 dan Xbox 360. Tahun 2009 Unity Technology menjadi 5 perusahaan game terbesar di dunia setelah tahun sebelumnya di tahun 2006, menjadi juara dua pada *Apple Design Awards*. Hingga puncaknya di tahun 2010 Unity berhasil memperoleh *Technology Innovation Award* yang diberikan oleh *Wall Street Journal*.

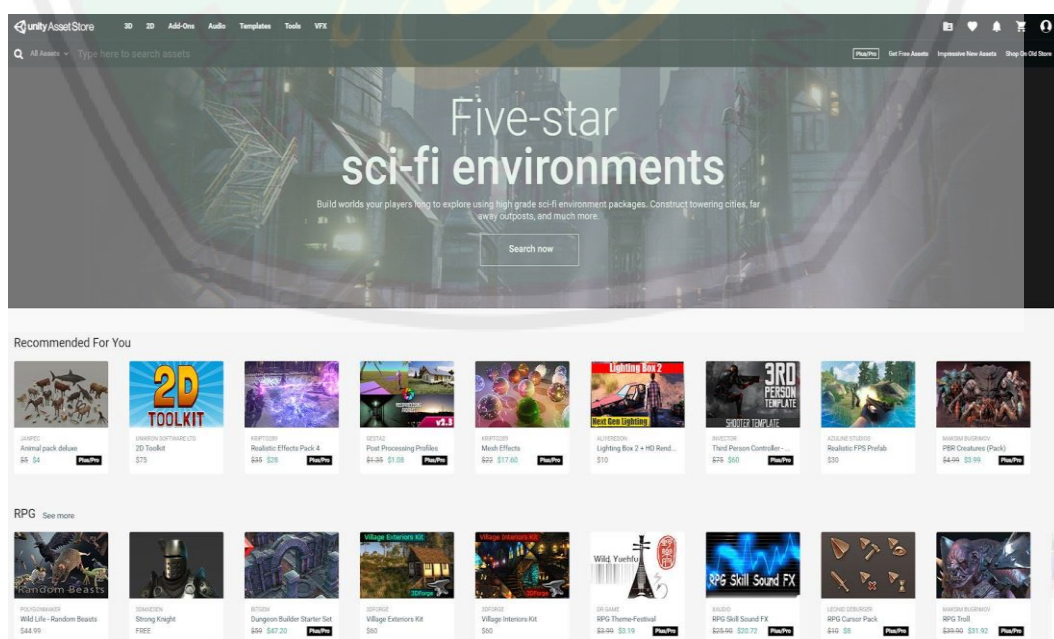
Unity Technology yang merupakan pengembang resmi Unity juga menyediakan *asset store*. *Asset store* terdiri dari berbagai konten, *model*, *prefab*, *script*, *sound* dan kebutuhan lain yang diperlukan dalam pembangunan sebuah permainan. Editor Unity dapat menyimpan metadata. Editor Unity dapat diperbarui dengan sesegera mungkin seperti file yang telah dimodifikasi. Server aset Unity berjalan pada Mac, Windows, Linux dan juga berjalan pada PostgreSQL, database server opensource.

Unity memiliki dua lisensi yakni Unity dan Unity Pro. Versi Unity tersedia dalam bentuk gratis, sedang versi Unity Pro didistribusikan secara berbayar. Versi Unity Pro memiliki berbagai fitur bawaan seperti efek *post processing*, *render*, efek tekstur serta berbagai fitur spesial yang tidak tersedia di versi gratis. Unity dan Unity Pro menyediakan berbagai tutorial, konten, *completed project*, wiki, dukungan melalui forum dan pembaruan ke depannya.

Fitur-fitur Unity

1. Asset Store

Diluncurkan November 2010, Unity Asset Store adalah sebuah resource yang hadir di Unity editor. Asset store terdiri dari koleksi lebih dari 4,400 asset packages, beserta 3D models, textures dan materials, sistem particle, musik dan efek suara, tutorial dan project, scripting package, editor extensions dan servis online.



Gambar 2.10 Asset Store Unity

2. Asset Tracking

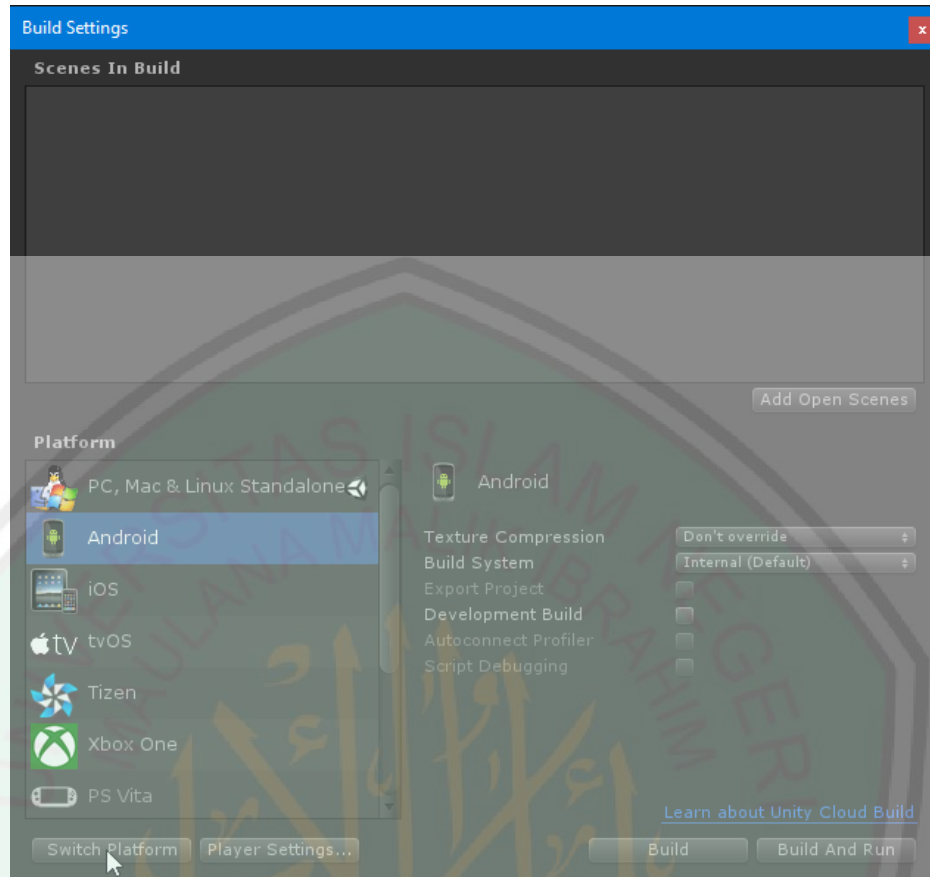
Unity juga menyertakan Server Unity Asset – sebuah solusi terkontrol untuk defeloper game asset dan script. Server tersebut menggunakan PostgreSQL sebagai backend, sistem audio dibuat menggunakan FMOD library (dengan kemampuan untuk memutar Ogg Vorbis compressed audio), video playback menggunakan Theora codec, engine daratan dan vegetasi (dimana mensupport tree billboard, Occlusion Culling dengan Umbra), built-in lightmapping dan global illumination dengan Beast, multiplayer networking menggunakan RakNet, dan navigasi mesh pencari jalur built-in.

3. Platform

Unity support pengembangan ke berbagai platform. Didalam project, developer memiliki kontrol untuk mengirim perangkat mobile, web browser, desktop, and console. Unity juga mengijinkan spesifikasi kompresi tekstur dan pengaturan resolusi di setiap platform yang didukung.

Saat ini platform yang didukung adalah BlackBerry 10, Windows 8, Windows Phone 8, Windows, Mac, Linux, Android, iOS, Unity Web Player, Adobe Flash, PlayStation 3, Xbox 360, Wii U and Wii. Meskipun tidak semua terkonfirmasi secara resmi, Unity juga mendukung PlayStation Vita yang dapat dilihat pada game Escape Plan dan Oddworld: New 'n' Tasty.

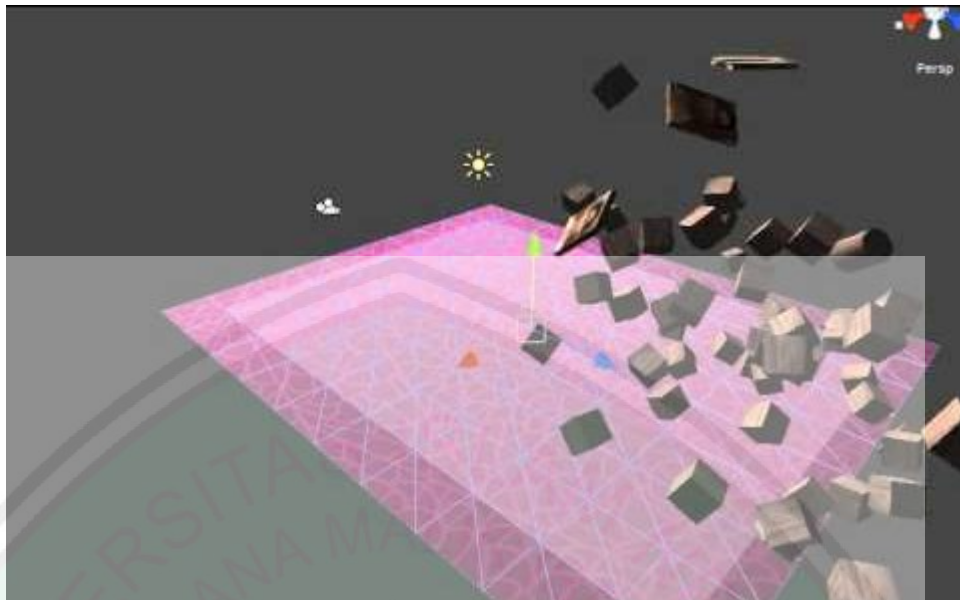
Rencana platform berikutnya adalah PlayStation 4 dan Xbox One. Dan juga rumor untuk kedepanya mengatakan HTML akan menjadi platformnya, dan plug-in Adobe baru dimana akan disubtitusikan ke Flash Player, juga akan menjadi platform berikutnya.



Gambar 2.11 Platform yang di dukung Unity

4. Physics

Unity juga memiliki suport built-in untuk PhysX physics engine (sejak Unity 3.0) dari Nvidia (sebelumnya Ageia) dengan penambahan kemampuan untuk simulasi real-time cloth pada arbitrary dan skinned meshes, thick ray cast, dan collision layers.



Gambar 2.12 Physics

5. Rendering

Graphics engine yang digunakan adalah Direct3D (Windows, Xbox 360), OpenGL (Mac, Windows, Linux, PS3), OpenGL ES (Android, iOS), dan proprietary APIs (Wii). Ada pula kemampuan untuk bump mapping, reflection mapping, parallax mapping, screen space ambient occlusion (SSAO), dynamic shadows using shadow maps, render-to-texture and full-screen post-processing effects.

Unity dapat mengambil format desain dari 3ds Max, Maya, Softimage, Blender, modo, ZBrush, Cinema 4D, Cheetah3D, Adobe Photoshop, Adobe Fireworks and Allegorithmic Substance. Asset tersebut dapat ditambahkan ke game project dan diatur melalui graphical user interface Unity.

ShaderLab adalah bahasa yang digunakan untuk shaders, dimana mampu memberikan deklaratif “programming” dari fixed-function pipeline dan program shader ditulis dalam GLSL atau Cg. Sebuah shader dapat menyertakan banyak

varian dan sebuah spesifikasi fallback declarative, dimana membuat Unity dapat mendeteksi berbagai macam video card terbaik saat ini, dan jika tidak ada yang kompatibel, maka akan dilempar menggunakan shader alternatif yang mungkin dapat menurunkan fitur dan performa.

Pada 3 Agustus 2013, seiring dengan diluncurkannya versi 4.2, Unity mengizinkan developer indie menggunakan Realtime shadows hanya untuk Directional lights, dan juga menambahkan kemampuan dari DirectX11 yang memberikan shadows dengan resolusi pixel yang lebih sempurna, tekstur untuk membuat objek 3d dari grayscale dengan lebih grafik facial, animasi yang lebih halus dan mempercepat FPS.

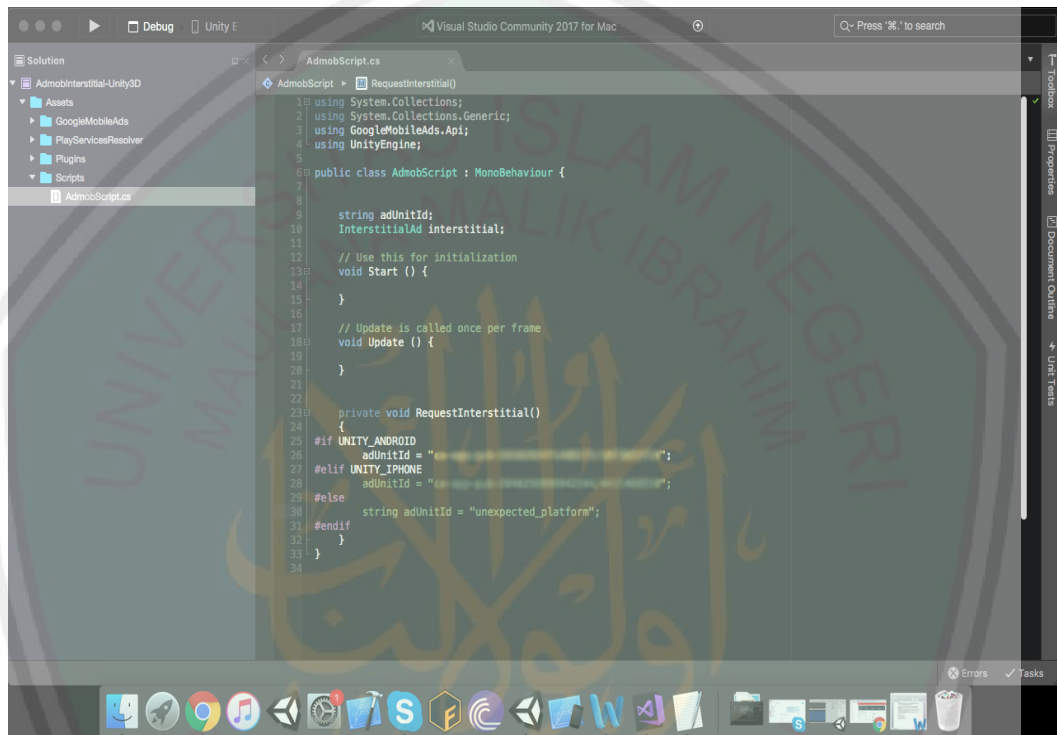


Gambar 2.13 Rendering

6. Scripting

Script game engine dibuat dengan Mono 2.6, sebuah implementasi open-source dari .NET Framework. Programmer dapat menggunakan UnityScript

(bahasa terkustomisasi yang terinspirasi dari syntax ECMAScript, dalam bentuk JavaScript), C#, atau Boo (terinspirasi dari syntax bahasa pemrograman python). Dimulai dengan dirilisnya versi 3.0, Unity menyertakan versi MonoDevelop yang terkustomisasi untuk debug script.



Gambar 2.14 Visual Studio 2017

2.3 Penelitian Terkait

Penelitian yang memiliki keterkaitan dengan penelitian ini antara lain :

1. Penelitian yang dilakukan oleh Ni Ketut Dewi Ari Jayanti (2014), penggunaan Algoritma Floyd Warshall dalam Masalah Jalur Terpendek pada Penentuan Tata Letak Parkir. Algoritma Floyd Warshall merupakan algoritma salah satu pencarian yang dapat digunakan dalam menghitung jalur terpendek dan mampu membandingkan semua kemungkinan lintasan pada graph untuk setiap sisi dari semua simpul yang ada. Floyd Warshall digunakan untuk melakukan perhitungan jalur terpendek pada letak parkir dengan menghitung jarak seluruh jalur/lintasan yang ada antara blok parker.
2. Y. Rudi Kriswanto dkk (2014), Penentuan Jarak Terpendek Rute Transmisi dengan Algoritma Floyd Warshall. Dalam penelitian ini, peneliti bertujuan mengembangkan perangkat lunak aplikasi untuk menentukan jarak terdekat yang dapat dilalui penumpang. Peneliti ini menggunakan algoritma Floyd Warshall untuk menghitung jarak terdekat antar dua titik. Model proses pengembangan perangkat lunak yang digunakan adalah model waterfall.
3. Penelitian yang dilakukan Iqbal Said (2015), Perancangan Simulasi Jalur Terpendek Kota Wisata Medan menggunakan Algoritma Floyd Warshall. Pencarian jalur terpendek merupakan suatu permasalahan untuk menentukan sebuah jalur yang mempunyai dua simpul atau node yaitu node awal dan node akhir atau tujuan dengan jumlah bobot minimal. Dalam proses pencarian rute terpendek dari satu lokasi ke lokasi lain tentu kita membutuhkan suatu peta pembantu, dan dalam kasus pencarian rute terpendek pada sebuah node akan

merepresentasikan suatu lokasi pada peta tersebut, dan suatu bobot akan merepresentasikan.



BAB III

DESAIN DAN PERANCANGAN SISTEM

3.1 Alur Cerita (*Storyline*)

Game ini adalah *game* yang dimainkan secara *single player*. Pada *game* ini terdapat karakter *player* sebagai pemain utama yang akan dijalankan oleh pengguna. *Game* ini lebih bersifat bermain sambil belajar dan dapat memberikan informasi tentang jalur tercepat menuju lokasi yang diinginkan.

Game ini menceritakan tentang seorang wisatawan yang memutuskan untuk jalan-jalan menuju alun-alun Batu. Pada *game* ini pemain harus mengikuti NPC untuk memudahkan wisatawan (pemain) untuk menemukan jalan terpendek menuju alun-alun Batu.

3.2 Game Play

- Menu : *play, info, quit*
- Area *Game* (permainan) : jalan menuju batu
- Kontrol *Game* (permainan) : *game* dikontrol dengan keyboard.
- Misi selesai : misi dinyatakan selesai jika wisatawan (pemain) sudah sampai di tempat tujuan

3.3 Konten Permainan



- Latar Permainan (*Game*)

Latar pada *game* ini adalah area perkotaan yang berupa jalan dan bangunan dll.

- **Karakter**

Terdapat dua karakter utama yaitu *Non Playable Character (NPC)* dan *player*, rincian karakter di bawah ini :

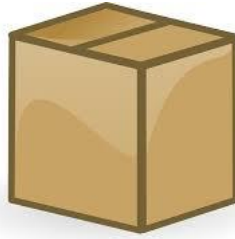
Tabel 3.1 Karakter

Nama	Gambar Karakter	Jenis	Kemampuan
Mobil NPC		NPC (Non Player Character)	Menunjukkan jalur terpendek menuju alun-alun Batu
Player (wisatawan)		Wisatawan	Mengikuti NPC agar mendapat jalur terpendek menuju alun-alun Batu.

- **Item**

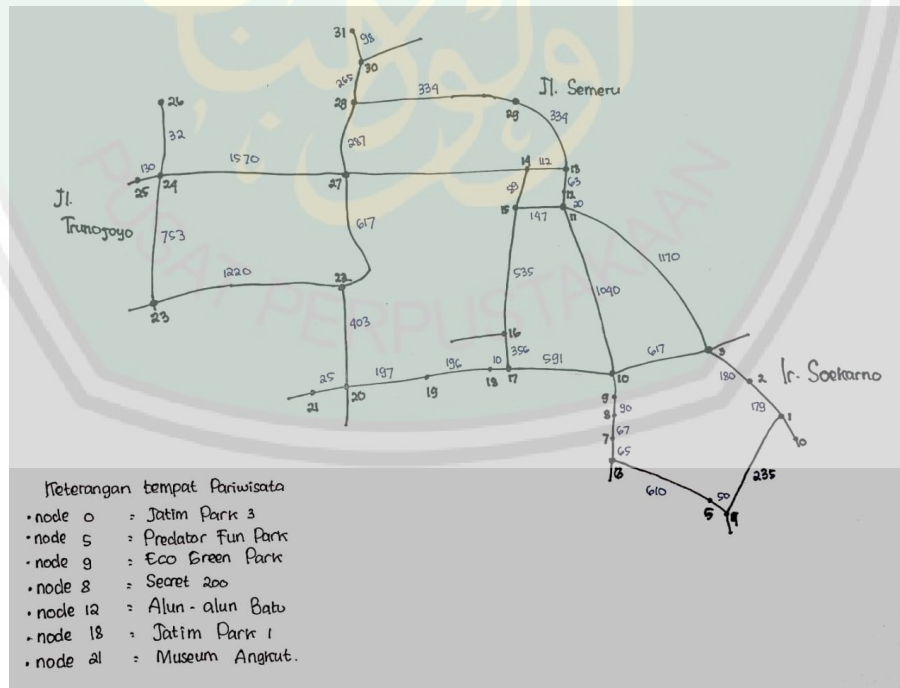
- a. *Box* (kotak)

Jika NPC melewati kotak tersebut akan muncul informasi jarak yang telah ditempuh.



Gambar 3.1 Box

- b. Rancangan Peta
- c. Peta yang dibuat dalam *game* ini dibangun menyerupai kondisi yang berada di Malang kota dan Batu. Peta untuk *game* ini dibuat berbeda-beda. Peta yang berdasar oada kondisi jalan di Malang yang terdapat banyak sekali jalan alternatif menuju Batu. Jalan tersebut mempunyai berbagai tembusan menuju Batu.



Gambar 3.2 Peta

3.4 Layout Aplikasi

Berikut desain aplikasi game :

a. Splash Screen

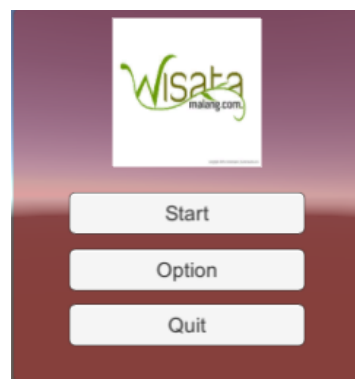
Splash screen adalah untuk menampilkan program pertama sebelum masuk ke menu utama.



Gambar 3.3 splash screen

b. Main Menu

Main menu game berisi button Start, Option dan Quit. Button start berfungsi untuk memulai game, button option digunakan untuk memberikan petunjuk cara bermain game kepada player, dan button terakhir adalah quit digunakan untuk keluar dari game.



Gambar 3.4 Main menu

c. Menu Option

Untuk menu option berfungsi untuk memberikan petunjuk kepada player agar memahami alur cerita dari game. Petunjuk inilah yang memberitahu kepada player apa yang harus player lakukan pada game ini.



Gambar 3.5 Menu Option

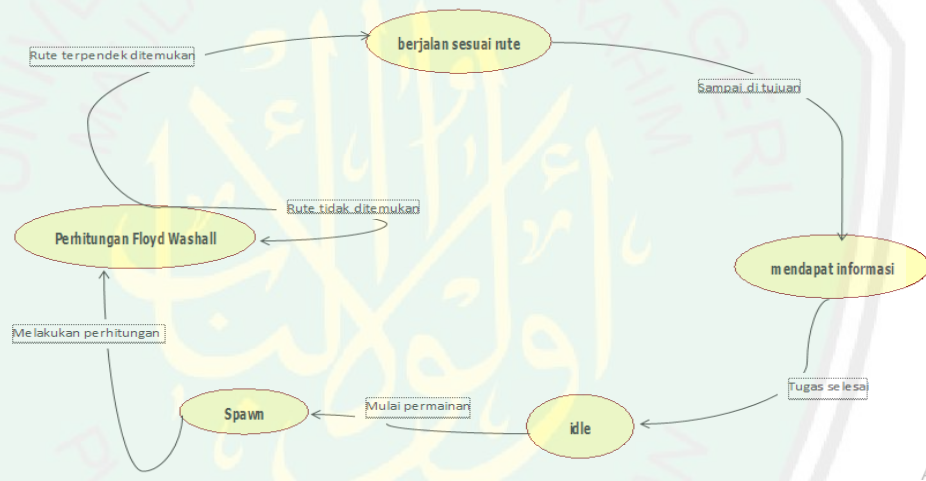
d. Latar Permainan

Pada latar permainan ini seorang player melaksanakan misinya untuk menuju objek wisata.



Gambar 3.6 Latar Permainan

3.5 Perancangan Finite State Machine



Act

Gambar 3.7 Finite State Machine

Implementasi FSM di *game* ini adalah untuk mengatur perilaku NPC. Sedangkan perilaku karakter utama mengikuti perintah orang yang bermain *game*.

3.6 Sumber Data

Data yang diperoleh dalam penelitian ini ada dua sumber data yaitu:

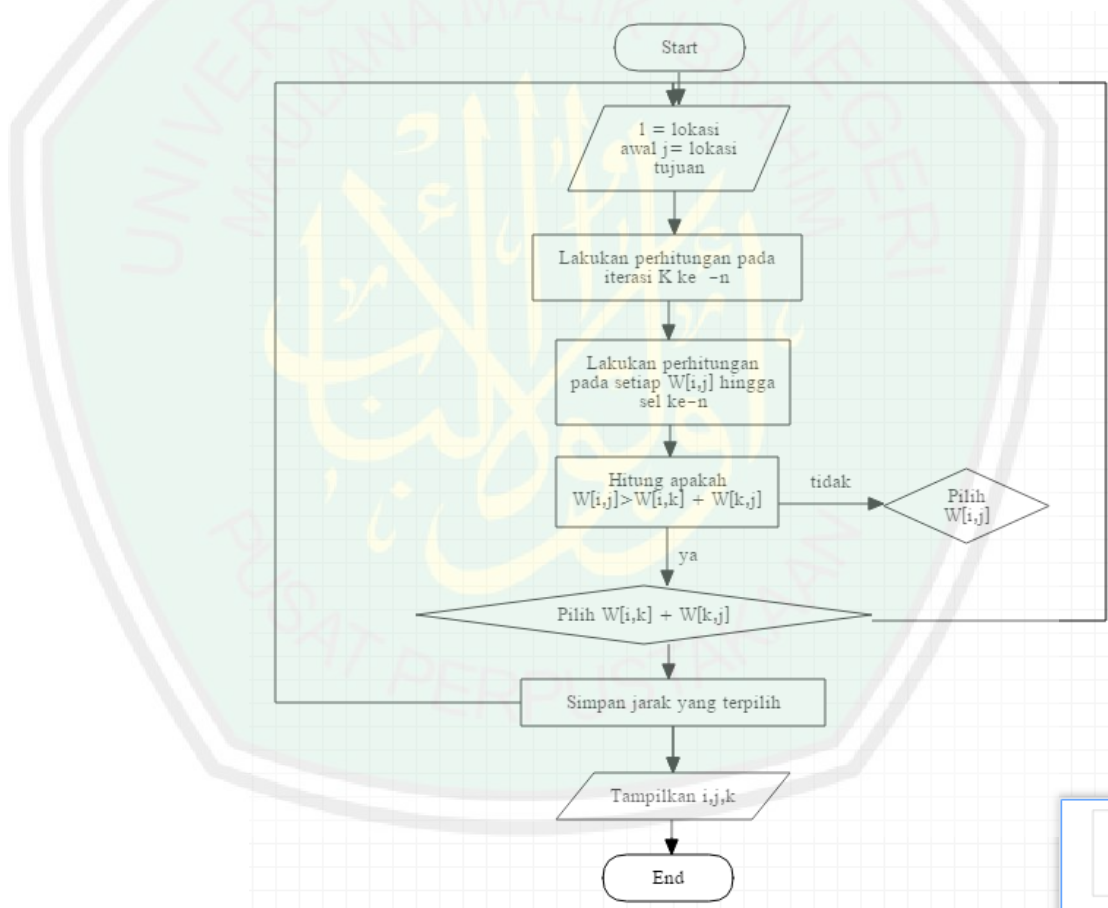
1. Data Sekunder

Saya melakukan pencarian, pembelajaran dari berbagai macam literatur dan dokumen yang menunjang pengerjaan Tugas Akhir ini khususnya yang berkaitan dengan kepadatan kendaraan di jalan.

2. Data Primer

Melakukan pengamatan terhadap data yang diteliti dan melakukan survey lapangan dengan survey pada tiga titik masuk kendaraan.

3.7 Flowchart



Gambar 3.8 Flowchart Algoritma Floyd Warshall

Pada gambar 3.11 ini menjelaskan alur pencarian rute terpendek dengan Algoritma Floyd Warshall untuk pencarian rutenya. Berikut keterangan dari *Flowchart* pencarian rute terpendek berdasarkan gambar diatas:

1. Proses pertama yang dilakukan adalah dengan menginputkan lokasi asal dan lokasi tujuan sebagai data awal.
2. Pada iterasi 1, setiap sel matriks dilakukan pengecekan apakah jarak antar dua titik mula-mula lebih besar dari penjumlahan antar titik asal ke titik tujuan (tcekan apakah jarak antar dua titik mula-mula lebih besar dari penjumlahan antar titik asal ke titik tujuan (titik tujuan = iterasi ke- 1) dengan jarak titik asal(titik asal = iterasi ke- 1) ke titik tujuan. Dengan kata lain apakah $W[i,j] > w[i,k] + W[k,j]$.
3. Jika iya maka jarak antar dua titik mula-mula diganti dengan penjumlahan antar jarak asal ke titik tujuan (titik tujuan= iterasi ke- 1) dengan jarak titik asal (titik asal= iterasi ke- 1) ke titik tujuan $W[i,k] + W[k,j]$.
4. Jika tidak, maka jarak yang digunakan yaitu jarak antar titik mula-mula $W[i,j]$.
5. Selanjutnya Proses iterasi dilakukan hingga iterasi akhir (jumlah iterasi = jumlah total titik).

3.8 Proses Perhitungan Algoritma Floyd Warshall

Algoritma *Floyd Warshall* dalam permainan ini akan diterapkan ke dalam gerakan musuh mengejar *player*. Perancangan Algoritma *Floyd Warshall* dapat dijelaskan melalui tahapan sebagai berikut:

Algoritma *Floyd Warshall* memiliki input graf berarah dan berbobot (V,E) , yang berupa himpunan titik (titik V) dan himpunan sisi (sisi E). Bobot sisi e dapat diberi simbol $d\{i,j\}$.

Misalkan W_0 adalah matriks keterhubungan graf berarah berbobot mula-mula. W^* adalah matriks keterhubungan minimal dengan $W_{i,j}$ = lintasan terpendek dari titik v_i ke v_j . Algoritma *Floyd Warshall* untuk mencari lintasan terpendek adalah sebagai berikut:

1. $W = W_0$
2. Untuk $k = 1$ hingga n , lakukan :
 Untuk $i = 1$ hingga n , lakukan :
 Untuk $j = 1$ hingga n , lakukan :
 Jika $W_{[i,j]} > W_{[i,k]} + W_{[k,j]}$ maka tukar $W_{[i,j]}$ dengan $W_{[i,k]} + W_{[k,j]}$.
3. $W^* = W$

Dalam iterasinya untuk mencari lintasan terpendek, algoritma *Floyd Warshall* membentuk n matriks, sesuai dengan iterasi- k . Ini akan menyebabkan prosesnya lambat, terutama untuk nilai n yang besar. Meskipun waktu prosesnya bukanlah yang tercepat, algoritma *Floyd Warshall* di atas hanya menghitung jarak terpendek dari semua titik ke semua titik, tetapi tidak menjelaskan bagaimana *path* terpendeknya, Untuk menentukan *path* yang menghasilkan jarak terpendek, maka harus ditambahkan matriks bujur sangkar Z (ukuran $n \times n$) yang disusun sebagai berikut:

$$\text{Inisialisasi } Z^{(0)}_{i,j} = \begin{cases} j & \text{jika } W_{i,j}^0 \neq \infty \\ 0 & \text{jika } W_{i,j}^0 = \infty \end{cases}$$

Dalam iterasi ke-k, apabila titik v_k disisipkan antara titik- i dan titik-j (berarti menukar $W_{i,j}$ dengan $W_{i,k} + W_{k,j}$, maka ganti $Z_{i,j}$ dengan $Z_{i,k}$. Agar lebih efisien, penggantian matriks Z dilakukan bersama-sama dengan iterasi pencarian jarak terpendeknya.

Revisi algoritma *Floyd Warshall* dengan melibatkan *path* terpendeknya adalah sebagai berikut :

1. $W = W_0; Z = Z_0$
2. Untuk $k = 1$ hingga n , lakukan :
 - Untuk $i = 1$ hingga n , lakukan :
 - Untuk $j = 1$ hingga n , lakukan :
 - Jika $W_{[i,j]} > W_{[i,k]} + W_{[k,j]}$ maka
 - a. tukar $W_{[i,j]}$ dengan $W_{[i,k]} + W_{[k,j]}$.
 - b. Ganti $Z_{i,j}$ dengan $Z_{i,k}$
3. $W^* = W$

- 1) $W = W_0$
- 2) Untuk $k = 1$ hingga n , lakukan :
 - Untuk $i = 1$ hingga n , lakukan :
 - Untuk $j = 1$ hingga n , lakukan :

Karena $W [1,2]$ tidak $> W[1,1] + W[1,2]$, maka $W[1,2]$ tidak diubah.

$$2. \quad d(2,4) = \infty \min \{d(2,1), d(1,4)\} = 3+4=7$$

Karena $W [2,4] > W[2,1] + W[1,4]$, maka $W[2,4]$ diubah menjadi 7. Ini berarti bahwa ada lintasan dari v_2 ke v_4 melalui v_1 yang mempunyai bobot lebih kecil (yaitu lintasan $v_2v_1v_4$ dengan jumlah bobot 7) dibandingkan dengan lintasan dari v_2 ke v_4 secara langsung (bobot = ∞ karena tidak ada lintasan dari v_2 ke v_4 secara langsung). Dengan cara yang sama, harga $W[i,j]$ di hitung untuk setiap i dan j .

Tabel 3.3 D (18)

i/j	V1	V2	V3	V4	V5	V6	V7	V8	V9	V10	V11	V12	V13	V14	V15	V16	V17	V18
V1	0	0,235	0,359	0,845	0,976	1,529	1,567	1,676	1,724	2,211	3,004	2,601	2,536	2,271	1,612	3,554	1,984	3,857
V2	0,235	0	0,594	0,61	0,875	1,764	1,466	1,911	1,959	2,446	3,239	2,836	2,771	2,506	1,847	3,789	2,219	4,092
V3	0,359	0,594	0	0,882	0,617	1,17	1,208	1,317	1,365	1,852	2,645	2,242	2,177	1,912	1,253	3,195	1,625	3,498
V4	0,845	0,61	0,882	0	0,265	1,305	0,856	1,452	1,5	1,987	2,78	2,377	2,312	2,047	1,388	3,33	1,76	3,633
V5	0,976	0,875	0,671	0,265	0	1,04	0,591	1,187	1,235	1,722	2,515	2,112	2,047	1,782	1,123	3,065	1,495	3,368
V6	1,529	1,764	1,17	1,305	1,04	0	1,631	0,147	0,195	0,682	1,475	1,072	1,007	0,742	0,083	2,025	0,455	2,328
V7	1,567	1,446	1,208	0,856	0,591	1,631	0	1,772	1,683	2,307	0,403	0,806	1,975	1,71	1,714	2,993	1,423	3,296
V8	1,676	1,911	1,317	1,452	1,187	0,147	1,772	0	0,089	0,535	1,369	0,966	0,901	0,636	0,201	1,919	0,349	2,222
V9	1,724	1,959	1,365	1,5	1,235	0,195	1,683	0,089	0	0,624	1,28	0,877	0,812	0,547	0,112	1,83	0,26	2,133
V10	2,211	2,446	1,852	1,987	1,722	0,682	2,307	0,535	0,624	0	1,904	1,501	1,436	1,171	0,736	2,454	0,884	2,757
V11	3,004	3,239	2,645	2,78	2,515	1,475	0,403	1,369	1,28	1,904	0	0,403	2,325	1,307	1,392	2,376	1,02	1,623
V12	2,601	2,836	2,242	2,377	2,112	1,072	0,806	0,966	0,877	1,501	0,403	0	1,922	0,904	0,989	1,973	0,617	1,22
V13	2,536	2,771	2,177	2,312	2,047	1,007	1,975	0,901	0,812	1,436	2,325	1,922	0	0,265	0,924	2,875	0,552	3,178
V14	2,271	2,506	1,912	2,047	1,782	0,742	1,71	0,636	0,547	1,171	1,307	0,904	0,265	0	0,659	1,857	0,287	2,16
V15	1,612	1,847	1,253	1,388	1,123	0,083	1,714	0,201	0,112	0,736	1,392	0,989	0,924	0,659	0	1,942	0,372	2,245
V16	3,554	3,789	3,195	3,33	3,065	2,025	2,993	1,919	1,83	2,454	2,376	1,973	2,875	1,857	1,942	0	1,57	0,753
V17	1,984	2,219	1,625	1,76	1,495	0,455	1,423	0,349	0,26	0,884	1,02	0,617	0,552	0,287	0,372	1,57	0	1,873
V18	3,857	4,092	3,498	3,633	3,368	2,328	3,296	2,222	2,133	2,757	1,623	1,22	3,178	2,16	2,245	0,753	1,873	0

BAB IV HASIL DAN PEMBAHASAN

4.1 Peralatan yang digunakan

Berikut keterangan spesifikasi *hardware* dan *software* yang digunakan sebelum diimplementasikan, yaitu:

4.1.1 Kebutuhan Perangkat Keras

Perangkat keras atau *hardware* yang digunakan untuk pembuatan klasifikasi dan pengujian metode tersebut adalah sebagai berikut:

- a. Laptop Asus A456U : Processor intel Core i5 RAM 2GB

Tabel 4.1 Kebutuhan Perangkat Keras

No.	Perangkat Keras	Spesifikasi
1.	Processor	Core i516Ghz
2.	RAM	2 Gb
3.	HDD	1 Tb
4.	Monitor	14'
5.	Speaker	On
6.	Mouse dan Keyboard	On

4.1.2 Kebutuhan Perangkat Lunak

Adapun *software* atau perangkat lunak yang digunakan dalam pembuatan klasifikasi dan pengujian metode yaitu:

Tabel 4.2 Kebutuhan Perangkat Lunak

No.	Perangkat Keras	Spesifikasi
1.	Sistem Operasi	Windows 10 64 Bit
2.	<i>GameEngine</i>	Unity3d 2018
3.	<i>ScriptWriter</i>	Visual 2017

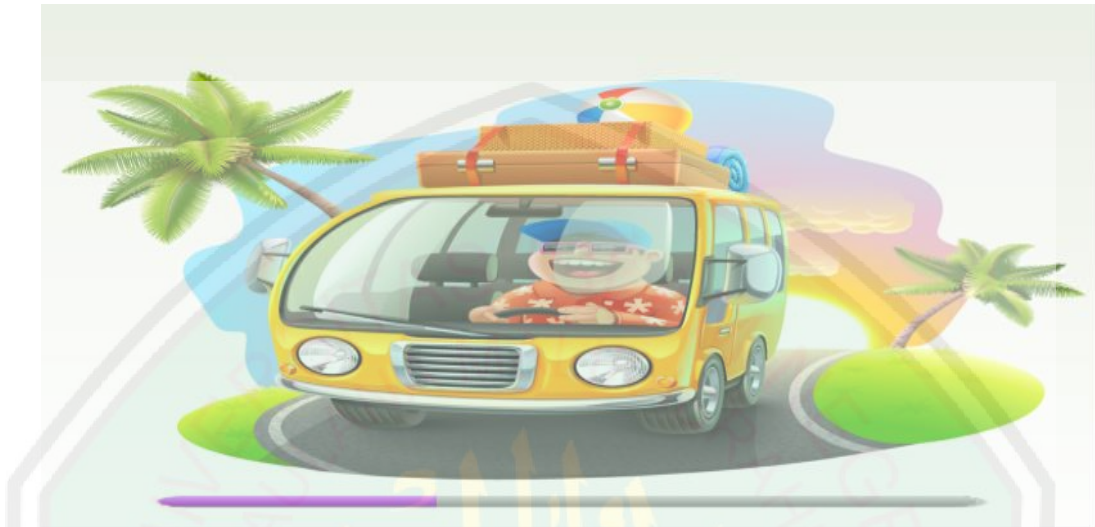
4.1.3 Implementasi Aplikasi *Game*

Berikut ini adalah tampilan game yang telah selesai dibuat :



Gambar 4.1 Tampilan Splash Screen

Gambar 4.1 adalah gambar tampilan splash screen. Tampilan splash screen merupakan tampilan yang akan muncul sebelum berpindah menuju loading.



Gambar 4.2 Tampilan Loading

Gambar 4.2 adalah gambar tampilan loading. Tampilan loading merupakan tampilan yang akan muncul sebelum berpindah menuju menu utama. Loading ini bertujuan untuk memberi jeda waktu sebelum simulasi berjalan. Loading akan muncul sekitar 5 detik.



Gambar 4.3 Tampilan Menu utama

Gambar 4.3 adalah tampilan menu awal merupakan tampilan yang pertama kali akan muncul ketika aplikasi simulasi dijalankan.



Gambar 4.4 Tampilan Utama

Gambar 4.4 merupakan tampilan utama Stage game adalah halaman permainan pada game, yang berisi player, *Non Player Character* (NPC) dan terrain permainan yang akan dimainkan dalam game yang telah dibuat. Dalam stage ini pemain mengikuti *non player character* (npc) agar player dapat menemukan jalur terpendek menuju tempat wisata yaitu alun-alun batu.

4.2 Pengujian Algoritma Floyd Warshall

4.2.1 Pengujian Algoritma *Floyd Warshall*



Gambar 4.5 Jarak antar node

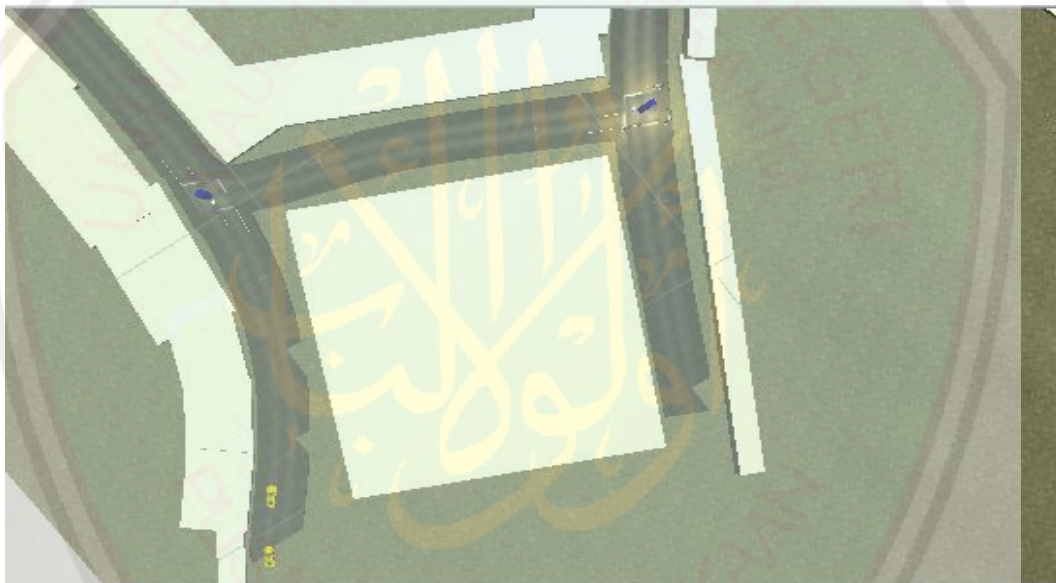


Gambar 4.6 Jarak node yang digambarkan dengan bentuk cube

Pengujian ini dilakukan untuk melihat proses bagaimana pencarian jarak terpendek menuju ke tempat wisata Kota Batu. Pengujian ini dilakukan pada *non*

player character (npc) yang berada titik koordinat yang telah ditentukan. Setiap node memiliki jarak berbeda. Pengujian ini dilakukan oleh 1 *player* dan 1 *non player character* (npc).

Dengan Algoritma *Floyd Warshall* diharapkan npc (*non player character*) dapat mencari jarak terdekat menuju tempat wisata Kota Batu. Pengujian ini dilakukan untuk mengecek apakah algoritma ini telah berfungsi sehingga dapat membuat *non player character* (npc) menuju tempat wisata yang di tuju.



Gambar 4.7 Posisi awal NPC dan player



Gambar 4.8 NPC menuju target

Dengan Algoritma *Floyd Warshall Non Player Character* (NPC) diharapkan dapat bergerak menuju target (tempat wisata Kota Batu) dengan memilih jarak yang terpendek terhadap target (tempat wisata Kota Batu). Pengujian ini dilakukan untuk mengecek Algoritma *Floyd Warshall* telah berfungsi sehingga dapat membuat *Floyd Warshall Non Player Character* (NPC) bergerak menuju target (tempat wisata Kota Batu) seperti yang diharapkan. Dan terbukti Algoritma *Floyd Warshall* dapat diterapkan untuk menentukan jarak terdekat terhadap target (tempat wisata Kota Batu).

Algoritma *Floyd Warshall* dalam penelitian ini akan di tuliskan dalam bahasa C# Unity dengan source dan comman berikut.

Source Code	Command
<pre>public NavMeshAgent nmAgent; public GameObject[] destinasi; public int i; public int p;</pre>	Deklarasi variabel

<pre> public int r; public int[] routes; public int[] routes3; public int[] path; public int route; public int ptouch; public System.Diagnostics.Stopwatch timer; public bool once = true; </pre>	
<pre> void Start () { int[,] weights = { { 1, 2, 74}, { 2, 1, 74}, { 2, 3, 179}, { 3, 2, 179}, { 3, 4, 180}, { 4, 3, 180}, {5, 2, 235 }, {2, 5, 235 }, { 5, 6, 50}, { 6, 5, 50}, { 6, 7, 560}, { 7, 6, 560}, { 7, 8, 65}, { 8, 7, 65}, { 8, 9, 67}, { 9, 8, 67}, { 9, 10, 90}, { 10, 9, 90}, { 10, 11, 43}, { 11, 10, 43}, { 4, 11, 617}, { 11, 4, 617}, { 4, 12, 1170}, { 12, 4, 1170}, { 12, 13, 20}, {13, 12, 20}, {13, 14, 63}, { 14, 13, 63}, { 14, 15, 112}, { 15, 14, 112}, { 11, 12, 1040}, { 12, 11, 1040}, { 15, 16, 89}, { 16, 15, 89}, { 16, 17, 535}, { 17, 16, 535}, { 17, 18, 356}, { 18, 17, 359}, { 18, 11, 591}, { 11, 18, 591}, { 18, 19, 10}, { 19, 18, 10}, { 19, 20, 196}, { 20, 19, 196}, { 20, 21, 197}, { 21, 20, 197}, { 21, 22, 25}, { 22, 21, 25}, { 21, 23, 403}, { 23, 21, 403}, { 24, 23, 1220}, { 23, 24, 1220}, { 24, 25, 753}, { 25, 24, 753}, { 25, 26, 130}, { 26, 25, 130}, { 27, 25, 32}, { 25, 27, 32}, { </pre>	<p>Array jarak antar node</p>

<pre>25, 28, 1570}, { 28, 25, 1570}, { 28, 15, 260}, { 15, 28, 260}, { 28, 29, 287}, { 29, 28, 287}, { 29, 30, 334}, { 30, 29, 334}, { 30, 14, 334}, { 14, 30, 334}, { 29, 31, 287}, { 31, 29, 287}, { 31, 32, 98}, { 32, 31, 98} };</pre>	
<pre>next = new int[numVerticies, numVerticies]; for (int i = 0; i < numVerticies; i++) { for (int j = 0; j < numVerticies; j++) { if (i != j) { next[i, j] = j + 1; } } }</pre>	<p>Untuk menyimpan jalur dalam matriks.</p>
<pre>for (int k = 0; k < numVerticies; k++) { for (int i = 0; i < numVerticies; i++) { for (int j = 0; j < numVerticies; j++) { if (dist[i, k] + dist[k, j] < dist[i, j]) {</pre>	<p>Perhitungan <i>Floyd Warshall</i>.</p>

<pre> dist[i, j] = dist[i, k] + dist[k, j]; next[i, j] = next[i, k]; } } } } } </pre>	
<pre> public int[] GetPath(int awal, int tujuan) { int[] path = new int[100]; for (int i = 0; i < next.GetLength(0); i++) { for (int j = 0; j < next.GetLength(1); j++) { if (awal == i && tujuan == j) { int p = 0; int u = i+1; int v = j+1; path[0] = awal; do { p++; u = next[u - 1,v -1]; </pre>	Mencari jalur terpendek

```
        path[p] = u;
    } while (u != v);
    }
}
return path;
}

public int[] GetPath3(int[] path)
{
    List<int> listPath = new List<int>();
    for (int i = 0; i < path.GetLength(0); i++)
    {
        if (i + 1 != path.GetLength(0))
        {
            int awal = path[i];
            int tujuan = path[i + 1];
            int[] temp = GetPath(awal, tujuan);
            for (int f = 0; f < temp.GetLength(0);
f++)
            {
                listPath.Add(temp[f]);
            }
        }
    }
    int[] r = new int[100];
```

<pre> foreach (int t in listPath) { Debug.Log(t); } int[] tes = listPath.ToArray(); return tes; } </pre>	
--	--

Tabel 4.3 Keterangan Implementasi *Floyd Warshall*

Data yang ada 3 skenario yang tiap skenario berisikan 20 kali percobaan.

Pengujian pertama diuji dari Jl. Trunojoyo (Pujon) – Museum Angkut – Jatim Park 1 – Secret Zoo berhasil mendapatkan jalur terdekat dengan menempuh node (25-24-23-22-20-21) - (21-20-19-18) – (18-17-10-9-8) jarak yang ditempuh 3707 meter pada game.

Tabel 4.4 Pengujian di Jl. Trunojoyo menuju 3 tempat wisata

No	Awal	Tujuan	Manual		Floyd		Keterangan
			Jalur	Jarak Tempuh	Jalur	Jarak Tempuh	
1	Jl. Trunojoyo	Museum Angkut	25,24,27,22,20,21	2745	25,24,27,22,20,21	2745	sesuai
			25,24,23,22,20,21	2531	25,24,23,22,20,21	2531	sesuai
			25,24,27,28,29,13,14,15,16,17,18,19,20,21	4175	25,24,27,28,29,13,14,15,16,17,18,19,20,21	4175	sesuai
			25,24,27,28,29,13,12,11,15,16,17,18,19,20,21	4204	25,24,27,28,29,13,12,11,15,16,17,18,19,20,21	4204	sesuai
			25,24,27,28,29,13,12,11,10,17,18,19,20,21	4562	25,24,27,28,29,13,12,11,10,17,18,19,20,21	4562	sesuai
			25,24,27,28,29,13,12,11,3,10,17,18,19,20,21	5309	25,24,27,28,29,13,12,11,3,10,17,18,19,20,21	5309	sesuai
			25,24,23,22,27,28,29,13,14,15,16,17,18,19,20,21	5195	25,24,23,22,27,28,29,13,14,15	3876	tidak sesuai
			25,24,23,22,27,28,29,13,12,11,10,17,18,19,20,21	5807	25,24,23,22,27,28,29,13,12,11,10,17,18,19,20,21	5807	sesuai
			25,24,23,22,27,28,29,13,12,11,15,16,17,18,19,20,21	5224	25,24,23,22,27,28,29,13,12,11,15,16,17,18,19,20,21	5224	sesuai

25,24,23,22,27,28,29,13,14,15,11,10,17,18,19,20,21	6082	25,24,23,22,27,28,29,13,14,15,11,10,17,18,19,20,21	6082	sesuai
25,24,23,22,27,14,15,16,17,18,19,20,21	4388	25,24,23,22,27,14,15,16,17,18,19,20,21	4388	sesuai
25,24,23,22,27,14,13,12,11,15,16,17,18,19,20,21	4641	25,24,23,22,27,14,13,12,11,15,16,17,18,19,20,21	4641	sesuai
25,24,23,22,27,14,13,12,11,15,16,17,18,19,20,21	4641	25,24,23,22,27,14,13,12,11,15,16,17,18,19,20,21	4641	sesuai
25,24,23,22,27,14,13,12,11,10,17,18,19,20,21	5234	25,24,23,22,27,14,13,12,11,10,17,18,19,20,21	5234	sesuai
25,24,23,22,27,14,15,11,10,17,18,19,20,21	5265	25,24,23,22,27,14,15,11,10,17,18,19,20,21	5265	sesuai
25,24,23,22,27,14,15,11,3,10,17,18,19,20,21	6012	25,24,23,22,27,14,15,11,3,10,17,18,19,20,21	6012	sesuai
25,24,23,22,27,14,13,12,11,3,10,17,18,19,20,21	5981	25,24,23,22,27,14,13,12,11,3,10,17,18,19,20,21	5981	sesuai
25,24,27,14,13,12,11,3,10,17,18,19,20,21	4961	25,24,27,14,13,12,11,3,10,17,18,19,20,21	4961	sesuai
25,24,27,14,15,11,3,10,17,18,19,20,21	5002	25,24,27,14,15,11,3,10,17,18,19,20,21	5002	sesuai
25,24,27,28,29,13,12,11,3,10,17,18,19,20,21	4797	25,24,27,28,29,13,12,11,3,10,17,18,19,20,21	4797	sesuai

2	Museum Angkut	Jatim Park 1	21,20,19,18	418	21,20,19,18	418	sesuai
			21,20,22,27,14,15,16,17,18	2492	21,20,22,27,14,15,16,17,18	2492	sesuai
			21,20,22,27,14,13,12,11,15,16,17,18	2485	21,20,22,27,14,13,12,11,15,16,17,18	2485	sesuai
			21,20,22,27,14,13,12,11,10,17,18	3078	21,20,22,27,14,13,12,11,10,17,18	3078	sesuai
			21,20,22,27,14,13,12,11,3,10,17,18	3825	21,20,22,27,14,13,12,11,3,10,17,18	3825	sesuai
			21,20,22,27,14,15,11,10,17,18	3182	21,20,22,27,14,15,11,10,17,18	3182	sesuai
			21,20,22,27,14,15,11,3,10,17,18	3929	21,20,22,27,14,15,11,3,10,17,18	3929	sesuai
			21,20,22,27,28,29,13,14,15,16,17,18	3102	21,20,22,27,28,29,13,14,15,16,17,18	3102	sesuai
			21,20,22,27,28,29,13,12,11,15,16,17,18	3131	21,20,22,27,28,29,13,12,11,15,16,17,18	3131	sesuai
			21,20,22,27,28,29,13,14,15,11,10,17,18	3989	21,20,22,27,28,29,13,14,15,11,10,17,18	3989	sesuai
		21,20,22,27,28,29,13,12,11,10,17,18	3724	21,20,22,27,28,29,13,12,	2063	tidak sesuai	

			21,20,22,27,28,29,13,14,15,11,3,10,17,18	4736	21,20,22,27,28,29,13,14,15,11,3	3518	tidak sesuai
			21,20,22,27,28,29,13,12,11,3,10,17,18	4471	21,20,22,27,28	1332	tidak sesuai
			21,20,22,23,24,27,14,15,16,17,18	5221	21,20,22,23,24,27,14,15,16,17,18	5221	ya
			21,20,22,23,24,27,14,13,12,15,16,17,18	5464	21,20,22,23,24,27,14,13,12,15,16,17,18	5464	ya
			21,20,22,23,24,27,14,13,12,11,15,16,17,18	5474	21,20,22,23,24,27,14,13,12,11,15,16,17,18	5474	ya
			21,20,22,23,24,27,14,15,11,10,17,18	6108	21,20,22,23,24,27,14,15,11,10,17,18	6108	ya
			21,20,22,23,24,27,14,13,12,11,10,17,18	6067	21,20,22,23,24,27,14,13,12,11,10,17,18	6067	ya
			21,20,22,23,24,27,14,15,11,3,10,17,18	6855	21,20,22,23,24,27,14,15,11,3,10,17,18	6855	ya
			21,20,22,23,24,27,14,13,12,11,3,10,17,18	6814	21,20,22,23,24,27,14,13,12,11,3,10,17,18	6814	ya
3	Jatim Park 1	Secret Zoo	18,17,10,9,8	758	18,17,10,9,8	758	ya

		18,17,16,15,11,10,9,8	2245	18,17	10	tidak sesuai
		18,17,16,15,11,3,10,9,8	2912	18,17,16,15,11,3,10,9,8	2912	ya
		18,17,16,14,13,12,11,10,9,8	1847	18,17,16,14,13,12,11,10,9,8	1847	ya
		18,17,16,15,14,13,12,11,3,10,9,8	2514	18,17,16,15,14,13,12,11,3,10,9,8	2514	ya
		18,19,20,22,27,14,15,16,17,10,9,8	3377	18,19,20,22,27,14,15,16,17,10,9,8	3377	ya
		18,19,20,22,27,14,15,11,10,9,8	3082	18,19,20,22,27,14,15,11,10,9,8	3082	ya
		18,19,20,22,27,14,13,12,11,10,9,8	3041	18,19,20,22,27,14,13,12,11,10,9,8	3041	ya
		18,19,20,22,27,14,13,12,11,15,16,17,10,9,8	3630	18,19,20,22,27,14,13,12,11,15,16,17,10,9,8	3630	ya

		18,19,20,22,27,28,29,13,14,15,16,17,10,9,8	4182	18,19,20,22,27,28,29,13,14,15,16,17,10,9,8	4182	ya
		18,19,20,22,27,28,29,13,12,11,15,16,17,10,9,8	4213	18,19,20,22,27,28,29,13,12,11,15,16,17,10,9,8	4213	ya
		18,19,20,22,27,28,29,13,12,11,10,9,8	3624	18,19,20,22,27,28,29,13,12,11,10,9,8	3624	ya
		18,19,20,22,27,28,29,13,12,11,3,10,9,8	4371	18,19,20,22,27,28,29,13,12,11,3,10,9,8	4371	ya
		18,19,20,22,27,28,29,14,15,11,3,10,9,8	4521	18,19,20,22,27,28,29,14,15,11,3,10,9,8	4521	ya
		18,19,20,22,23,24,27,14,15,16,17,10,9,8	6303	18,19,20,22,23,24,27,14,15,16,17,10,9,8	6303	ya
		18,19,20,22,23,24,27,14,13,12,11,15,16,17,10,9,8	6556	18,19,20,22,23,24,27,14,13,12,11,15,16,17,10,9,8	6556	ya
		18,19,20,22,23,24,27,14,13,12,11,10,9,8	5967	18,19,20,22,23,24,27,14,13,12,11,10,9,8	5967	ya
		18,19,20,22,23,24,27,14,13,15,11,10,9,8	6114	18,19,20,22,23,24	2769	tidak
		18,19,20,22,23,24,27,14,13,12,11,3,10,9,8	6714	18,19,20,22,23,24,27,14,13,12,11,3,10,9,8	6714	ya
		18,19,20,22,23,24,27,14,13,15,11,3,10,9,8	6755	18,19,20,22,23,24,27,14,13,15,11,3,10,9,8	6755	ya
total jalur terpendek	Jalan Trunojoyo - Museum Angkut - Jatim park 1 - Secret Zoo	25,24,23,22,20,21 - 21,20,19,18 - 18,17,10,9,8				3707

Pengujian kedua diuji dari Jl. Ir. Soekarno (Malang) – Eco Green Park – Predator Fun Park – Jatim Park 3 berhasil mendapatkan jalur terdekat dengan menempuh node (2-3-10-9) - (9-8-7 -5) – (5-4-1-0), dengan jarak yang ditempuh 1988 meter pada game.

Tabel 4.5 Pengujian di Jl. Ir. Soekarno menuju 3 tempat wisata

No	Awal	Tujuan	Manual		Floyd		Keterangan
			Jalur	Jarak Tempuh	Jalur	Jarak Tempuh	
1	Jl. Ir. Soekarno	Eco Green Park	2,3,10,9	887	2,3,10,9	887	sesuai
			2,3,11,10,9	2480	2,3,11,10,9	2480	sesuai
			2,3,11,12,13,14,15,16,17,10,9	3206	2,3,11,12,13,14,15,16,17,10,9	3206	sesuai
			2,3,11,15,16,17,10,9	3069	2,3,11,15,16,17,10,9	3069	sesuai
			2,3,11,12,13,14,27,22,20,19,18,17,10,9	3862	2,3,11,12,13,14,27,22,20,19,18,17,10,9	3862	sesuai

		2,3,11,15,14,27,22,20,19,18,17,10,9	3912	2,3,11,15,14,27,22,20,19,18,17,10,9	3912	sesuai
		2,3,11,12,13,29,28,27,14,15,16,17,10,9	4262	2,3,11,12,13,29,28,27	2388	tidak sesuai
		2,3,11,15,14,13,29,28,27,14,15,16,17,10,9	4536	2,3,11,15,14,13,29,28,27,14,15,16,17,10,9	4536	sesuai
		2,3,11,12,13,29,28,27,22,20,19,18,17,10,9	4445	2,3,11,12,13,29,28,27,22,20,19,18,17,10,9	4445	sesuai
		2,3,11,15,14,13,29,28,27,22,20,19,18,17,10,9	4719	2,3,11,15,14,13,29,28,27,22,20,19,18,17,10,9	4719	sesuai
		2,3,11,12,13,14,27,22,,20,19,18,17,10,9	3862	2,3,11,12,13,14,27,22,,20,19,18,17,10,9	3862	sesuai

		2,3,11,15,14,27,22,,20,19,18,17,10,9	3912	2,3,11,15,14,27,22,,20,19,18,17,10,9	3912	sesuai
		2,3,11,12,13,14,27,24,23,22,20,19,18,17,10,9	6197	2,3,11,12,13,14,27,24,23,22,20,19,18,17,10,9	6197	sesuai
		2,3,11,15,14,27,24,23,22,20,19,18,17,10,9	6247	2,3,11,15,14,27,24,23,22,20,19,18,17,10,9	6247	sesuai
		2,3,11,12,13,29,28,27,24,23,22,20,19,18,17,10,9	6780	2,3,11,12,13,29,28,27,24,23,22,20,19,18,17,10,9	6780	sesuai
		2,3,11,15,14,13,29,28,27,24,23,22,20,19,18,17,10,9	7054	2,3,11,15,14,13,29,28,27,24,23,22,20,19,18,17,10,9	7054	sesuai
		2,3,11,15,14,27,22,,20,19,18,17,10,9	3912	2,3,11,15,14,27,22,,20,19,18,17,10,9	3912	sesuai

			2,3,11,15,14,27,24,23,22,20,19,18,17,10,9	6247	2,3,11,15,14,27,24,23,22,20,19,18,17,10,9	6247	sesuai
			2,3,11,15,14,13,29,28,27,24,23,22,20,19,18,17,10,9	7054	2,3,11,15,14,13,29,28,27,24,23,22,20,19,18,17,10,9	7054	sesuai
			2,3,11,12,13,29,28,27,24,23,22,20,19,18,17,10,9	6780	2,3,11,12,13,29,28,27,24,23,22,20,19,18,17,10,9	6780	sesuai
2	Eco Green Park	Predator Fun Park	9,8,7,6,5	742	9,8,7,6,5	742	sesuai
			9,10,3,2,1,4,5	1304	9,10,3,2,1,4,5	1304	sesuai
			9,10,11,3,2,1,4,5	2897	9,10,11,3,2,1,4,5	2897	sesuai
			9,10,17,16,15,11,3,2,1,4,5	3486	9,10,17,16,15,11,3,2,1,4,5	3486	sesuai
			9,10,17,16,15,14,13,12,11,3,2,1,4,5	3623	9,10,17,16,15,14,13,12,11,3,2,1,4,5	3623	sesuai
			9,10,17,18,19,20,22,23,24,27,28,29,13,12,11,3,2,1,4,5	7835	9,10,17,18,19,20,22,23,24,27,28,29,13,12,11,3,2,1,4,5	7835	sesuai

			9,10,17,18,19,20,22,23,24,27,28,29,13,14,15,11,3,2,1,4,5	8100	9,10,17,18,19,20,22,23,24,27,28,29,13,14,15,11,3,2,1,4,5	8100	sesuai
			9,10,17,18,19,20,22,27,28,29,13,14,15,11,3,2,1,4,5	5174	9,10,17,18,19,20,22,27,28,29,13,14,15,11,3,2,1,4,5	5174	sesuai
			9,10,17,18,19,20,22,27,28,29,13,14,15,11,3,2,1,4,5	5174	9,10,17,18,19,20,22,27,28,29,13,14,15,11,3,2,1,4,5	5174	sesuai
			9,10,17,18,19,20,22,27,14,15,11,3,2,1,4,5	4367	9,10,17,18,19,20,22,27,14,15,11,3,2,1,4,5	4367	sesuai
			9,10,17,18,19,20,22,27,14,13,12,11,3,2,1,4,5	4326	9,10,17,18,19,20,22,27,14,13,12,11,3,2,1,4,5	4326	sesuai
			9,10,17,16,15,14,13,12,11,3,2,1,4,5	3623	9,10,17,16,15,14,13,12,11,3,2,1,4,5	3623	sesuai
			9,10,17,16,15,11,3,2,1,4,5	3486	9,10,17,16,15,11,3,2,1,4,5	3486	sesuai
3	Predator Fun Park	Jatim Pak 3	5,4,1,0	359	5,4,1,0	359	sesuai

			5,6,7,8,9,10,3,2,1,0	1925	5,6,7,8,9,10,3,2,1,0	1925	sesuai
			5,6,7,8,9,10,11,3,2,1,0	3518	5,6,7,8,9,10,11,3,2,1,0	3518	sesuai
			5,6,7,8,9,10,17,16,15,11,3,2,1,0	4107	5,6,7,8,9,10,17,16,15,11,3,2,1,0	4107	sesuai
			5,6,7,8,9,10,17,16,15,14,13,12,11,3,2,1,0	4244	5,6,7,8,9,10,17,16,15,14,13,12,11,3,2,1,0	4244	sesuai
			5,6,7,8,9,10,17,18,19,20,22,23,24,27,28,29,13,12,11,3,2,1,0,	8422	5,6,7,8,9,10,17,18,19,20,22,23,24,27,28,29,13,12,11,3,2,1,0,	8422	sesuai
			5,6,7,8,9,10,17,18,19,20,22,23,24,27,28,29,13,14,15,11,3,2,1,0,	8687	5,6,7,8,9,10,17,18,19,20,22,23,24,27,28,29	6736	tidak sesuai
			5,6,7,8,9,10,17,18,19,20,22,23,24,27,14,13,12,11,3,2,1,0,	7839	5,6,7,8,9,10,17,18,19,20,22,23,24,27,14,13,12,11,3,2,1,0,	7839	sesuai

		5,6,7,8,9,10,17,18,19,20,22,23,24,27,14,15,11,3,2,1,0,	7880	5,6,7,8,9,10,17,18,19,20,22,23,24,27,14,15,11,3,2,1,0,	7880	sesuai
		5,6,7,8,9,10,17,18,19,20,22,27,14,13,12,11,3,2,1,0,	4913	5,6,7,8,9,10,17,18,19,20,22,27,14,13,12,11,3,2,1,0,	4913	sesuai
		5,6,7,8,9,10,17,18,19,20,22,27,14,15,11,3,2,1,0,	4954	5,6,7,8,9,10,17,18,19,20,22,27,14,15,11,3,2,1,0,	4954	sesuai
		5,6,7,8,9,10,17,18,19,20,22,27,28,13,12,11,3,2,1,0,	5496	5,6,7,8,9,10,17,18,19,20,22,27,28,13,12,11,3,2,1,0,	5496	sesuai
		5,6,7,8,9,10,17,18,19,20,22,27,28,13,14,15,11,3,2,1,0,	5761	5,6,7,8,9,10,17,18,19,20,22,27,28,13,14,15,11,3,2,1,0,	5761	sesuai
total jalur terpendek	Jl. Ir. Soekarno - Eco Green Park - Predator Fun Park - Jatim park 3			2,3,10,9 - 9,8,7,6,5 - 5,4,1,0		1988

Pengujian ketiga diuji dari Jl. Semeru (Arah Cangar) – Alun-alun – Jatim Park 1 – Museum Angkut berhasil mendapatkan jalur terdekat dengan menempuh node (29-13-12) - (13-14-15-16-17-18) - 18-19-20-21, dengan jarak yang ditempuh 1903 pada game,

Tabel 4.6 Pengujian di Jl. Semeru menuju 3 tempat wisata

No	Awal	Tujuan	Manual		Floyd		Sesuai
			Jalur	Jarak Tempuh	Jalur	Jarak Tempuh	
1	Jalan Semeru	Alun-alun	29,13,12	417	29,13,12	417	sesuai
			29,13,14,15,11,12	702	29,13,14,15,11,12	702	sesuai
			29,28,27,14,13,12,	1056	29,28,27,14,13,12,	1056	sesuai
			29,28,27,14,15,11,12	1137	29,28,27,14,15,11,12	1137	sesuai
			29,28,27,14,15,16,17,10,11,12	3512	29,28,27,14,15,16,17,10,11,12	3512	sesuai

		29,28,27,14,15,16,17,10,3,11,12	4259	29,28,27,14,15,16,17,10,3,11,12	4259	sesuai
		29,13,14,15,16,17,10,11,12	3077	29,13,14,15,16,17	1406	idak sesua
		29,13,14,15,16,17,10,3,11,12	3207	29,13,14,15,16,17	1406	tidak sesuai
		29,13,14,15,16,17,10,9,8,7,6,5,4,1,2,3,11,12	4726	29,13,14,15,16,17,10,9,8,7,6,5,4,1,2,3,11,12	4726	Sesua i
		29,28,27,14,15,16,17,10,9,8,7,6,5,4,1,2,3,11,12	5161	29,28,27,14,15,16,17,10,9,8,7,6,5,4,1,2,3,11,12	5161	Sesua i
		29,28,27,22,20,19,18,17,16,15,11,12	3102	29,28,27,22,20,19,18,17,16,15,11,12	3102	Sesua i

		29,28,27,22,20,19,18,17,16,15,14,13,12	3199	29,28,27,22,20,19,18,17,16,15,14,13,12	3199	Sesuai
		29,28,27,24,23,22,20,19,18,17,16,15,14,13,12	6091	29,28,27,24,23,22,20,19,18,17,16,15,14,13,12	6091	Sesuai
		29,28,27,24,23,22,20,19,18,17,16,15,11,12	5994	29,28,27,24,23,22,20,19,18,17,16,15,11,12	5994	Sesuai
		29,28,27,22,20,19,18,17,10,11,12	3695	29,28,27,22,20,19,18,17,10,11,12	3695	Sesuai
		29,28,27,22,20,19,18,17,10,3,11,12	4442	29,28,27,22,20,19,18,17,10,3,11,12	4442	Sesuai
		29,28,27,24,23,22,20,19,18,17,10,11,13,12	6587	29,28,27,24,23,22,20,19,18,17,10,11,13,12	6587	Sesuai

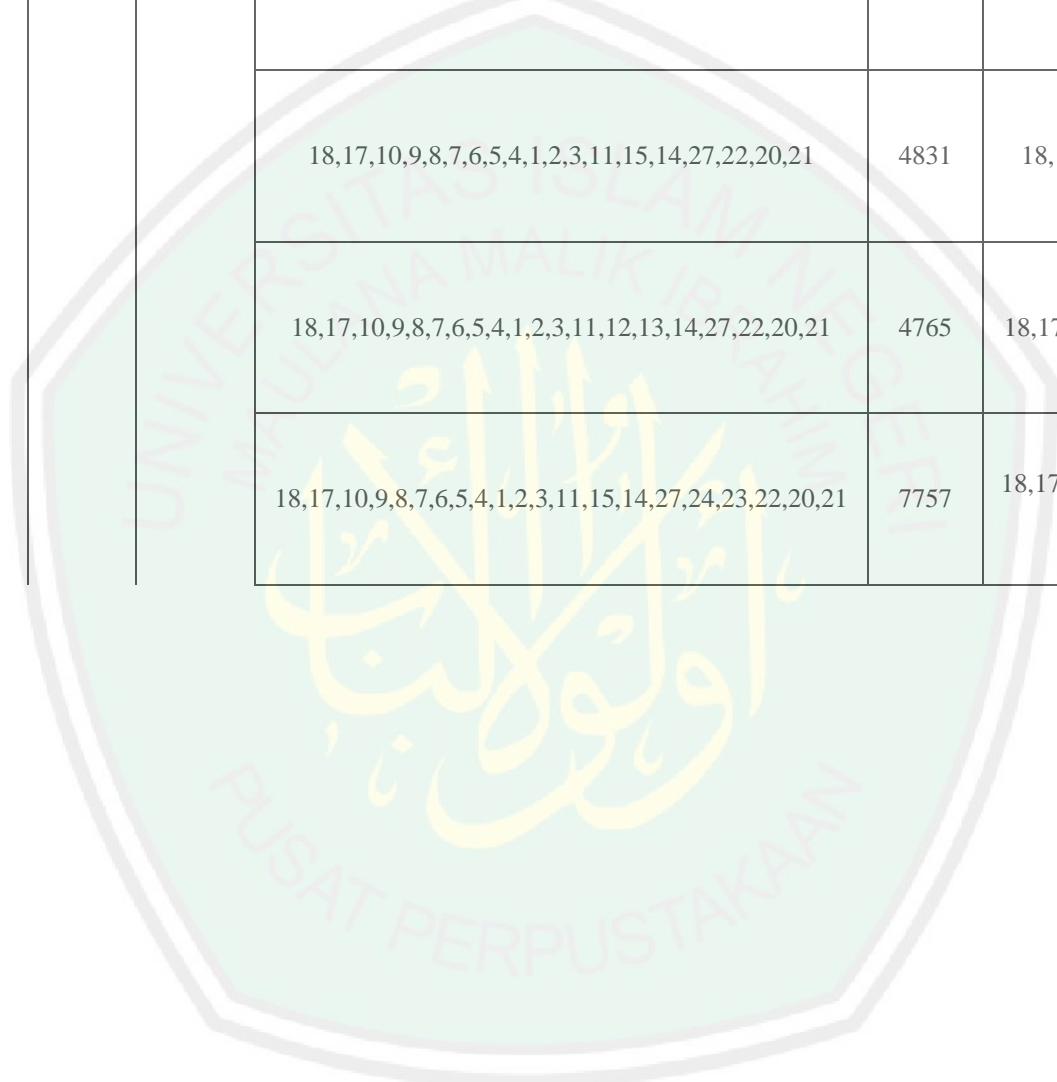
			29,28,27,24,23,22,20,19,18,17,10,3,11,12	7334	29,28,27,24,23,22,20,19,18,17,10,3,11,12	7334	Sesuai
			29,28,27,14,15,16,17,10,9,8,7,6,5,4,1,2,3,11,12	5161	29,28,27,14,15,16,17,10,9,8,7,6,5,4,1,2,3,11,12	5161	Sesuai
			29,28,27,22,20,19,18,17,16,15,11,12	3102	29,28,27,22,20,19,18,17,16,15,11,12	3102	Sesuai
2	Alun-alun	Jatim Park 1	12,11,15,16,17,18	1068	12,11,15,16,17,18	1068	Sesuai
			12,11,10,17,18	1661	12,11,10,17,18	1661	Sesuai
			12,11,3,10,17,18	2408	12,11,3,10,17,18	2408	Sesuai
			12,13,14,15,16,17,18	1165	12,13,14,15,16,17,18	1165	Sesuai
			12,13,14,27,22,20,19,18	1848	12,13,14,27,22,20,19,18	1848	Sesuai

		12,13,29,28,27,14,15,16,17,18	2268	12,13,29,28,27,14,15,16,17,18	2268	Sesuai
		12,13,29,28,27,14,15,11,10,17,18	3155	12,13,29,28,27,14,15,11,10,17,18	3155	Sesuai
		12,13,29,28,27,14,15,11,3,10,17,18	3902	12,13,29,28,27,14,15,11,3,10,17,18	3902	Sesuai
		12,13,29,28,27,24,23,22,20,19,18	5357	12,13,29,28,27,24,23,22,20,19,18	5357	Sesuai
		12,13,29,28,27,14,15,11,3,2,1,4,5,6,7,8,9,10,17,18	4804	12,13,29,28,27,14,15,11,3,2,1	2864	tidak sesuai

		12,11,3,2,1,4,5,6,7,8,9,10,17,18	3310	12,11,3,2,1,4,5,6,7,8,9,10,17,18	3310	sesuai
		12,13,14,27,24,23,22,20,19,18	4662	12,13,14,27,24,23	2646	tidak sesuai
		12,13,29,28,27,22,20,19,18	2431	12,13,29,28,27,22,20,19,18	2431	Sesuai
		12,11,15,14,27,22,20,19,18	1929	12,11,15,14,27,22,20,19,18	1929	Sesuai
		12,11,15,14,27,24,23,22,20,19,18,	4855	12,11,15,14,27,24,23,22,20,19,18,	4855	Sesuai
		12,13,14,15,11,10,17,18	1838	12,13,14,15,11,10,17,18	1838	Sesuai
		12,13,29,28,27,24,23,22,20,19,18	3902	12,13,29,28,27,24,23,22,20,19,18	3902	Sesuai

			12,13,29,28,27,22,20,19,18	2431	12,13,29,28,27,22,20,19,18	2431	Sesuai
			12,13,14,27,22,20,19,18	1838	12,13,14,27,22,20,19,18	1838	Sesuai
3	Jatim Park 1	Museum Angkut	18,19,20,21	418	18,19,20,21	418	sesuai
			18,17,16,15,14,27,22,20,21	2492	18,17,16,15,14,27,22,20,21	2492	sesuai
			18,17,16,15,11,12,13,14,27,22,20,21	2485	18,17,16,15,11,12,13,14,27,22,20,21	2485	sesuai

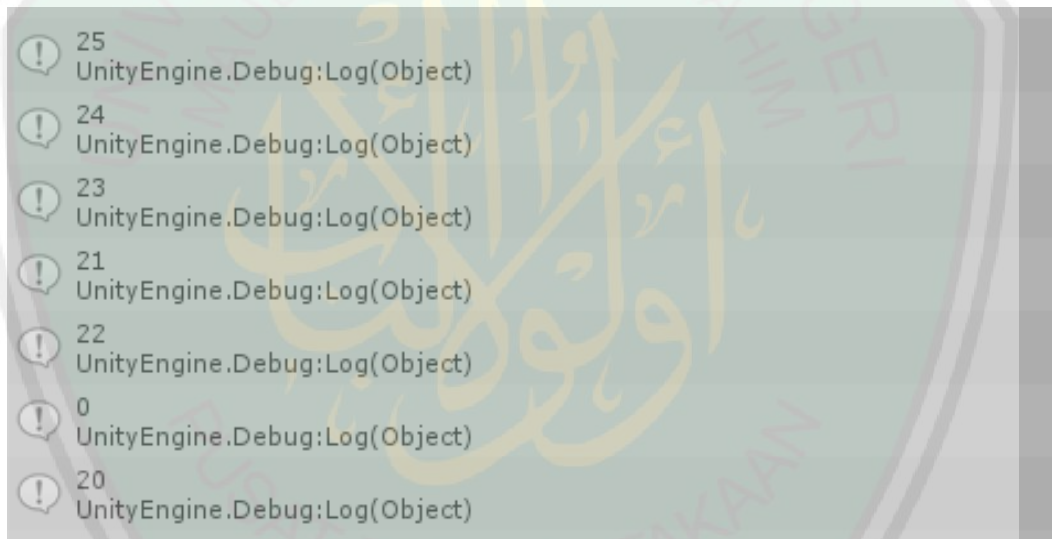
18,17,10,11,12,13,14,27,22,20,21	3078	18,17,10,11,12,13,14,27,22,20,21	3078	sesuai
18,17,10,3,11,12,13,14,27,22,20,21	3825	18,17,10,3,11,12,13,14,27,22,20,21	3825	sesuai
18,17,10,9,8,7,6,5,4,1,2,3,11,15,14,27,22,20,21	4831	18,17,10,9,8,7,6,5,4,1,2,3,11,15,14,27,22,20,21	4831	sesuai
18,17,10,9,8,7,6,5,4,1,2,3,11,12,13,14,27,22,20,21	4765	18,17,10,9,8,7,6,5,4,1,2,3,11,12,13,14,27,22,20,21	4765	sesuai
18,17,10,9,8,7,6,5,4,1,2,3,11,15,14,27,24,23,22,20,21	7757	18,17,10,9,8,7,6,5,4,1,2,3,11,15,14,27,24,23,22,20,21	7757	sesuai



		18,17,10,9,8,7,6,5,4,1,2,3,11,12,13,14,27,24,23,22,20,21	7716	18,17,10,9,8,7,6,5,4	1526	tidak sesuai
		18,17,10,9,8,7,6,5,4,1,2,3,11,12,13,29,28,27,24,23,22,21	8299	18,17,10,9,8,7,6,5,4,1,2,3,11,12,13,29	4041	tidak sesuai
		18,17,10,9,8,7,6,5,4,1,2,3,11,15,14,13,29,28,27,24,23,22,21	8564	18,17,10,9,8,7,6,5,4,1,2,3,11,15,14,13,29,28,27,	4593	tidak sesuai
		18,17,10,9,8,7,6,5,4,1,2,3,11,15,14,27,22,20,21	4821	18,17,10,9,8,7,6,5,4,1,2,3,11,15,14,27,22,20,21	4821	sesuai
total jalur terpendek	Jalan Semeru - Alun-alun - Jatim Park 1 - Museum Angkut			29,13,12 - 12,11,15,16,17,18 - 18,19,20,21		1903

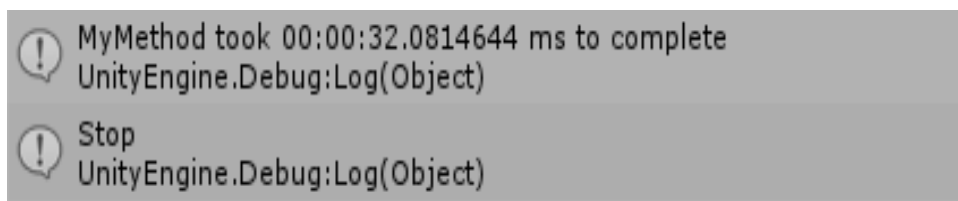
Dari Hasil uji coba algoritma *Floyd-warshall* pada *table* 4.4, tabel 4.5 dan tabel 4.6 dapat disimpulkan pada *game* pencarian rute terpendek di kota wisata Batu yang diterapkan pada NPC, memiliki tingkat kesuksesan sebesar 94.44%

Uji coba algoritma *Floyd Warsall* dilakukan untuk melihat hasil dari implementasi algoritma tersebut dalam *game* yang telah dibuat menggunakan *Unity*. Gambar 4.9 adalah tampilan *tab console* pada *Unity* yang menampilkan hasil-hasil dari perhitungan *Floyd Warshall* dan pada Gambar 4.10 adalah tampilan *tab console* pada *Unity* yang menampilkan *Timer/Stopwatch*.



Gambar 4.9 Rute Terpendek yang berhasil ditemukan pada *Console*

Unity



Gambar 4.10 *Timer/Stopwatch* pada *Console Unity*

4.3 Integrasi Dalam Islam

Islam datang untuk merubah banyak pemahaman keliru yang dibawa oleh akal manusia yang pendek, kemudian mengaitkan dengan nilai-nilai dan akhlak yang mulia. Wisata dalam pemahaman sebagian umat terdahulu dikaitkan dengan upaya menyiksa diri dan mengharuskannya untuk berjalan di muka bumi, serta membuat badan letih sebagai hukuman baginya atau zuhud dalam dunianya. Islam datang untuk menghapuskan pemahaman negatif yang berlawanan dengan (makna) wisata. Allah berfirman :

قَدْ خَلَتْ مِنْ قَبْلِكُمْ سُنَنٌ فَسِيرُوا فِي الْأَرْضِ فَانظُرُوا كَيْفَ كَانَ
عِقَابُ الْمُكَذِّبِينَ ﴿١٣٧﴾

“Sesungguhnya telah berlalu sebelum kamu sunnah-sunnah Allah swt, karena itu berjalanlah kamu di muka bumi dan perhatikanlah bagaimana akibat orang-orang yang mendustakan (rasul-rasul).” (QS. Al-Imran:137)

سُنَنٌ: plural untuk sunnah, yaitu jalan yang diteladani, contohnya: sunnah nabi. Maksudnya adalah kejadian-kejadian yang menimpa pendusta.

Dalam ayat-ayat sebelumnya Allah swt menganjurkan para hamba-Nya untuk bersabar dan bertaqwa, juga mengingatkan orang-orang mukmin tentang bantuan Allah swt berupa „kiriman“ malaikat kepada mereka dalam Perang Badar. Dalam ayat-ayat selanjutnya Allah swt menyuruh orang beriman untuk bersegera memperoleh ridha Allah swt. Kemudian Allah swt juga menerangkan secara terperinci kekalahan orang beriman dalam Perang Uhud setelah mereka memenangi Perang Badar, serta penyebab kekalahan itu, yang tidak lain dikarenakan ada sebagian orang membangkang perintah Rasulullah saw.

Kemudian Allah swt menjelaskan bahwa cobaan adalah hal yang biasa dalam hidup ini, dan terbunuhnya para nabi jangan sampai menjadikan hati orang-orang beriman menjadi lemah. Allah swt juga menjelaskan pelajaran dan hikmah yang diambil dari kekalahan dalam Perang Uhud.

Menurut Al-Maraghi, perintah berjalan di muka bumi dan merenungkan berbagai peristiwa yang telah menimpa umat sebelumnya. Menjadikan hal tersebut sebagai pelajaran, agar mendapatkan ilmu yang benar serta didasari oleh bukti. Dari situlah akan didapatkan sebuah petunjuk, bahwa pengaturan antara kebenaran dengan kebatilan memang telah terjadi pada umat terdahulu. Kemenangan selalu berada di pihak yang benar selama dibarengi dengan kesabaran dan ketaqwaan. Termasuk di antara ketaqwaan adalah mengikuti yang telah digariskan oleh Allah swt untuk mencapai kemenangan yaitu melatih diri, mempersiapkan peralatan senjata guna memerangi musuh, seperti yang telah diperintahkan oleh Allah swt melalui firman-Nya:

وَأَعِدُّوا لَهُمْ مَا اسْتَطَعْتُمْ مِنْ قُوَّةٍ وَمِنْ رِبَاطِ الْخَيْلِ
تُرْهِبُونَ بِهِ عَدُوَّ اللَّهِ وَعَدُوَّكُمْ وَءَاخِرِينَ مِنْ دُونِهِمْ لَا
نَعْلَمُونَ لَهُمُ اللَّهُ يَعْلَمُهُمْ وَمَا تُنْفِقُوا مِنْ شَيْءٍ فِي سَبِيلِ اللَّهِ يُوَفَّ
إِلَيْكُمْ وَأَنْتُمْ لَا تُظْلَمُونَ

“Dan siapkanlah untuk menghadapi mereka kekuatan apa saja yang kamu sanggupi dan dari kuda-kuda yang ditambat untuk berperang (yang dengan persiapan itu) kamu menggentarkan musuh Allah swt dan musuhmu dan orang-orang selain mereka yang kamu tidak mengetahuinya; sedang Allah swt

mengetahuinya. apa saja yang kamu nafkahkan pada jalan Allah swt niscaya akan dibalasi dengan cukup kepadamu dan kamu tidak akan dianiaya (dirugikan).”

Hal itu terus berlangsung dalam bentuk tatanan yang tetap dan sebab sebab yang tidak berubah dan berganti. Berjalan di muka bumi untuk menyelidiki keadaan orang-orang terdahulu guna mengetahui apa yang telah menimpa mereka, merupakan sebuah alat bantu yang paling baik untuk mengetahui sunnah dan mengambil pelajaran darinya. Terkadang bisa diambil juga pelajaran seperti ini dari buku-buku sejarah yang telah dicatat oleh orang-orang yang telah menyelidiki ikhwal mereka. Mereka ini telah melihat langsung bekas-bekas peninggalannya, sehingga bisa dibuat pelajaran dan petuah untuk generasi umat beragama.

Tetapi mengacu kepada buku-buku sejarah itu masih berada di bawah tingkat keyakinan orang-orang yang melakukan ekspedisi tersendiri, kemudian melihat peninggalan-peninggalan bersejarah secara langsung. Ibn Katsir dalam kitab tafsirnya menyebutkan bahwa, Allah swt berfirman kepada hamba-hamba-Nya yang mukmin tatkala mereka mendapat musibah dalam Perang Uhud dan gugur tujuh puluh orang diantara mereka sebagai syuhada. Bahwa hal yang serupa itu telah terjadi pada umat-umat terdahulu sebelum mereka, para pengikut nabi-nabi yang akhirnya merekalah yang beruntung dan orang-orang kafirlah yang binasa. Karenanya Allah swt memerintahkan hamba-hamba-Nya mengadakan perjalanan untuk melihat dan menyaksikan bagaimana akibat dari yang diderita oleh umat-umat yang mendustakan nabi-nabi-Nya.

Sayyid Quthb juga menjelaskan bahwa Alquran mengaitkan masa lalu manusia dan masa kininya, serta masa kininya dengan masa lalunya. Maka, dari celah-celah semua itu di isyaratkannya pula masa depannya. Ketika bangsa Arab mendapatkan firman ini pertama kali, kehidupan, pengetahuan dan pengalamam mereka (Sebelum Islam) tidak menolerir pandangan yang menyeluruh ini bagi mereka. Kalau bukan karena Islam (dan kitab sucinya Alquran) yang dengannya Allah swt menjadikan mereka lain dari yang lain, dan menjadikan bagian dari mereka sebagai umat yang memimpin dunia, niscaya mereka tetap dalam kejahiliyaan.

Sistem kabilah (kesukuan) yang mereka hidup di bawah bayang-bayang, tidak dapat membimbing pikiran mereka untuk mengaitkan penduduk Jazirah Arab dengan apa yang berlaku dalam kehidupan mereka. Apalagi, mengaitkan antara penduduk bumi ini dengan segala peristiwa yang dialaminya dan menghubungkan antara peristiwa-peristiwa yang terjadi di dunia dengan *sunnah kauniyah* (hukum alam) yang berlaku dalam semua aspek kehidupan. Ini merupakan lompatan jauh yang tidak bersumber dari lingkungan dan tidak terjadi sebagai tuntutan kehidupan pada masa itu. Tetapi, yang membawa mereka kepada pandangan yang demikian adalah akidah. Akidah inilah yang membawa mereka dan mengangkat derajat mereka sedemikian tinggi dalam kurun waktu seperempat abad saja. Sementara itu, manusia lain yang katanya lebih modernis tidak dapat mencapai tingkat pemikiran yang demikian tinggi melainkan setelah memakan waktu berabad-abad dan mereka tidak mengetahui kebakuan sunnah serta undang-undang alam ini melainkan setelah berlalu masa beberapa generasi. Namun setelah mereka mengetahui bakunya sunnatullah ini, mereka lupa bahwa

keberlakuan hukum alam ini disertai oleh kehendak Ilahi yang mutlak dan bahwa segala sesuatu akan kembali kepada Allah swt.

Adapun umat pilihan ini, maka mereka telah meyakini semua itu, *tasawwur*-nya (pandangannya) begitu luas. Mereka merasakan keseimbangan antara sunnatullah dan sesudah itu, merasa tenanglah hatinya terhadap kehendak Allah swt yang mutlak.

Sunnahlah yang mengatur kehidupan. Sunnah yang telah ditetapkan oleh kehendak yang mutlak. Maka apa yang terjadi pada masa sebelum kamu akan terjadi pula (dengan kehendak Allah swt) pada masa kalian. Keadaan yang terjadi pada umat sebelum kalian, juga akan terjadi pada kalian.

Itulah akibat yang dapat disaksikan bekas-bekasnya di muka bumi dan dapat disaksikan pula langkah-langkah perjalanan mereka yang ditapaktilasi oleh generasi sesudahnya. Alquran banyak menyebutkan perjalanan dan bekas-bekas mereka ini dalam berbagai tempat didalamnya. Sebagian dibatasi tempatnya, masanya dan pelaku-pelakunya. Sedangkan sebagian lagi hanya diisyaratkan tanpa batasan dan perincian. Disini Alquran mengemukakan isyarat secara global untuk tujuan secara global pula, yaitu apa yang terjadi terhadap orang-orang yang mendustakan rasul- rasul (ayat-ayat) Allah swt kemarin, juga akan terjadi pada para pendusta itu sekarang dan pada waktu yang akan datang.

Hal itu dimaksudkan agar hati kaum muslimin menjadi tenang terhadap akibat ini dari satu sisi dan dari sisi lain agar mereka berhati-hati serta tidak tergelincir seperti orang-orang yang mendustakan ayat-ayat Allah swt itu. Jadi, disana terdapat seruan untuk menenteramkan hati dan untuk berhati-hati.

Bumi itu seluruhnya adalah satu dan merupakan panggung kehidupan

manusia. Bumi dan kehidupan di dalamnya adalah buku terbuka yang dapat dibaca oleh mata kepala dan mata hati.

Ali Ash Shobuni juga menjelaskan bahwasanya telah berlalu sunnah-sunnah Allah swt kepada umat-umat yang terdahulu dengan kehancuran dan kebinasaan, akibat perselisihan mereka terhadap para nabinya. Perintah untuk mencari berbagai berita dan informasi orang-orang yang mendustakan rasul-rasul-Nya dan apa yang ditimpakan kepada mereka, bertujuan agar umat manusia dapat mengambil pelajaran dari jejak-jejak mereka.



BAB V PENUTUP

5.1. Kesimpulan

Berdasarkan hasil dari implementasi dan pengujian yang dilakukan peneliti maka dapat ditarik kesimpulan bahwa :

1. Dalam penelitian ini dilakukan proses uji coba sebanyak 3 skenario yang masing-masing satu skenario di uji coba dihasilkan 94.44% sukses dari ketepatan dan kecepatan dalam menentukan jarak terpendek.

5.2 Saran

Penulis sadar, dalam pembuatan *game* ini masih banyak kekurangan yang nantinya perlu untuk dilakukan pengembangan, diantaranya :

1. Perbaiki *asset* yang lebih bagus baik dari *asset* 2D maupun 3D.
2. Penambahan tempat tujuan wisata, pada game pencarian rute terpendek wisata kota Batu ini masih ada 3 tempat tujuan wisata.
3. Perbaiki tampilan sangat diperlukan
4. Game ini hanya dapat dioperasikan pada Desktop, alangkah baiknya jika dikembangkan ke berbagai platform.

DAFTAR PUSTAKA

- A, Yoeti, Oka. 1980, *Pemasaran Pariwisata*, Penerbit Angkasa, Bandung
- Al-Qur'anul Karim dan Terjemahannya. 2003. Semarang: Toha Putera
- Arif, Yunifa Miftachul. 2010. "*Strategi Menyerang pada Game FPS Menggunakan Hierarchy Finite State Machine dan Logika Fuzzy*". Thesis. Surabaya: Pasca Sarjana Teknik Elektro ITS.
- Eriq M. Adams J, Satria A. Wicaksono.2012. *Rancang Bangun Web Pencarian Rute Terpendek Antar Gedung di Kampus Menggunakan Algoritma Floyd Warshall*. Program Teknologi Informasi dan Ilmu Komputer Universitas Brawijaya.
- <https://www.pricebook.co.id/article/review/2016/01/26/3593/ada-4-jenis-dan-11-genre-game-yang-mana-favorit-kamu>, diakses pada tanggal 29 Maret 2019 pukul 21.27
- Muktiyah Kumala, Soelistyo Aris , Ida Nuraini (2017). *ANALISIS POTENSI SEKTOR PARIWISATA SEBAGAI SEKTOR UNGGULAN DI WILAYAH JAWA TIMUR*. Ilmu Ekonomi UMM Malang.
- Kristanto Harris (2015), *Skripsi tentang Perangkat Lunak Seeking Path dengan algoritma All-Pairs Shorts-Path Floyd Warshall*. Program Studi Teknik Informatika Sekolah Tinggi Manajemen Informatika Dan Komputer STMIK TIME MEDAN.
- Mustofa, Saiful, *Strategi Pembelajaran Mufradat*, UIN Malang, 2010
- Nendya, Matahari Bhakti. Gunanto, Samuel Gandang & Santosa, R, Gunawan. 2012. "*Pemetaan Perilaku Non-Playable Character Pada Permainan Berbasis Role Playing Game Menggunakan Metode Finite State Machine*". Yogyakarta: Fakultas Teknologi Informasi, Universitas Kristen Duta Wacana dan Fakultas Seni Media Rekam, Institut Seni Indonesia.

- Ni Ketut Dewi Ari Jayanti (2014), *Penggunaan Algoritma Floyd Warshall dalam Masalah Jalur Terpendek pada Penentuan Tata Letak Parkir*. STMIK STIKOM BALI. Seminar Nasional Informatika.
- Nur, H.E. & Setiawan, A. 2013. *Program Dinamis untuk Penentuan Lintasan Terpendek dengan Pendekatan Algoritma Floyd-Warshall*. *Dinamika Teknik*. VII(1): 17-25.
- Said Iqbal,(2015), *Perancangan Simulasi Jalur Terpendek Kota Wisata Medan menggunakan Algoritma FloydWarshall*. Program Studi Teknik Informatika STMIK Budi Darma Medan.
- Siang, Jong Jek (2011), *Riset Operasi Dalam Pendekatan Algoritmis*, Penerbit Andi, Yogyakarta.
- Sutojo, T., Mulyanto, E., & Suhartono, V., 2011, *Kecerdasan Buatan*, Penerbit Andi, Yogyakarta.
- Y. Rudi Kriswanto dkk (2014), *Penentuan Jarak Terpendek Rute Transmisi dengan Algoritma FloydWarshall*. Seminar Nasional Teknologi Informasi & Komunikasi Terapan 2014(SEMATIK 2014) ISBN : 979-26-0276-3 15 November, Semarang.
- Yunifa, Fahrul, Fressy. 2011. “*Desain Perubahan Perilaku pada NPC Game Menggunakan Logika Fuzzy*”. Malang : Seminar on electrical, informatic, and ITS education.

