BAB 3

ANALISIS SISTEM DAN PERANCANGAN

3.1. Analisis Sistem

Pada bagian analisis sistem merupakan penelitian atas sistem yang telah ada dengan tujuan untuk merancang sistem yang baru atau diperbarui. Analisis sistem ini mencakup analisis-analisis terhadap masalah yang akan dilakukan terhadap game mulai dari analisis masalah, analisis game sejenis, analisis kebutuhan non-fungsional, analisis pengguna, analisis algoritma, analisis kebutuhan fungsional hingga perancangan antarmuka dari game yang akan dibuat.

3.1.1. Analisis Masalah

Berdasarkan tinjauan analisis sistem yang telah dilakukan dengan melakukan evaluasi terhadap game yang akan dibangun. Beberapa permasalahan yang merupakan hasil dari analisis yang dilakukan dijadikan sebagai latar belakang dalam pembangunan game Legenda Gatotkaca. Berikut permasalahan dari hasil analisis yang dilakukan yaitu Banyak remaja yang mengenal karakter Gatotkaca, akan tetapi tidak mengetahui cerita hidup Gatotkaca.

3.1.2. Analisis Game Sejenis

Analisis game sejenis dilakukan untuk memberikan gambaran sistem agar lebih mudah dalam pembangunan dan sebagai pembanding untuk kesempurnaan game yang akan dicapai.

3.1.2.1. Abimanyu Sang Ksatria Mahabrata

1. Pengenalan

Game ini mengambil latar belakang kisah pewayangan Jawa, Wiracarita Mahabrata. Setelah mengalami beberapa perubahan pada cerita aslinya, akhirnya terbentuklah cerita versi ASKA dengan sosok Abimanyu sebagai tokoh utama. Penyajian game ini sudah dikemas sedemikian rupa supaya menarik dan tidak terlihat kolot, namun tetap mempertahankan unsur budaya.



Gambar 3.1 Game Abimanyu Sang Ksatria Mahabrata

2. Storyline

Game ini menceritakan tentang persiapan sebelum perang baratayudha yang akan menggemparkan. Abimanyu yang sudah tumbuh dewasa diperintahkan untuk menemui Gatotkaca di negara petir untuk mengetahui seberapa kuat ilmu yang sudah didapatkan abimanyu, apakah sudah siap untuk menghadapi perang baratayudha. Dalam perjalanan menuju negara petir abimanyu dihadapkan pada beberapa musuh yang menghalangi, dan abimanyu harus mengalahkan musuh-musuh tersebut untuk menemui dengan gatotkaca di negara petir.

Di level pertama Abimanyu melawan musuh yang telah disediakan dan wajib mengumpulkan kunci sebagai syarat masuk ke level berikutnya, di level berikutnya musuh semakin kuat bisa menembak dan tetap wajib mengumpulkan kunci untuk masuk ke level berikutnya.

Pada level selanjutnya pemain akan melawan boss pertama, pemain bisa menggunakan senjata berupa panah dengan cara mengambil item panah terlebih dahulu, setelah mengalahkan boss pertama pemain masuk ke level berikutnya dan melawan boss kedua dan selanjutnya setelah mengalahkan boss kedua, pemain akan melawan boss yang ketiga, dan setelah berhasil mengalahkan boss ketiga maka permainan selesai.

3. Gameplay

Tugas utama pemain dalam memainkan aplikasi ini yaitu melalui level-level, dimana setiap bagian level menyediakan petualangan dengan menampilkan beberapa karakter level tersebut maupun karakter musuh yang harus dilawan, sehingga pemain memperoleh *score* (nilai). Dalam game Abimanyu Sang Ksatria Mahabrata, pemain memainkan game sebagai *single-player* yang dapat menjelajahi permainan dengan menggerakkan karakter pemain (*player character*) ke arah kanan, kiri, atas atau bawah, dan spasi untuk mengalahkan lawan dengan menggunakan senjata berupa pedang.

Saat memulai game, pemain dapat berinteraksi dengan game pada halaman menu utama yang diantaranya :

1. Memilih menu Main untuk memulai permainan dari awal.
2. Memilih menu Bantuan untuk melihat penjelasan kontrol dan item dalam *game*.
3. Memilih menu Skor Tertinggi untuk melihat nilai tertinggi yang sudah ada
4. Memilih menu Hapus Skor untuk menghapus nilai
5. Memilih menu Info Game untuk melihat informasi game Abimanyu Sang Ksatria Mahabrata.
6. Memilih menu WWW untuk masuk ke website pembuat game
7. Memilih menu Exit yang digunakan untuk keluar dari permainan atau mengakhiri game.

Saat melakukan perjalanan, pemain dapat menemukan beberapa item yang disediakan diantaranya diamond untuk menambah nilai, kendi untuk menambahkan darah pemain, dan wajib mengumpulkan semua item kunci sebanyak 3 buah sebagai syarat untuk masuk ke level berikutnya. Dan pada pertarungan akhir melawan boss terdapat item panah yang dapat digunakan sebagai senjata untuk melawan boss.

Kelebihan Game Abimanyu Sang Ksatria Mahabrata

1. Gameplay cukup mudah untuk dimainkan

2. Antarmuka cukup sederhana sehingga memudahkan pemain baru

Kekurangan Game Abimanyu Sang Ksatria Mahabrata

1. Jalur musuh statis (single line)
2. Storyline tidak sesuai dengan cerita aslinya.

Tabel 3.1 Perbandingan Dengan Game Yang Akan Dibangun

Komponen	Game Abimanyu Sang Ksatria Mahabrata	Game yang akan dibangun
Genre	Adventure	Beat ‘em up, Labirin, Shoot ‘em up
Storyline	Tidak mengambil cerita abimanyu	Mengambil cerita Gatotkaca
Grafis	Grafis 2D	Grafis 2D
Jumlah pemain	*Single player*	*Single player*
Karakter	Abimanyu	Jabang Tetuka, Gatotkaca
Musik	Gamelan	Gamelan

3.1.3. Analisis Game yang Akan Dibangun

Game yang akan dibangun berjudul Legenda Gatotkaca merupakan game yang multi genre, genre yang diterapkan dalam game ini yaitu *beat ‘em up,* labirin dan *shoot ‘em up*.

3.1.3.1. Storyline

Game legenda gatotkaca ini menceritakan sepenggal kisah hidup Gatotkaca, Gatotkaca kecil/Tetuka (selanjutnya disebut Tetuka) sejak kecil ceritanya sudah menggemparkan karena tali pusarnya belum bisa dipotong dengan senjata apapun sampai usia 1 tahun, pada akhirnya tali pusar Tetuka bisa dipotong juga oleh sarung senjata konta yang dibawakan oleh Arjuna.

Pada game ini cerita berawal ketika khayangan diserang oleh kerajaan trabelasuket yang dikarenakan lamaran dari kerajaan trabelasuket ke salah satu putri khayangan ditolak. Pada level 1 Tetuka harus melawan kerajaan trabelasuket untuk mempertahankan khayangan. Tetuka harus melawan semua musuh yang menyerang, setelah tinggal melawan pemimpin pasukan yang menyerang, pemimpin pasukan kerajaan trabelasuket kabur meninggalkan khayangan.

Setelah berhasil mengusir kerajaan trabelasuket, masuk ke level 2 dengan cerita Tetuka yang dimasukkan kedalam kawah candradimuka oleh untuk berlatih,

dalam latihannya jabang tetuka harus mengumpulkan 7 senjata dan menghindari penjaga kawah yang menyerang, dan setelah keluar dari kawah candradimuka Tetuka muncul sebagai laki-laki dewasa dan masuk ke level 3 yang mulai saat itu berganti nama menjadi Gatotkaca, setelah berlatih dan mendapatkan banyak ilmu Gatotkaca melakukan serangan balik ke kerajaan Trabelasuket dan mngalahkan raja kerajaan Trabelasuket yaitu Naga Percona. Setelah mengalahkan Naga Percona, pemain berhasil menyelesaikan game ini dan akan disajikan slide *ending game*.

3.1.3.2. Gameplay

Tugas utama pemain (*gamer*) dalam memainkan *game* ini yaitu melalui level-level dimana setiap bagian level menyediakan petualangan dengan menampilkan beberapa karakter level tersebut maupun karakter musuh yang harus dilawan. Dalam *game* Legenda Gatotkaca, pemain memainkan *game* sebagai *single-player* yang dapat menjelajahi permainan dengan menggerakkan karakter pemain (*player character*) ke arah kanan, kiri, atas atau bawah, dan spasi untuk memukul lawan.

Saat memainkan *game*, pemain dapat berinteraksi dengan game yang diantaranya :

1. Memilih menu Mulai Baru untuk memulai permainan dari awal.
2. Memilih menu Petunjuk menampilkan aturan – aturan pada game seperti kontrol yang digunakan dan penjelasan item-item yang tersedia di dalam *game*.
3. Saat menjelajahi wilayah tertentu dalam arena permainan, karakter pemain dapat bertemu dengan musuh dan melakukan pertarungan.
4. Mengambil item kendi untuk menambahkan darah.
5. Mengambil item kepala Gatotkaca untuk menambahkan nyawa.
6. Pada level tertentu karakter pemain diwajibkan mengumpulkan item senjata yaitu keris sebanyak 7 buah, kalau sudah berhasil mengumpulkan 7 keris maka permainan dapat dilanjutkan ke level berikutnya, sedangkan kalau

belum berhasil mengumpulkan 7 keris maka permainan masih terus berlanjut di level tersebut sampai berhasil mendapatkan 7 keris.

7. Menggunakan Menu Permainan untuk menghentikan permainan sementara dan memilih menu yang tersedia, pemain dapat memilih merestart permainan, melanjutkan permainan, dan keluar dari permainan.
8. Menggerakkan Karakter untuk berjalan mendekati atau menjauhi musuh dan item.

Ketika pemain dihadapkan pada musuh yang ada pada arena permainan, pemain dapat melakukan pertarungan dengan musuh dengan cara memukul musuh tersebut. Pemain juga dapat melewati musuh tersebut tanpa melakukan pertarungan. Namun pada musuh terakhir atau boss musuh dalam *game* harus dikalahkan. Ketika pemain kalah dalam pertarungan maka pemain dapat melanjutkan permainan kalau masih mempunyai nyawa, sedangkan kalau sudah tidak mempunyai nyawa maka pemain diberikan pilihan untuk kembali ke menu utama atau bermain kembali dari awal.

3.1.3.3. Analisis Tingkat Kesulitan

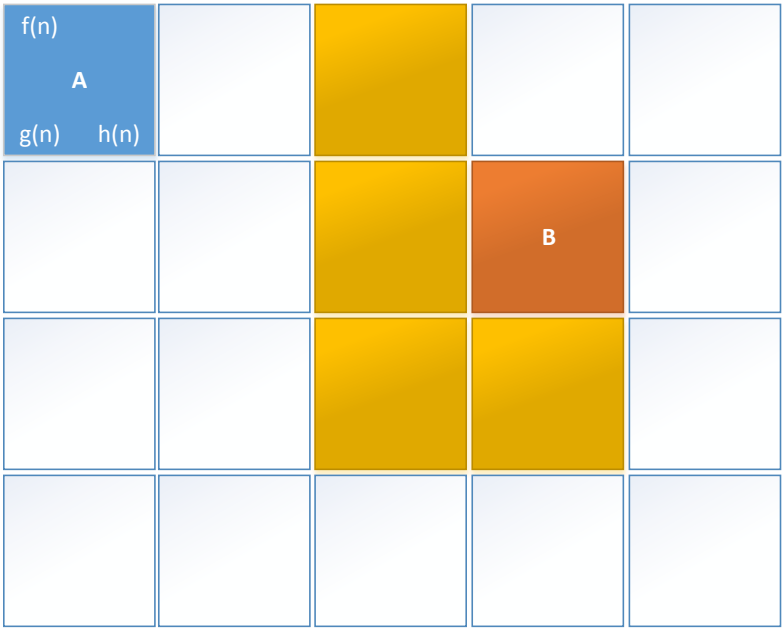
Tingkat kesulitan yang disediakan dalam *game* Legenda Gatotkaca terdiri dari tiga tingkat yang berdasarkan level permainan, yaitu level 1, level 2, dan level 3. Perbedaan tingkat kesulitan dalam game Legenda Gatotkaca dapat dilihat pada tabel 3.2 berikut :

Tabel 3.2 Analisis Tingkat Kesulitan

Perbedaan	Tingkat Kesulitan		
	Level 1	Level 2	Level 3
Musuh (*Enemy*)	Kemampuan musuh (*Health Point*, *Attack Power* dan *Defense*) sesuai dengan yang telah ditetapkan sebagaimana terlihat pada tabel perancangan musuh.	Kemampuan musuh (*Health Point*, *Attack Power* dan *Defense*) meningkat 25% dari kemampuan level 1 sebagaimana terlihat pada tabel perancangan musuh	Kemampuan musuh (*Health Point*, *Attack Power* dan *Defense*) meningkat 25% dari kemampuan level 2 sebagaimana terlihat pada tabel perancangan musuh

3.1.4. Analisis Algoritma A*

Algoritma A* di dalam *game* akan diterapkan pada NPC (*Non Playable Character*) musuh sebagai penentuan jalur terpendek atau *pathfinding* menuju target yaitu *player*, dalam hal ini *node* A merupakan NPC dan *node* B adalah *player*. Cara kerja algoritma A* di dalam *game* lebih jelasnya bisa dilihat pada gambar 3.2.



Gambar 3.2 Tahap Awal Pencarian Algoritma A*

Pada gambar 3.2 pencarian wilayah dibagi ke dalam *node-node* yang terdiri dari *node* berwarna biru merupakan *node* awal (*node* A), *node* berwarna merah merupakan *node* akhir (*node* B) dan yang berwarna coklat merupakan penghalang (*unwalkable node)*.

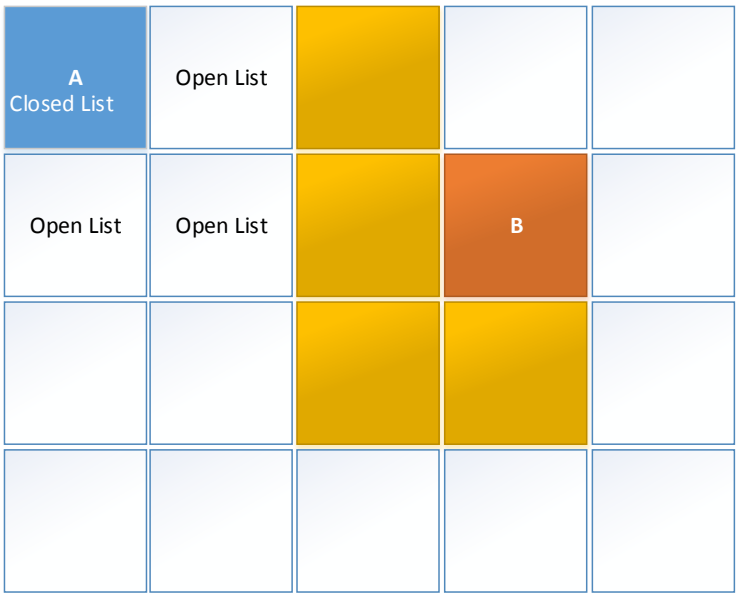
Selanjutnya adalah, melakukan pencarian untuk menemukan jalan terpendek dengan cara memeriksa *node-node* yang berdekatan dengan *node* awal untuk bisa sampai *node* B. Berikut adalah langkah pencarian yang dilakukan oleh *node* awal:

1. Langkah pertama, dimulai dari *node* awal yang ditambahkan ke dalam *Open list* yang merupakan *list* untuk *node-node* yang harus diperiksa.
2. Periksa semua *node* yang berdekatan dengan *node* awal apakah bisa dilalui atau tidak, *node* penghalang yang berwarna abu-abu diabaikan lalu

tambahkan semua *node* tersebut ke dalam *open list*, untuk setiap *node* yang baru ditambahkan, simpan *node* awal sebagai "*parent*" yang nantinya akan digunakan untuk menelusuri jalan.

3. Hapus *node* awal dari *open list*, dan tambahkan ke dalam *closed list. Closed list* merupakan *list* untuk *node* yang tidak perlu diperiksa atau dalam artian merupakan bagian dari jalur terpendek yang sudah didapatkan.

Untuk lebih jelasnya *open list* dan *closed list* diilustrasikan pada gambar 3.3

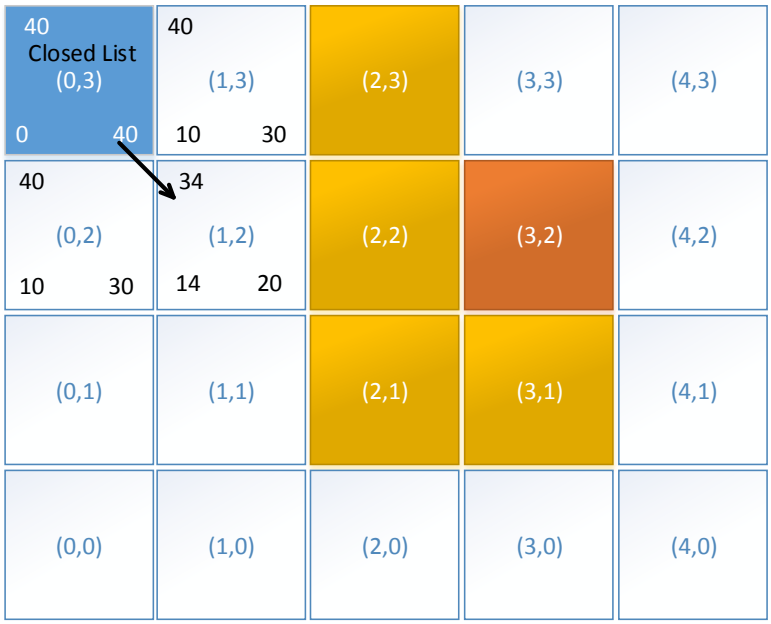


Gambar 3.3 Ilustrasi Open List dan Closed List

Pada gambar 3.3 diatas *node* awal yang berwarna biru sudah ditambahkan ke dalam *closed list.* Sedangkan kotak yang berada di sekelilingnya adalah *open list* yang harus diperiksa satu persatu nilai dari masing-masing *node open list*.

Langkah berikutnya adalah memilih *node* yang memiliki nilai f terendah di dalam *open list* dengan menggunakan rumus (f = g + h). g adalah nilai yang dikeluarkan dari *node* A ke *node* yang ada disekitarnya. Sedangkan h adalah fungsi *heuristic* yang digunakan untuk menghitung perkiraan biaya yang dikeluarkan dari *node* awal menuju *node* tujuan.

Proses ini dilakukan secara berulang-ulang, dimulai dari *open list* dan menemukan *node* dengan nilai f paling rendah. Perhitungan persamaan untuk mendapatkan nilai f dijelaskan pada ilustrasi gambar 3.4.



Gambar 3.4 Perhitungan menentukan nilai F

Gambar 3.4 menunjukkan pada *open list* dari *node* awal yaitu NPC nilai f yang dihasilkan adalah 34 yang didapatkan dari menambahkan nilai g = 14 dan nilai h = 20. Di dalam contoh ini g akan diberi nilai 10 apabila *node* bergerak secara *vertical* atau *horizontal* dan diberi nilai 14 apabila *node* bergerak secara *diagonal*.

Perhitungan nilai h yaitu nilai estimasi jalur terpendek dari *node* awal ke *node* tujuan yaitu *player* dengan menggunakan rumus h = ((Xtujuan – Xstart) + (Ytujuan – Ystart) * 10). Sebagai contoh *node* yang terletak dibawah dari *node* awal diberikan nilai h = 30 yang didapatkan dari perhitungan sebagai berikut:

h = ((Xtujuan – Xstart) + (Ytujuan – Ystart) * 10)

= ((3 – 0) + (2 - 2) *10

= 3 * 10 = 30

Setelah didapatkan nilai f dari setiap *node* yang ada di *open list*, maka akan dipilih *node* yang memiliki nilai f paling rendah yang kemudian disebut dengan *current node*, langkah selanjutnya adalah sebagai berikut.

1. Hapus *current node* dari *open list* dan masukkan ke dalam *closed list.*
2. Periksa semua *node* yang berdekatan dengan *current node*, abaikan *node* penghalang. Jika tidak ada pada *open list,* tambahkan ke dalam *open list,* lalu tambahkan "*parent*" untuk *node* tersebut.
3. Jika *node* sudah ada pada *open list,* lalu bandingkan dengan *node* awal apakah jalur yang di ambil lebih baik dari *node* awal, dengan dengan memeriksa nilai G-nya apakah lebih rendah jika dalam menggunakan *node* ini, jika lebih rendah maka hitung ulang nilai f dan g serta merubah arah *pointer,* jika tidak maka jangan lakukan apa-apa.

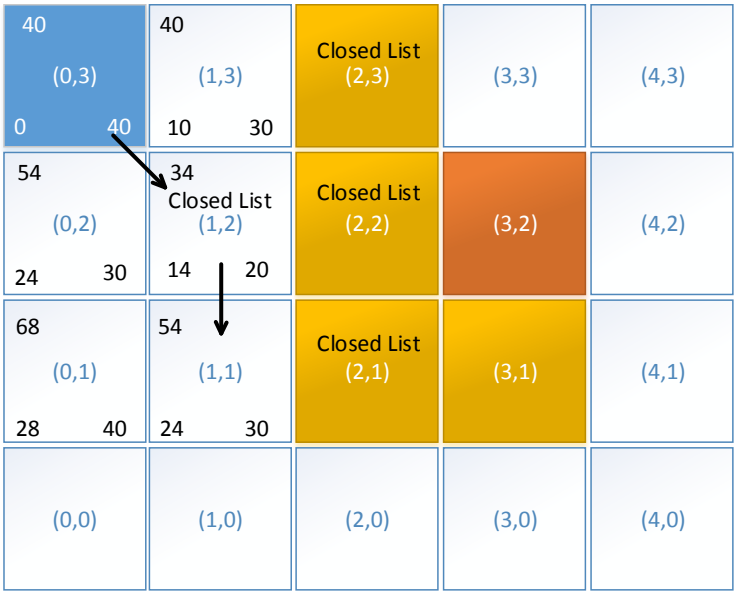
Langkah pertama menghasilkan *node-node* di dalam *open list* dan *closed list* dijelaskan pada tabel 3.3

Tabel 3.3 Perhitungan Langkah Pertama

Open List	(0,2) (1,2) (1,3)
Closed List	(0,3)
Parent	(0,3)
Current Node	(1,2)

Gambar 3.4 merupakan penjelasan langkah pertama pencarian jalur dari algoritma *A**.

Perhitungan pada langkah kedua yang berada pada *node* dengan koordinat (1,2) dapat dilihat pada gambar 3.5



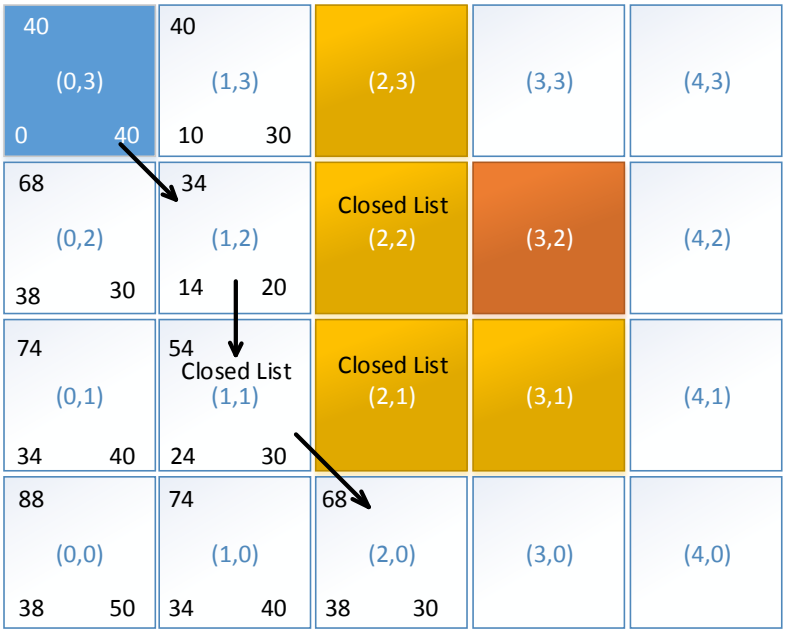
Gambar 3.5 Penjelasan A* Pada Langkah Kedua

Tabel 3.4 menunjukan penjelasan tentang pencarian jalur terpendek untuk langkah kedua.

Tabel 3.4 Perhitungan Langkah Kedua

Open List	(1,3) (0,2) (0, 1) (1, 1)
Closed List	(0,3) (2,3) (1,2) (2,2) (2,1)
Parent	(1,2)
Current Node	(1,1)

Perhitungan langkah ketiga yang berada pada *node* dengan koordinat (1,1) dapat dilihat pada gambar 3.6.



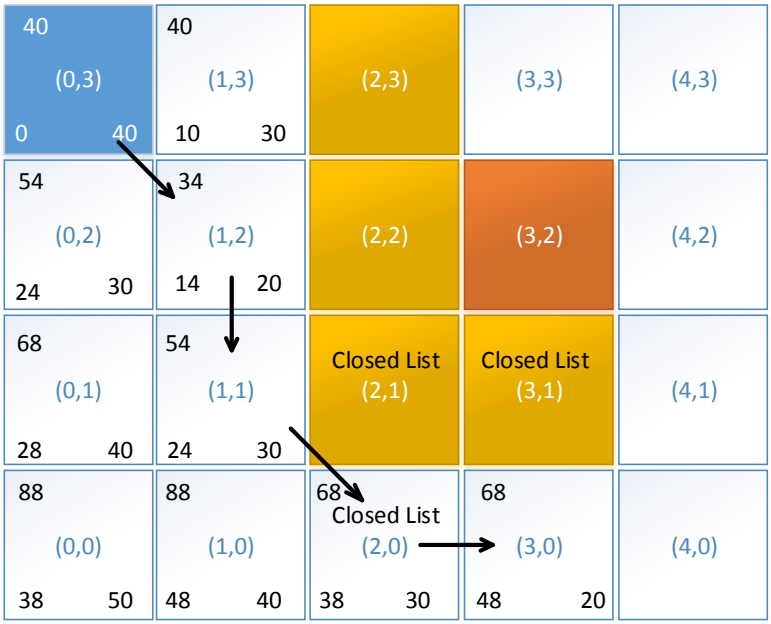
Gambar 3.6 Penjelasan A* Pada Langkah Ketiga

Tabel 3.5 menunjukan penjelasan tentang pencarian jalur terpendek untuk langkah ketiga dapat dilihat pada tabel 3.5.

Tabel 3.5 Perhitungan Langkah Ketiga

Open List	(0,2) (0,1) (0,0) (1,0)
Closed List	(1,2) (2,2) (1,1) (2,1)
Parent	(1,1)
Current Node	(2,0)

Perhitungan langkah keempat yang berada pada *node* dengan kooedinat (2,0) dapat dilihat pada gambar 3.7.



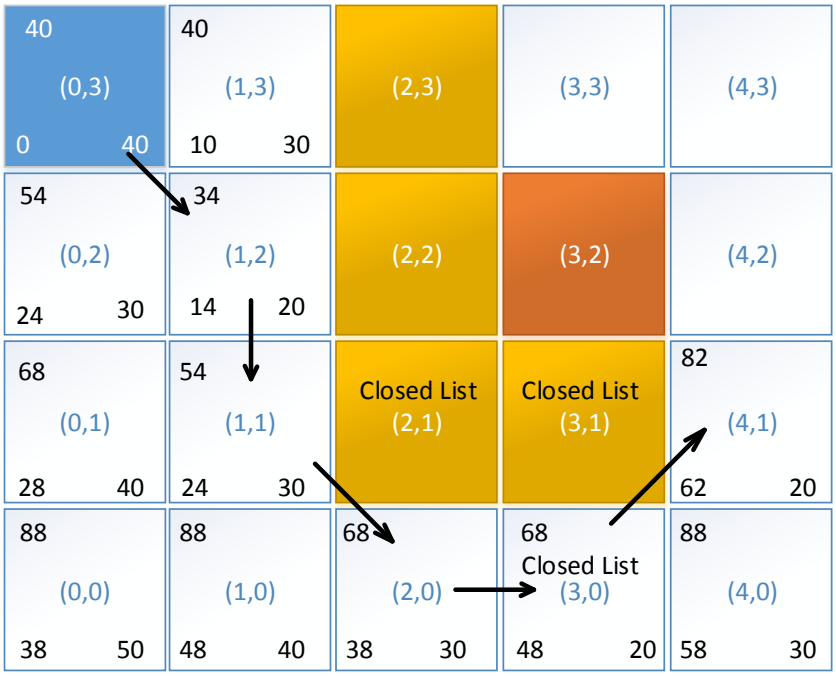
Gambar 3.7 Penjelasan A* Pada Langkah Keempat

Tabel 3.6 menunjukan penjelasan tentang pencarian jalur terpendek untuk langkah keempat.

Tabel 3.6 Perhitungan Langkah Keempat

Open List	(1,0) (3,0)
Closed List	(1,1) (2,1) (3,1) (2,0)
Parent	(2,0)
Current Node	(3,0)

Perhitungan pada langkah kelima yang berada pada *node* dengan koordinat (3,0) yang memiliki nilai f = 68 bisa dilihat pada gambar 3.8.



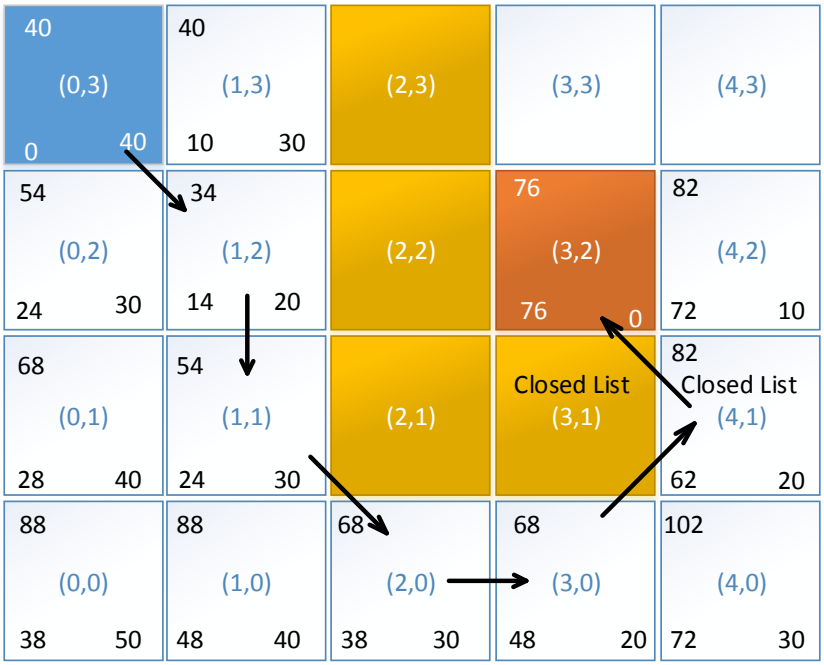
Gambar 3.8 Penjelasan A* Pada Langkah Kelima

Pada tabel 3.7 penjelasan melalui tabel untuk pencarian jalur terpendek pada langkah kelima.

Tabel 3.7 Perhitungan Langkah Kelima

Open List	(4,0) (4,1)
Closed List	(2,0) (2,1) (3,0) (3,1)
Parent	(3,0)
Current Node	(4,1)

Perhitungan pada langkah keenam yang berada pada *node* dengan koordinat (4,1) yang memiliki nilai f = 82 bisa dilihat pada gambar 3.9.



Gambar 3.9 Penjelasan A* Pada Langkah Keenam

Pada tabel 3.8 penjelasan melalui tabel untuk pencarian jalur terpendek pada langkah keenam.

Tabel 3.8 Perhitungan Langkah Keenam

Open List	(3,2) (4,0) (4,2)
Closed List	(3,0) (3,1) (4,1)
Parent	(4,1)
Current Node	(3,2)

3.1.5. Analisis Kebutuhan *Non*-Fungsional

Analisis kebutuhan *non*-fungsional merupakan analisis yang dibutuhkan untuk menentukan spesifikasi kebutuhan sistem. Spesifikasi ini juga meliputi elemen atau komponen-komponen apa saja yang dibutuhkan untuk sistem yang akan dibangun sampai dengan sistem tersebut diimplementasikan. Analisis kebutuhan ini juga menentukan spesifikasi masukan yang diperlukan sistem, keluaran yang akan dihasilkan sistem dan proses yang dibutuhkan untuk mengolah masukan sehingga menghasilkan suatu keluaran yang diinginkan. Pada analisis kebutuhan sistem non-fungsional ini dijelaskan analisis kebutuhan perangkat lunak, analisis kebutuhan perangkat keras dan analisis pengguna.

3.1.5.1. Analisis Perangkat Keras

Perangkat keras atau hardware merupakan salah satu hal yang penting karena tanpa hardware yang memenuhi syarat, program yang akan dibuat tidak akan dapat berjalan. Agar aplikasi dapat berjalan dengan baik, maka dibutuhkan perangkat keras yang sesuai dengan kebutuhan aplikasi.

Spesifikasi standar perangkat keras yang dibutuhkan dalam pembangunan game ini adalah sebagai berikut:

1. CPU dengan kecepatan minimal 1.6 Ghz
2. RAM 1 GB
3. VGA 512 MB
4. Hard disk 32 GB
5. Monitor standar 1024 x 768
6. Mouse, keyboard dan speaker

Adapun spesifikasi standar perangkat keras yang dibutuhkan untuk dapat menjalankan aplikasi yang akan dibangun adalah:

1. CPU dengan kecepatan minimal 1.6 Ghz
2. RAM 512 MB
3. Hard disk 8 GB
4. Monitor standar 1024 x 768

5. Keyboard, mouse dan speaker

Berdasarkan analisis diatas dapat disimpulkan bahwa kebutuhan perangkat keras untuk membangun game Legenda Gatotkaca tidak terlalu membutuhkan spesifikasi yang tinggi. Sedangkan untuk menjalankan *game* Legenda Gatotkaca dengan spesifikasi rendah sudah dapat dimainkan.

3.1.5.2. Analisis Perangkat Lunak

Perangkat lunak atau software merupakan hal yang terpenting dalam mendukung kinerja sebuah sistem. Perangkat lunak digunakan dalam sebuah sistem merupakan perintah-perintah yang diberikan kepada perangkat keras agar dapat saling berinteraksi diantara keduanya. Perangkat lunak yang dibutuhkan untuk membangun aplikasi game legenda gatotkaca ini adalah sebagai berikut:

1. Sistem Operasi Windows 7.
2. Construct 2 sebagai tools yang digunakan untuk pengembangan *game*.
3. Corel Draw untuk membuat karakter *game*.

Perangkat lunak yang dibutuhkan pengguna untuk dapat menjalankan aplikasi ini adalah Sistem Operasi Windows XP keatas.

Berdasarkan hasil analisis diatas dapat disimpulkan bahwa untuk membangun game Legenda Gatotkaca membutuhkan beberapa perangkat lunak yang disebutkan diatas. Sedangkan untuk menjalankan game Legenda Gatotkaca membutuhkan minimal Windows XP untuk menjalankannya.

3.1.5.3. Analisis Pengguna

Analisis pengguna memberikan gambaran tentang target maupun profil pengguna untuk game yang akan dibangun..

3.1.5.3.1. Target Pengguna

Target pengguna (target user) untuk game ini adalah pengguna dengan karakteristik sebagai berikut :

Tabel 3.9 Target Pengguna

Pengguna	Usia	Tingkat Pendidikan	Keterampilan	Pengalaman
Siapa saja / bebas	13 tahun keatas	Minimal SMP atau sederajat	Dapat mengoperasikan komputer	Pernah bermain game sejenis *beat 'em up*, labirin, atau *shoot 'em up*

3.1.5.3.2. Profil Pengguna

Profil pengguna (user profile) untuk game Legenda Gatotkaca dapat dikelompokkan sebagai berikut :

1. User Pshycology Characteristic

Karakteristik psikologi pengguna untuk game ini haruslah pengguna yang memiliki kegemaran atau setidaknya berminat untuk memainkan game berjenis ini. Selain itu juga tertarik dengan kisah pewayangan dan ingin mempelajarinya.

2. User Knowledge and Experience Characteristic

Pengguna yang memainkan game ini harus memiliki pengetahuan dasar tentang cara pengoperasian komputer.

3. User Job and Task Characteristic

Untuk lebih memudahkan dalam memainkan game ini, pengguna harus memiliki tingkat frekuensi penggunaan (*frequency of use*) yang sedang. Kategori pekerjaan pengguna yang memainkan game ini mencakup Siswa, Mahasiswa atau Siapa saja yang ingin mempelajari cerita pewayangan dan gemar bermain game.

4. User Physical Characteristic

Kondisi fisik pengguna game ini diharuskan yang tidak memiliki kekurangan secara fisik sehingga nantinya tidak mengalami kesulitan dalam memainkan game ini. Selain itu, gender pengguna untuk game ini tidak dibatasi atau dengan kata lain, game ini dapat dimainkan oleh pengguna dengan gender apapun.

5. User Physical Environment

Dalam memainkan game ini, dibutuhkan cukup konsentrasi pengguna agar cerita dan pesan dalam game ini dapat tersampaikan dan dapat diingat dengan baik oleh pengguna.

6. User Tools

Selain seperangkat komputer, pengguna tidak perlu menggunakan alat tambahan lainnya dikarenakan hanya dengan memiliki komputer (lengkap dengan *keyboard* dan disarankan memiliki *speaker*), pengguna dapat dengan mudah memainkan game ini.

3.1.6. Analisis Kebutuhan Fungsional

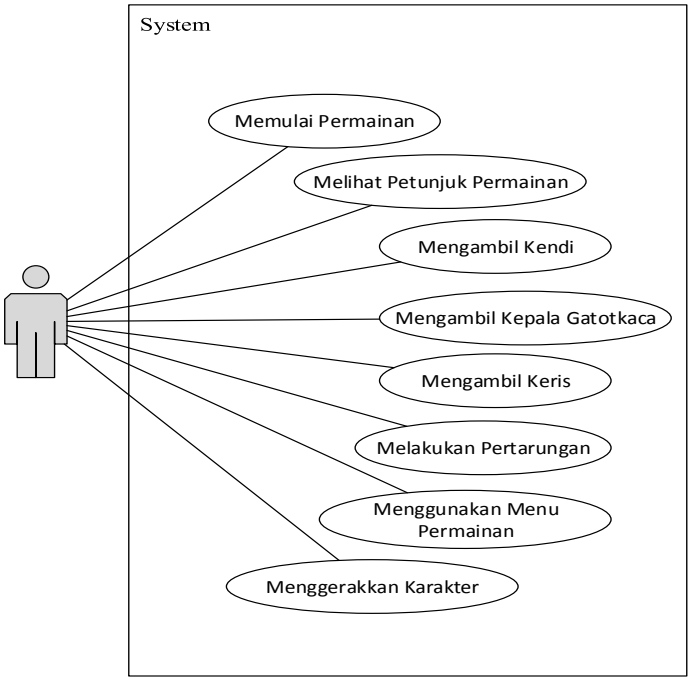
Analisis kebutuhan fungsional menggambarkan proses kegiatan yang akan diterapkan dalam sebuah sistem dan menjelaskan kebutuhan yang diperlukan sistem agar sistem dapat berjalan dengan baik sesuai kebutuhan.

3.1.6.1. Identifikasi Aktor

Tahap Pertama yang dilakukan dalam melakukan analisis berorientasi objek menggunakan UML adalah menentukan aktor atau pengguna sistem. Kata aktor dalam kontek UML, menampilkan peran (*roles*) yang pengguna (sesuatu diluar sistem yang dikembangkan yang dapat berupa perangkat keras, *end user*, sistem yang lain dan sebagainya).

3.1.6.2. *Use Case Diagram*

Use case diagram merupakan gambaran skenario dari interaksi antara user dengan sistem. Sebuah diagram menggambarkan hubungan antara aktor dan kegiatan yang dapat dilakukannya terhadap aplikasi.



Gambar 3.10 *Use Case* Game Legenda Gatotkaca

3.1.6.3. Definisi Aktor

Definisi *Actor* berfungsi untuk menjelaskan *Actor* yang terdapat pada *Diagram*. Definisi *Actor* diterangkan pada Tabel 3.10.

Tabel 3.10 Definisi Actor

Actor	**Deskripsi**
Pemain	Orang yang akan memainkan *Game.*

3.1.6.4. Definisi *Use Case*

Definisi *Use Case* berfungsi untuk menjelaskan proses yang terdapat pada setiap *Use Case*.

Tabel 3.11 Definisi *Use Case*

No	***Use Case***	**Deskripsi**
1	Memulai Permainan	Proses memulai permainan di dalam game
2	Melihat Petunjuk Permainan	Proses melihat petunjuk permainan di dalam game
3	Mengambil item kendi	Proses untuk mengambil item kendi
4	Mengambil item kepala Gatotkaca	Proses untuk mengambil item kepala Gatotkaca
5	Mengambil item keris	Proses untuk mengambil item keris
6	Melakukan Pertarungan	Proses menyajikan pertarungan antara karakter pemain dan musuh kepada *user*

7	Menggunakan Menu Permainan	Proses untuk menggunakan menu di dalam permainan
8	Menggerakkan Karakter	Proses untuk menggerakkan karakter

3.1.6.5. *Use Case* Skenario

Skenario *Use Case* menggambarkan alur penggunaan sistem dimana setiap skenario digambarkan dari sudut pandang aktor, seseorang atau piranti yang berinteraksi dengan perangkat lunak dalam berbagai cara.

3.1.6.5.1. Skenario *Use Case* Memulai Permainan

Tabel 3.12 Skenario *Use Case* Memulai Permainan

Identifikasi
Nama Use Case : Memulai Permainan
Deskripsi : Proses untuk menampilkan permainan baru
Aktor : Pemain
Kondisi Awal : Pemain berada di menu utama permainan
Pemicu : Pemain memilih menu mulai baru dari menu utama
Skenario Utama : 1. Sistem menampilkan slide intro permainan 2. Pemain menekan tombol lanjut 3. Sistem menampilkan permainan level 1
Skenario Alternatif : Tidak ada

3.1.6.5.2. Skenario *Use Case* Melihat Petunjuk Permainan

Tabel 3.13 Skenario *Use Case* Melihat Petunjuk Permainan

Identifikasi
Nama Use Case : Melihat Petunjuk Permainan
Deskripsi : Proses untuk menampilkan petunjuk permainan
Aktor : Pemain
Kondisi Awal : Pemain berada di menu utama permainan
Pemicu : Pemain memilih menu petunjuk dari menu utama
Skenario Utama : 1. Sistem menampilkan petunjuk permainan
Skenario Alternatif : Tidak ada

3.1.6.5.3. Skenario *Use Case* Mengambil Kendi

Tabel 3.14 Skenario *Use Case* Mengambil Kendi

Identifikasi
Nama Use Case : Mengambil Kendi
Deskripsi : Proses untuk mengambil kendi
Aktor : Pemain
Kondisi Awal : Pemain berada di dalam permainan
Pemicu : Pemain menggerakkan karakter mendekati kendi
Skenario Utama : 1. Karakter menyentuh kendi 2. Sistem menambahkan darah karakter
Skenario Alternatif : Tidak ada.

3.1.6.5.4. Skenario *Use Case* Mengambil Kepala Gatotkaca

Tabel 3.15 Skenario *Use Case* Mengambil Kepala Gatotkaca

Identifikasi
Nama Use Case : Mengambil Kepala Gatotkaca
Deskripsi : Proses untuk mengambil Kepala Gatotkaca
Aktor : Pemain
Kondisi Awal : Pemain berada di dalam permainan
Pemicu : Pemain menggerakkan karakter mendekati kepala Gatotkaca
Skenario Utama : 1. Karakter menyentuh kepala Gatotkaca 2. Sistem menambahkan nyawa karakter
Skenario Alternatif : Tidak ada.

3.1.6.5.5. Skenario Use Case Mengambil Keris

Tabel 3.16 Skenario *Use Case* Mengambil Keris

Identifikasi
Nama Use Case : Mengambil Keris
Deskripsi : Proses untuk mengambil Keris
Aktor : Pemain
Kondisi Awal : Pemain berada di dalam permainan
Pemicu : Pemain menggerakkan karakter mendekati keris
Skenario Utama : 1. Karakter menyentuh keris 2. Sistem menambahkan koleksi keris pemain 3. Sistem memeriksa apakah koleksi sudah terkumpul sebanyak 7 4. Keris sudah terkumpul 7 buah 5. Sistem membuka pintu keluar dari level 2

Skenario Alternatif : 4a. Keris belum terkumpul 7 buah 5a. Sistem belum membuka pintu keluar dan pemain harus mengumpulkan keris sebanyak 7 buah.

3.1.6.5.6. Skenario *Use Case* Melakukan Pertarungan

Tabel 3.17 Skenario *Use Case* Melakukan Pertarungan

Identifikasi
Nama Use Case : Melakukan Pertarungan
Deskripsi : Proses untuk melakukan pertarungan
Aktor : Pemain
Kondisi Awal : Pemain berada di dalam permainan
Pemicu : Pemain menggerakkan karakter untuk melakukan pertarungan
Skenario Utama : 1. Membuat karakter memukul musuh 2. Memeriksa apakah pukulan mengenai musuh atau tidak 3. Musuh melakukan serangan ke karakter 4. Memeriksa apakah karakter terkena serangan musuh atau tidak 5. Karakter terkena serangan sampai darahnya habis 6. Memeriksa apakah karakter mempunyai nyawa atau tidak
Skenario Alternatif : 2a. Pukulan mengenai musuh dan darah musuh berkurang 2b. Pukulan tidak mengenai musuh dan darah musuh tidak berkurang 4a. Karakter terkena serangan dan darah karakter berkurang 4b. Karakter tidak terkena serangan dan darah karakter tidak berkurang 6a. Karakter mempunyai nyawa dan darah karakter kembali penuh 6b. Karakter tidak mempunyai nyawa dan karakter mati/kalah

3.1.6.5.7. Skenario Use Case Menggunakan Menu Permainan

Tabel 3.18 Skenario *Use Case* Menggunakan Menu Permainan

Identifikasi
Nama Use Case : Menggunakan Menu Permainan
Deskripsi : Proses untuk menggunakan menu permainan
Aktor : Pemain
Kondisi Awal : Pemain berada di dalam permainan
Pemicu : Pemain menekan tombol ESC
Skenario Utama : 1. Pemain menekan tombol ESC 2. Sistem menghentikan permainan sementara dan menampilkan menu permainan 3. Pemain menekan tombol restart untuk memulai permainan dari awal 4. Pemain menekan tombol play untuk melanjutkan permainan yang sedang berjalan 5. Pemain menekan tombol keluar untuk kembali ke menu utama
Skenario Alternatif : Tidak ada

3.1.6.5.8. Skenario Use Case Menggerakkan Karakter

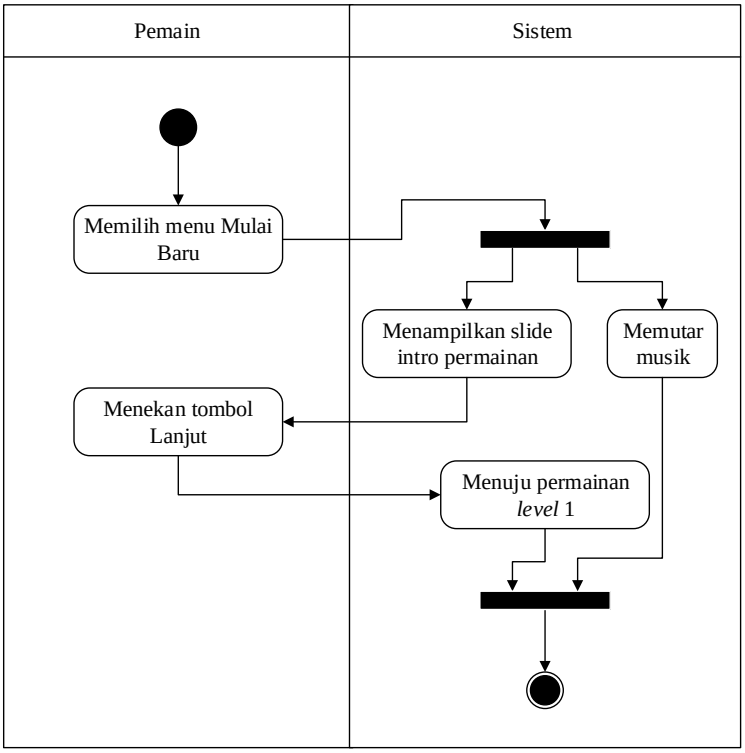
Tabel 3.19 Skenario *Use Case* Menggerakkan Karakter

Identifikasi
Nama Use Case : Menggerakkan Karakter
Deskripsi : Proses untuk menggerakkan karakter
Aktor : Pemain
Kondisi Awal : Pemain berada di dalam permainan
Pemicu : Pemain menekan tombol arah pada *keyboard*
Skenario Utama : 1. Pemain menekan tombol arah atas 2. Karakter bergerak ke atas 3. Pemain menekan tombol arah bawah 4. Karakter bergerak ke bawah 5. Pemain menekan tombol arah kanan 6. Karakter bergerak ke kanan 7. Pemain menekan tombol arah kiri 8. Karakter bergerak ke kiri
Skenario Alternatif : Tidak ada

3.1.6.6. *Activity Diagram*

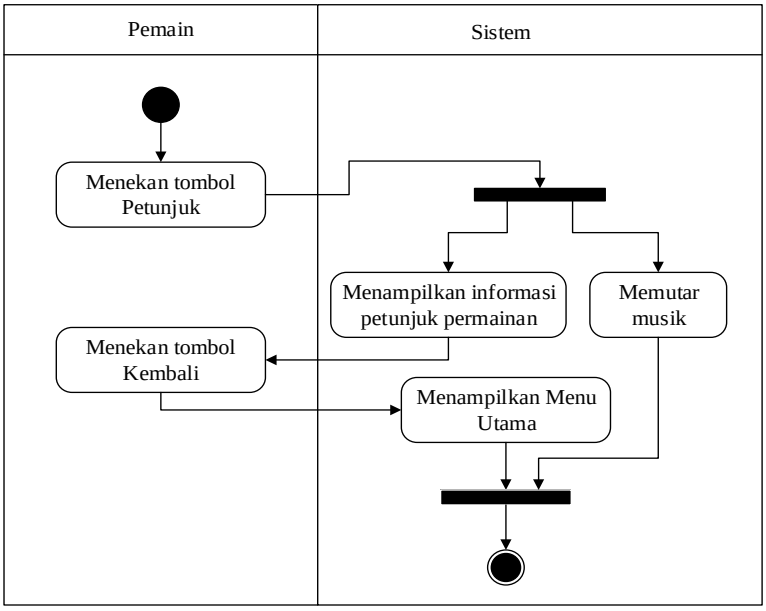
Activity Diagram menggambarkan berbagai alir aktivitas (*workflow*) dalam sistem yang dirancang, bagaimana masing-masing alir berawal, pilihan (*decision*) yang mungkin terjadi, dan bagaimana mereka berakhir. *Activity Diagram* dalam *game* Legenda Gatotkaca adalah sebagai berikut.

3.1.6.6.1. Activity Diagram Memulai Permainan



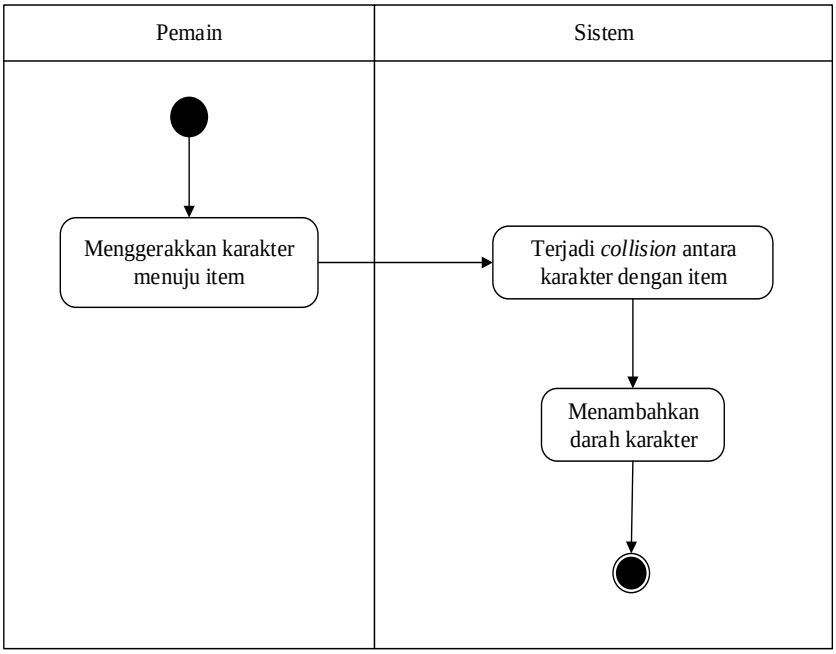
Gambar 3.11 Activity Diagram Memulai Permainan

3.1.6.6.2. Activity Diagram Melihat Petunjuk Permainan



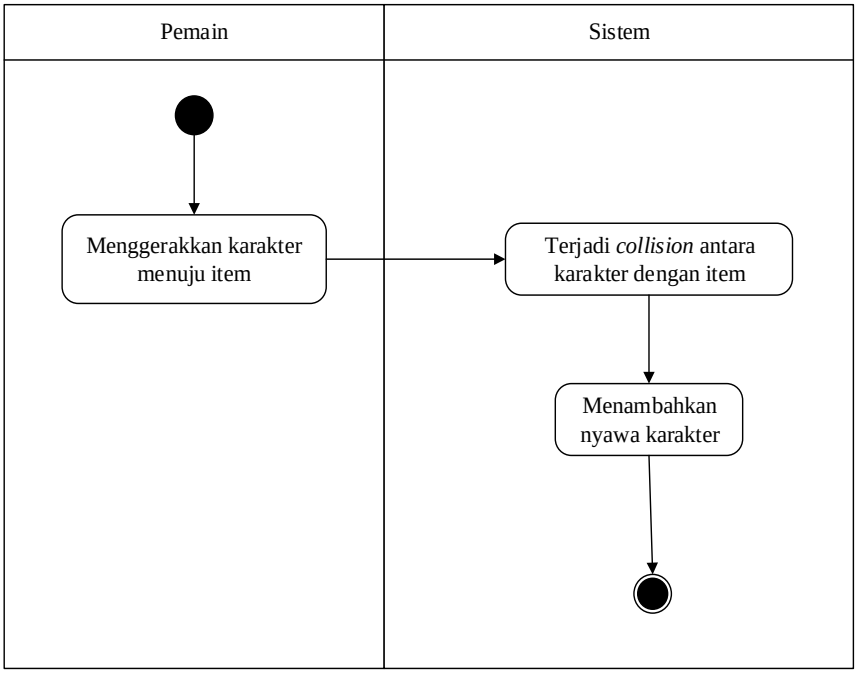
Gambar 3.12 Activity Diagram Melihat Petunjuk Permainan

3.1.6.6.3. *Activity Diagram* Mengambil Kendi



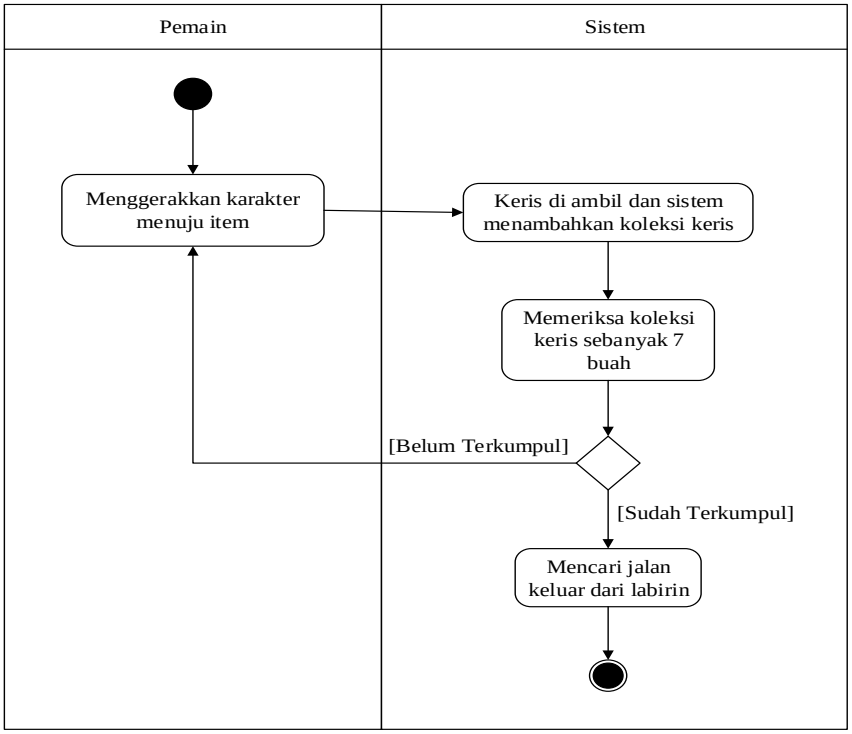
Gambar 3.13 Activity Diagram Mengambil Kendi

3.1.6.6.4. *Activity Diagram* Mengambil Kepala Gatotkaca



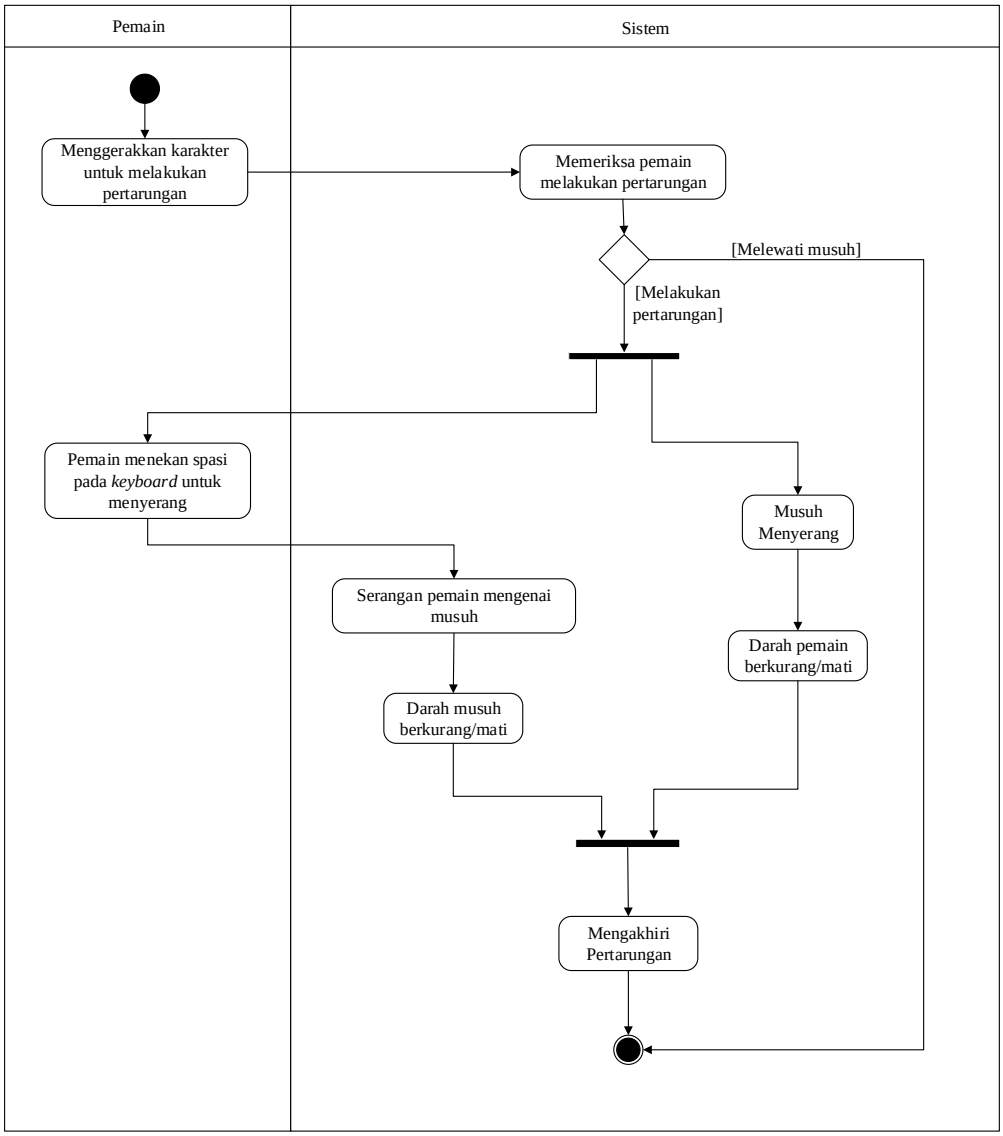
Gambar 3.14 Activity Diagram Mengambil Kepala Gatotkaca

3.1.6.6.5. *Activity Diagram* Mengambil Keris



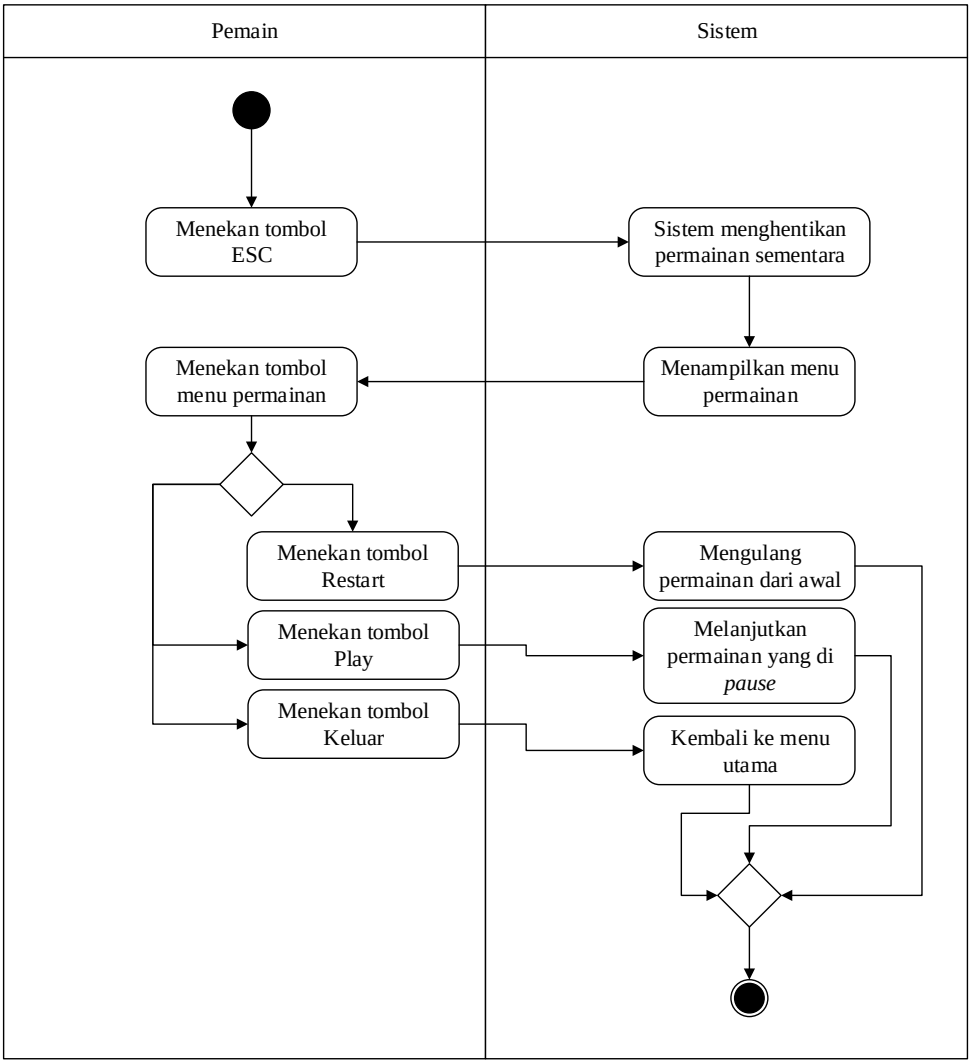
Gambar 3.15 Activity Diagram Mengambil Keris

3.1.6.6.6. *Activity Diagram* Melakukan Pertarungan



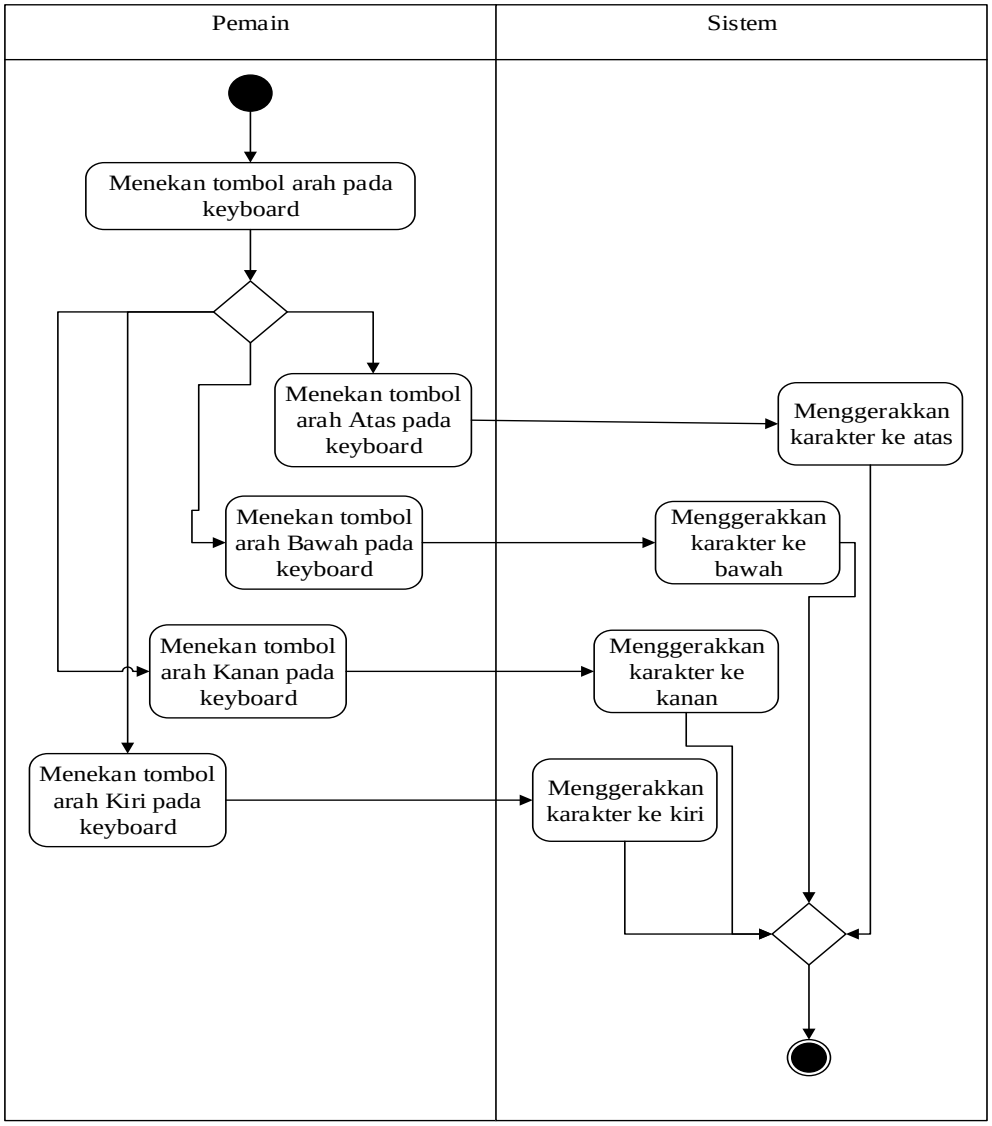
Gambar 3.16 Activity Diagram Melakukan Pertarungan

3.1.6.6.7. *Activity Diagram* Menggunakan Menu Permainan



Gambar 3.17 Activity Diagram Menggunakan Menu Permainan

3.1.6.6.8. *Activity Diagram* Menggerakkan Karakter



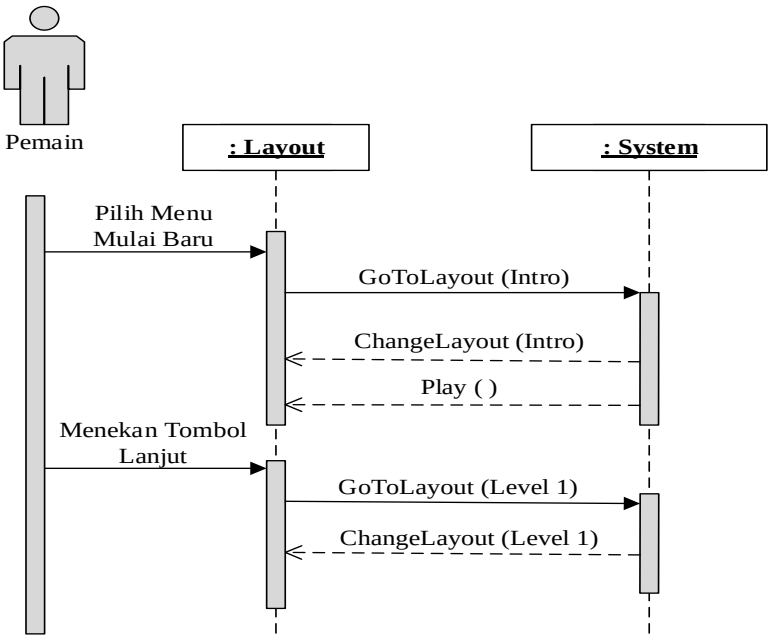
Gambar 3.18 Activity Diagram Menggerakkan Karakter

3.1.6.7. *Sequence Diagram*

Sequence Diagram menggambarkan interaksi antar objek di dalam dan di sekitar sistem berupa message yang digambarkan terhadap waktu. Sequence diagram terdiri atas dimensi vertikal (waktu) dan dimensi horizontal (objek-objek yang terkait).

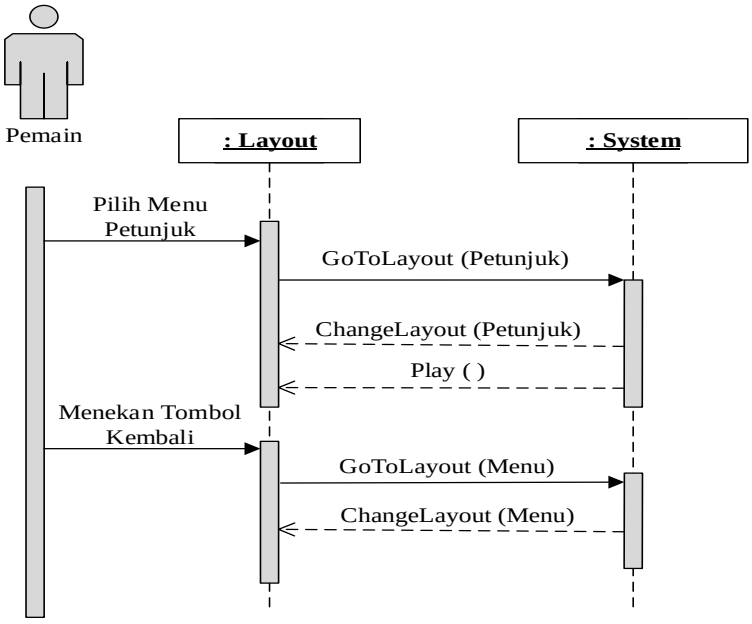
Sequence Diagram yang ada dalam game Legenda Gatotkaca adalah sebagai berikut :

1. *Sequence Diagram* Memulai Permainan



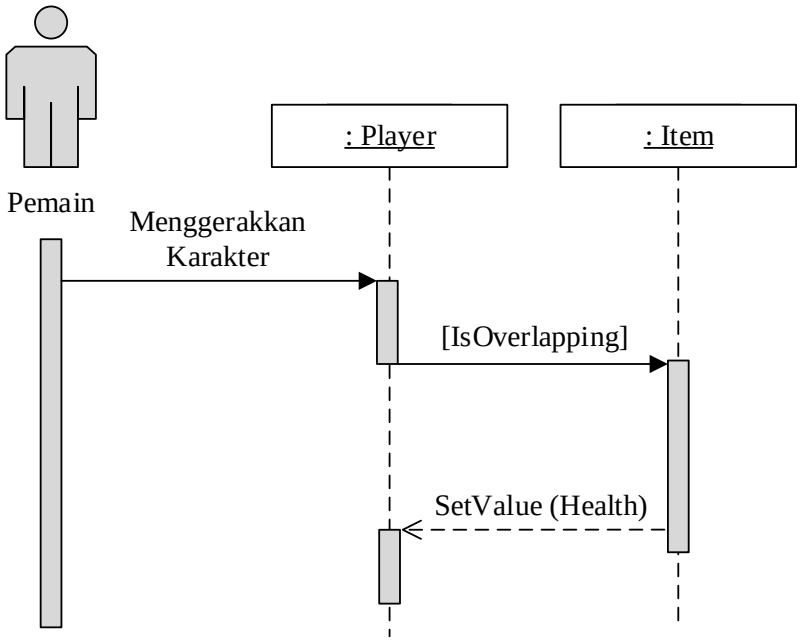
Gambar 3.19 Sequence Diagram Memulai Permainan

2. *Sequence Diagram* Melihat Petunjuk Permainan



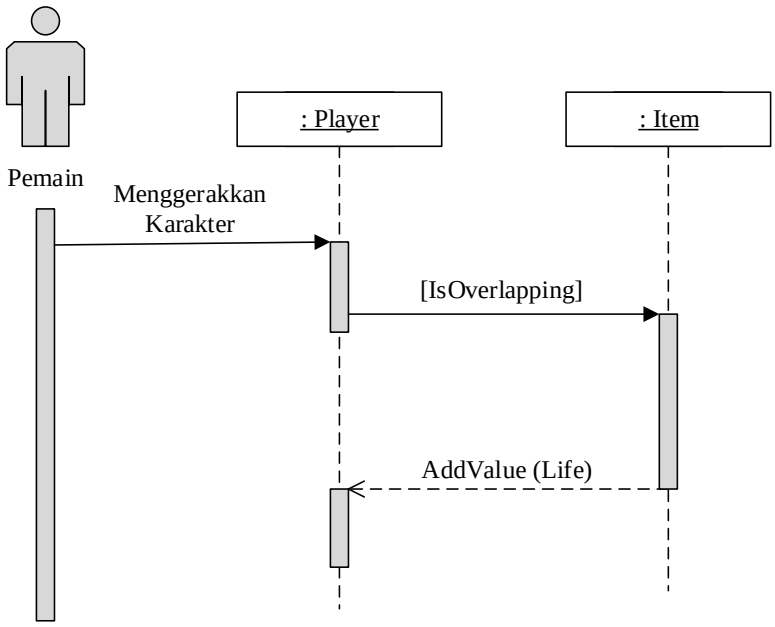
Gambar 3.20 Sequence Diagram Melihat Petunjuk Permainan

3. *Sequence Diagram* Mengambil Kendi



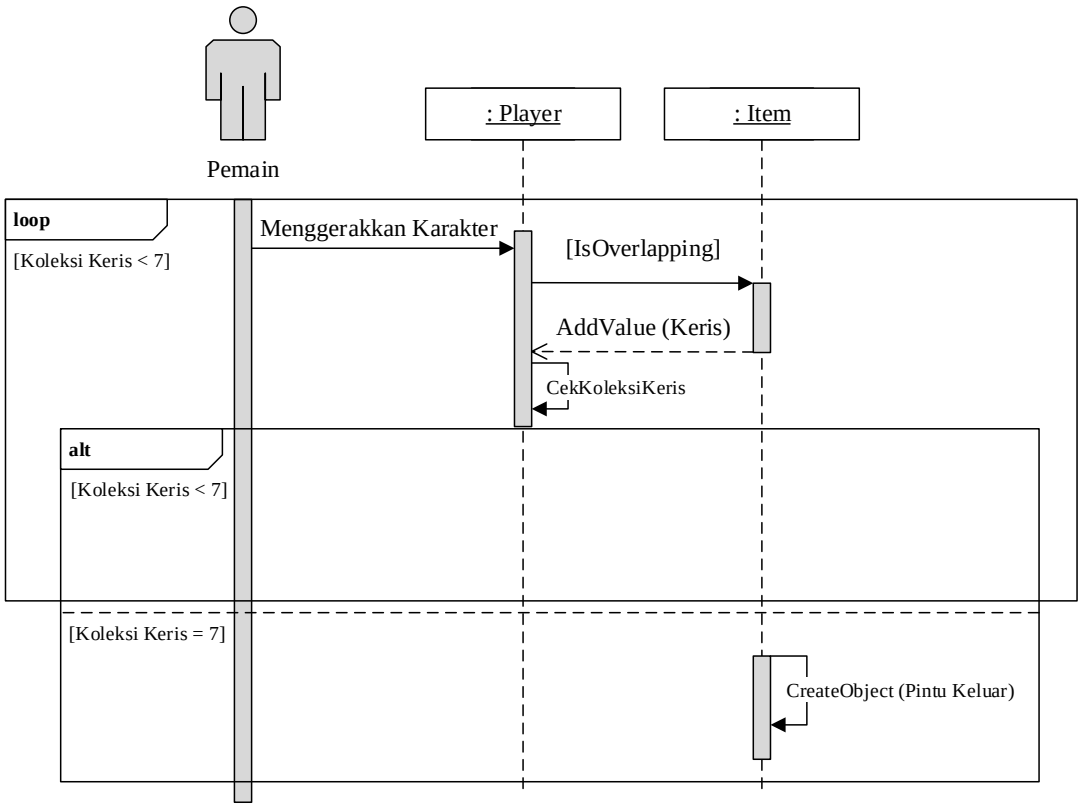
Gambar 3.21 Sequence Diagram Mengambil Kendi

4. *Sequence Diagram* Mengambil Kepala Gatotkaca



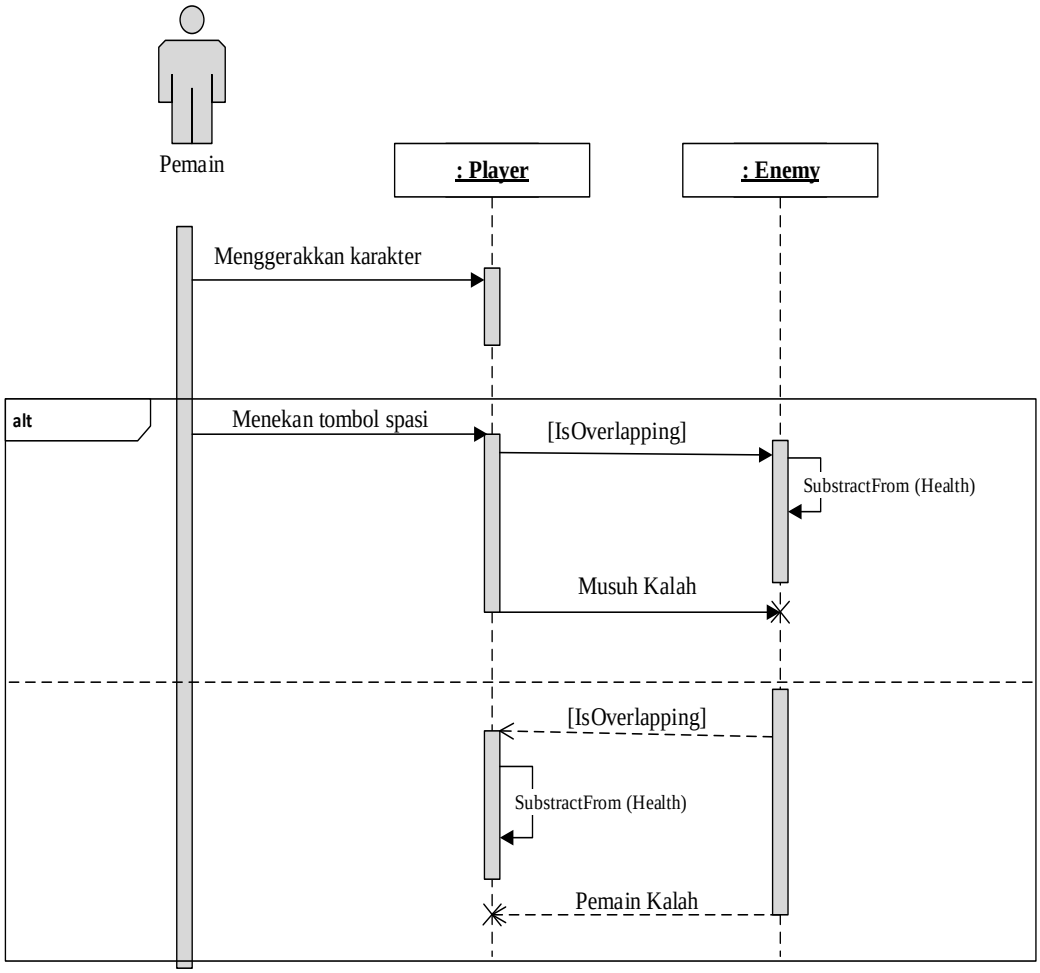
Gambar 3.22 Sequence Diagram Mengambil Kepala Gatotkaca

5. Sequence Diagram Mengambil Item Keris



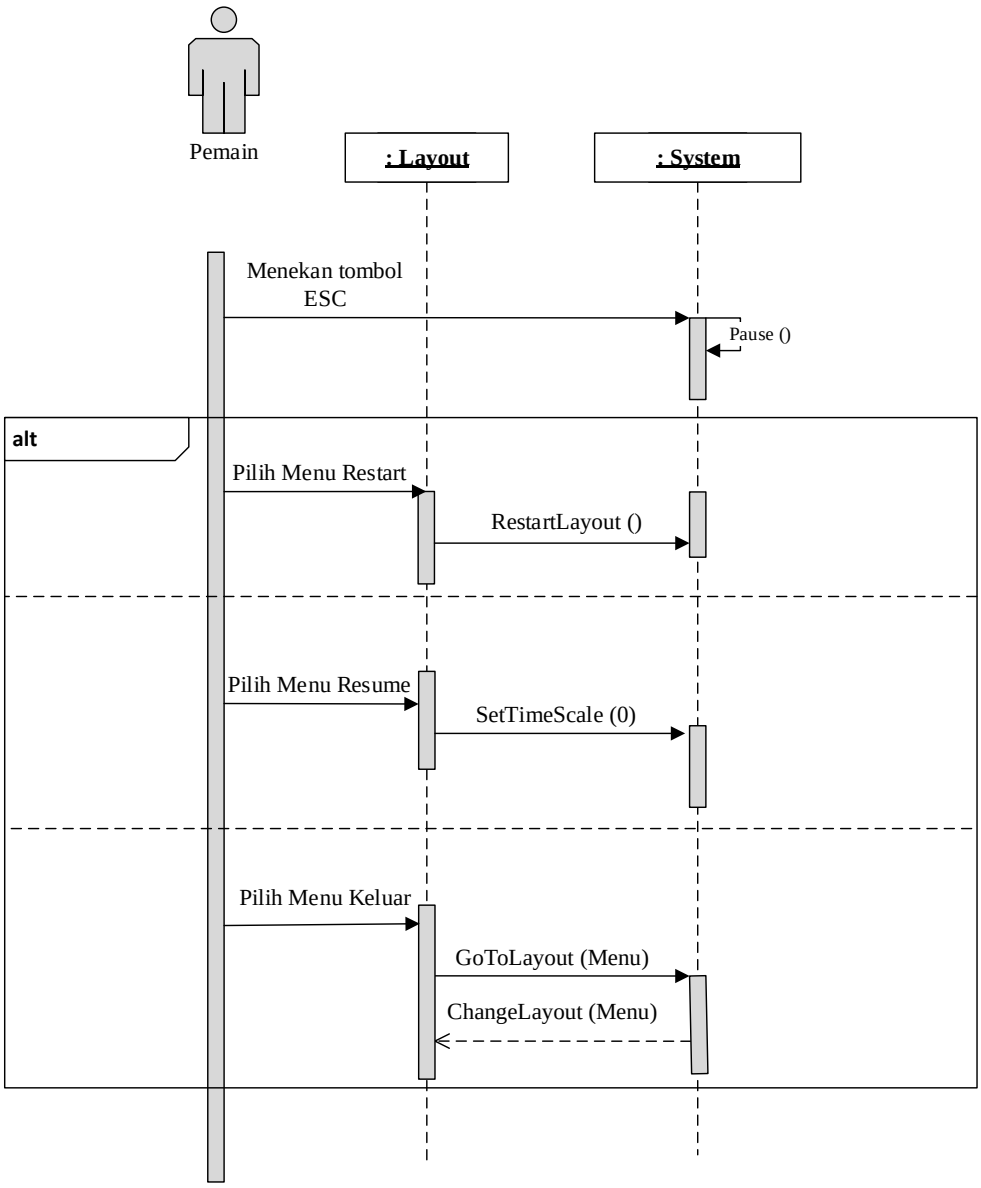
Gambar 3.23 Sequence Diagram Mengambil Keris

6. *Sequence Diagram* Melakukan Pertarungan



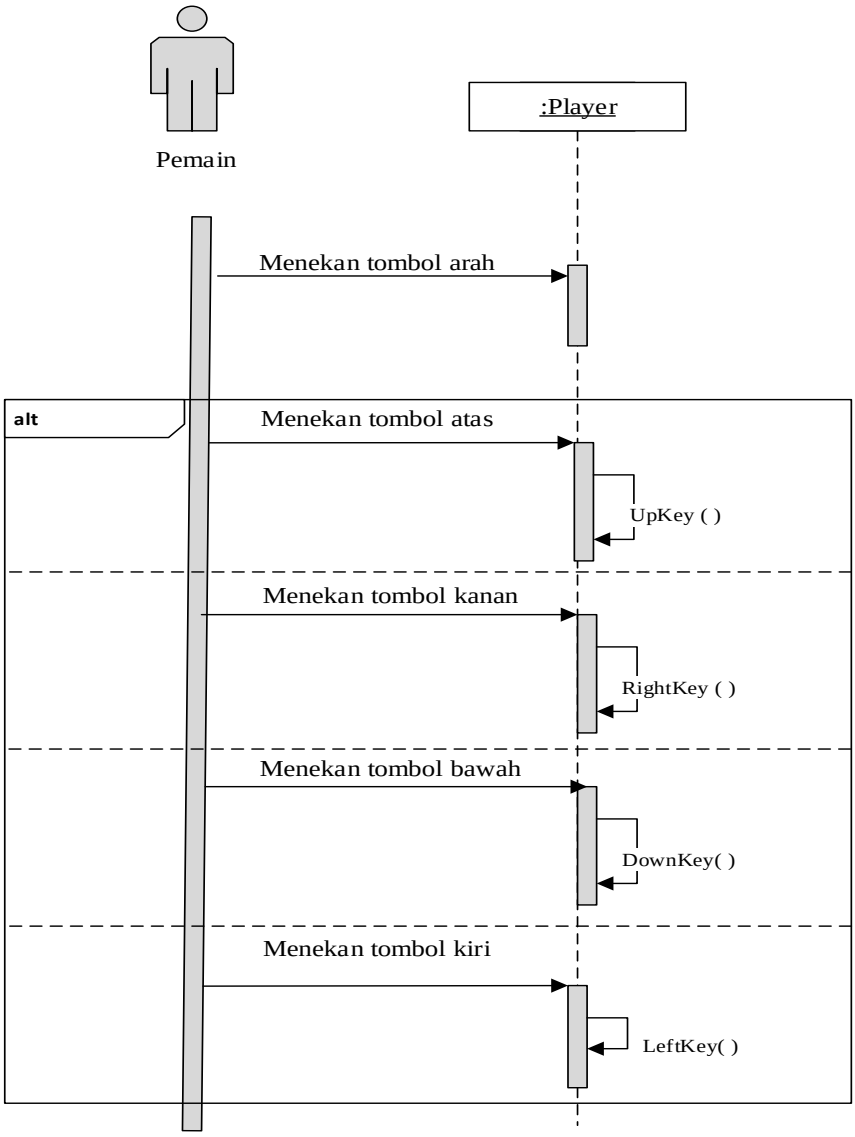
Gambar 3.24 Sequence Diagram Melakukan Pertarungan

7. Sequence Diagram Menggunakan Menu Permainan



Gambar 3.25 Sequence Diagram Menggunakan Menu Permainan

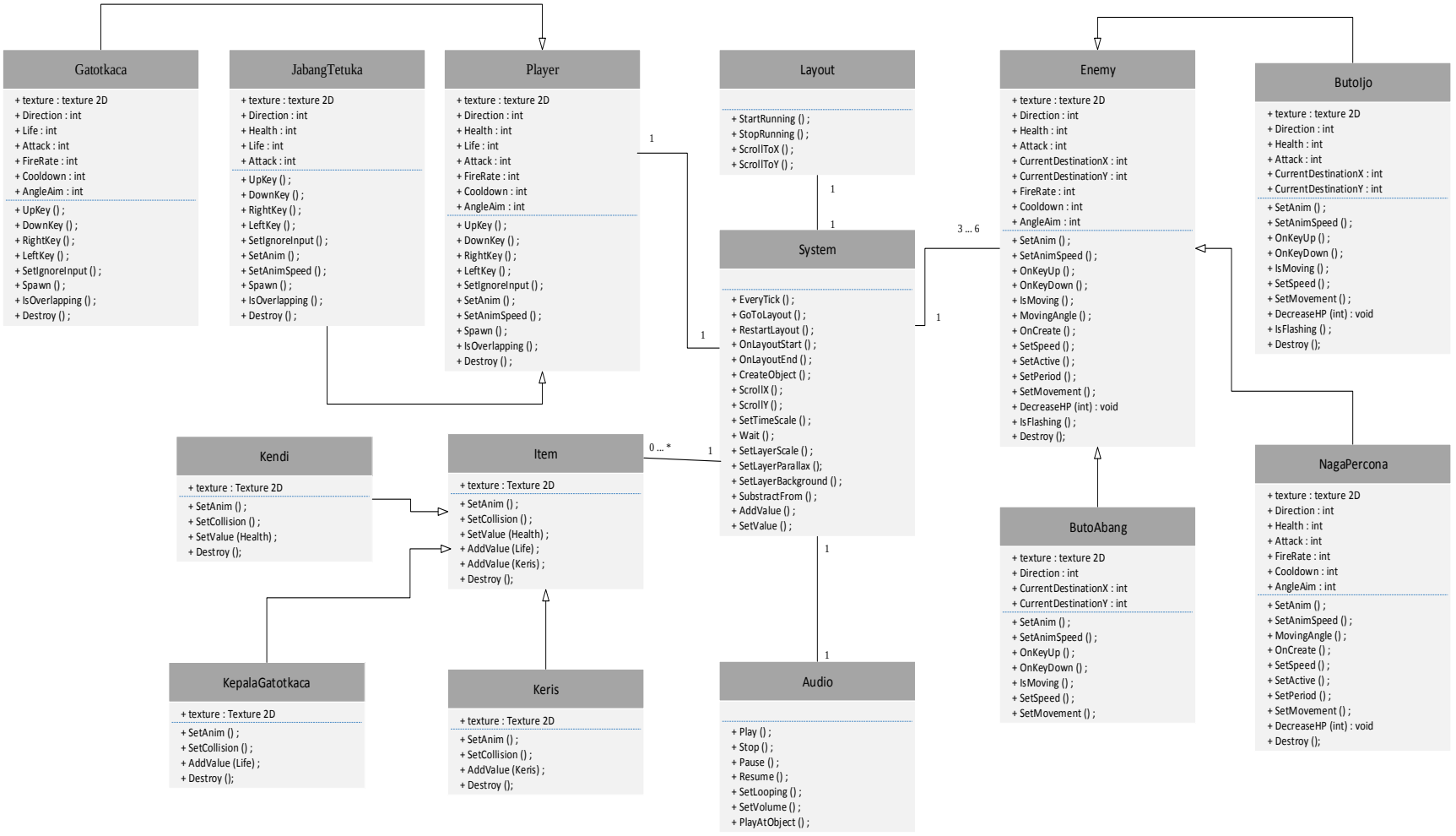
8. Sequence Diagram Menggerakkan Karakter



Gambar 3.26 Sequence Diagram Menggerakkan Karakter

3.1.6.8. *Class Diagram*

Class Diagram menggambarkan keadaan suatu sistem dengan menjelaskan keterhubungan antara suatu class dengan class yang lain yang terdapat pada sistem tersebut. Class Diagram dalam game Legenda Gatotkaca adalah sebagai berikut:



Gambar 3.27 Class Diagram Game Legenda Gatotkaca

3.2. Perancangan Sistem

Perancangan sistem menjelaskan tentang rancangan game yang dibangun dimana perancangan ini meliputi perancangan menu, antarmuka, pesan (dialog), jaringan semantik dan metode yang digunakan.

3.2.1. Perancangan Komponen *Game*

Perancangan komponen permainan menjelaskan tentang rancangan komponen dalam game Legenda Gatotkaca yang mana terdiri dari karakter pemain, musuh, *item* serta *storyboard*.

3.2.1.1. Karakter Pemain

Karakter pemain dalam game Legenda Gatotkaca beserta keterangan serta status (kondisi) awalnya dapat dilihat pada tabel 3.20 berikut :

Tabel 3.20 Karakter Pemain

No	Nama	Gambar	Keterangan	Status Awal
1.	Jabang Tetuka		Anak dari Bima, yang merupakan karakter dari Gatotkaca sewaktu masih kecil.	HP : 250 ATK : 60 DEF : 50
2.	Gatotkaca		Seorang ksatria sakti yang terkenal dengan sebutan otot kawat tulang besi.	HP : 300 ATK : 80 DEF : 70

Karakter pemain tersebut memiliki kondisi atau biasa disebut dengan istilah status yang menggambarkan keadaan karakter tersebut. Status karakter pemain dalam game Legenda Gatotkaca dapat dilihat pada tabel 3.21 berikut :

Tabel 3.21 Status Awal Karakter Pemain

No.	Nama Status	Keterangan
1.	HP (*Health Point*)	HP adalah poin yang menentukan berapa banyak kesakitan yang dapat diterima oleh karakter
2.	ATK (*Attack Power*)	ATK menentukan seberapa kuat serangan yang dimiliki oleh karakter
3.	DEF (*Defence Power*)	DEF menentukan seberapa kuat pertahanan yang dimiliki karakter dalam menghadapi serangan lawan

3.2.1.2. Musuh

Musuh dalam game Legenda Gatotkaca beserta kemampuannya dapat dilihat pada tabel berikut :

Tabel 3.22 Karakter Musuh

No.	Nama	Gambar	Kemampuan
1.	Buto Ijo		HP : 80 ATK : 30 DEF : 20
2.	Buto Abang		HP : 100 ATK : 50 DEF : 40
3.	Naga Percona		HP : 500 ATK : 100 DEF : 90

3.2.1.3. *Item*

Dalam game Legenda Gatotkaca pemain dapat menemukan item penambah darah dan penambah nyawa. Item yang ada di dalam game dapat dilihat pada tabel berikut :

Tabel 3.23 Item

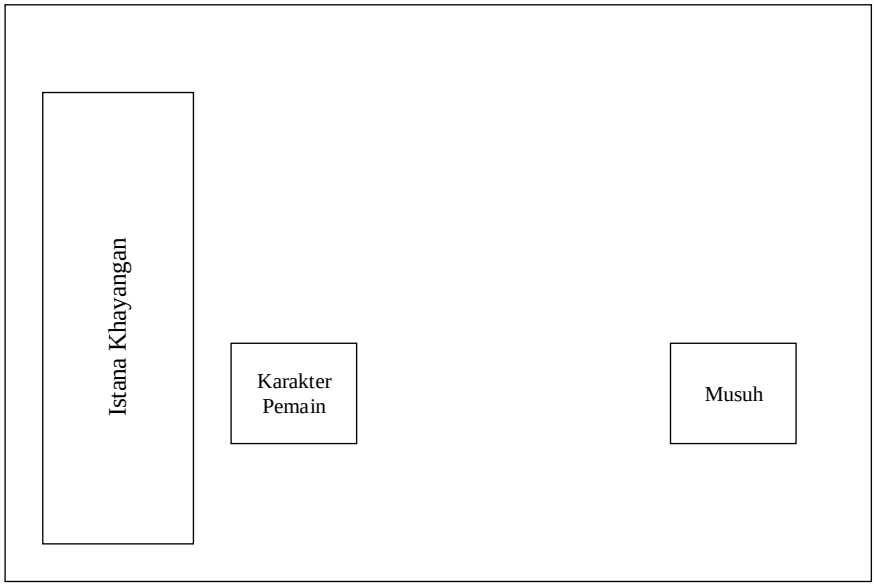
No.	Nama *Item*	Gambar	Keterangan
1.	Kendi		Memulihkan darah pemain menjadi penuh.
2.	Kepala Gatotkaca		Menambahkan 1 nyawa kepada pemain.
3.	Keris		Sebagai syarat untuk melewati level 2.

3.2.1.4. *Storyboard*

Story Board merupakan gambaran sketsa permainan secara umum yang disusun berdasarkan cerita atau *storyline* game. Berikut *storyboard* untuk game Legenda Gatotkaca :

1. Level 1

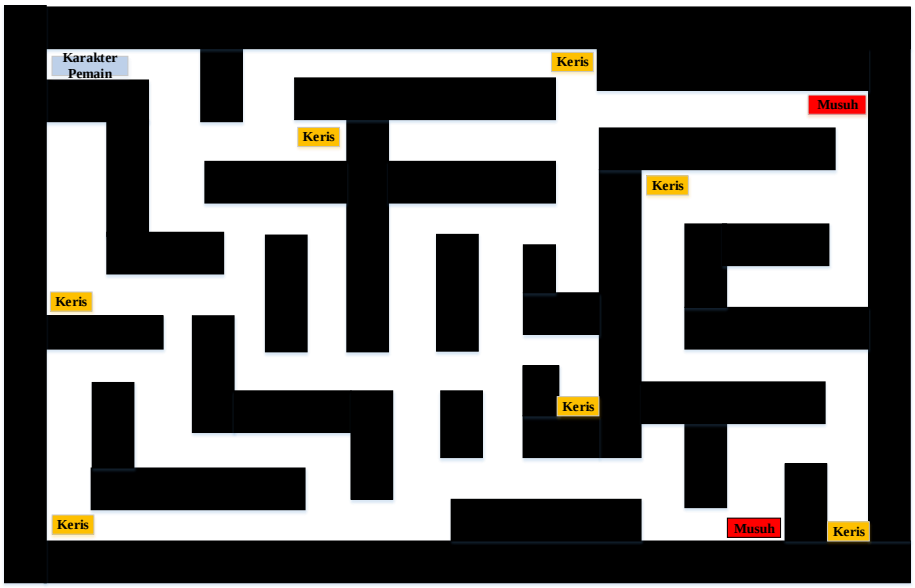
   Level 1 berlatar di Khayangan. Karakter pemain akan bertarung melawan pasukan Kerajaan Trabelasuket.



Gambar 3.28 Storyboard Level 1

2. Level 2

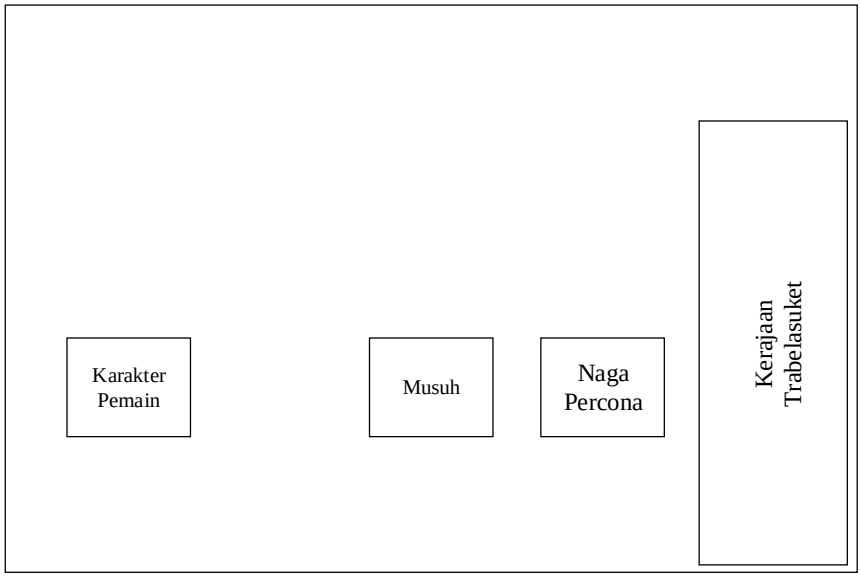
   Level 2 berlatar di Kawah Candradimuka. Karakter pemain mengumpulkan 7 keris pusaka dan menghindari musuh untuk keluar dari Kawah Candradimuka.



Gambar 3.29 Storyboard Level 2

3. Level 3

Level 3 berlatar di Kerajaan Trabelasuket. Karakter pemain melakukan serangan balik dan melawan Naga Percona.

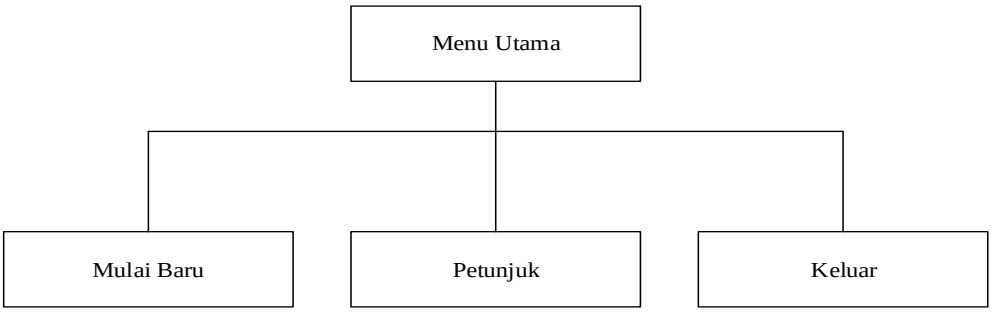


Gambar 3.30 Storyboard Level 3

3.2.2. Perancangan Struktur Menu

1. Menu Utama

Menu digunakan oleh pemain (user) untuk berinteraksi dengan sistem. Menu yang dipilih akan menentukan respon dari sistem.

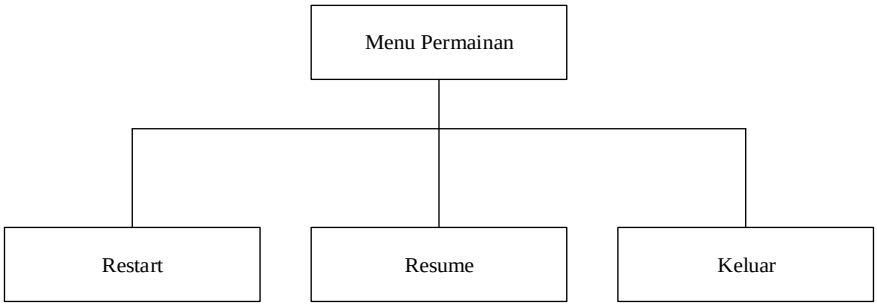


Gambar 3.31 Struktur Menu Utama

1. Mulai Baru : Memulai permainan dari awal
2. Petunjuk : Menampilkan petunjuk permainan
3. Keluar : Mengakhiri game (menutup aplikasi)

2. Menu Permainan

Menu permainan dapat diakses oleh pemain saat sedang memainkan game. Perancangan menu permainan adalah sebagai berikut :



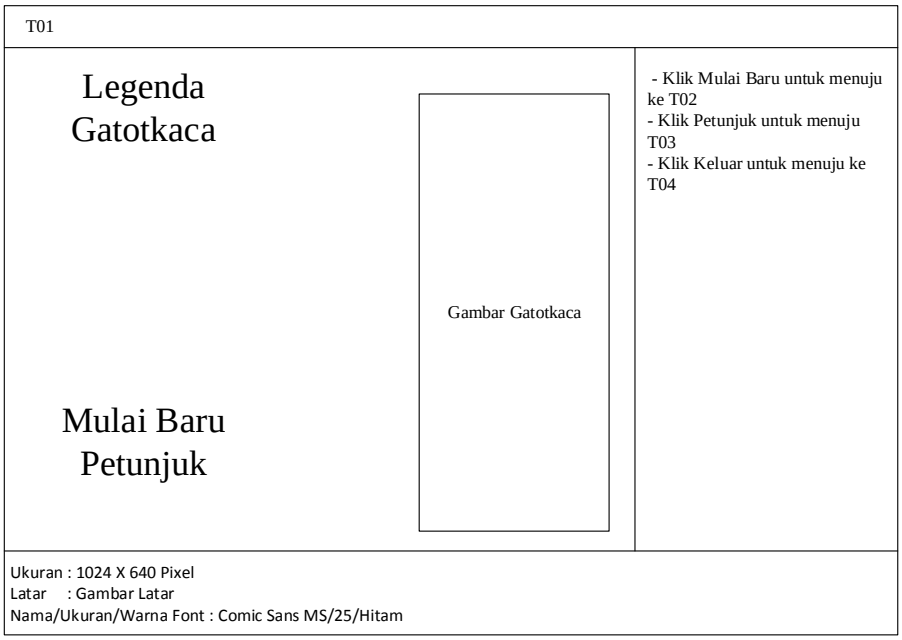
Gambar 3.32 Struktur Menu Permainan

1. Restart : Untuk mengulangi permainan dari awal
2. Resume : Untuk melanjutkan permainan yang sedang berjalan/sedang di pause.
3. Keluar : Untuk keluar dari permainan dan kembali ke menu utama.

3.2.3. Perancangan Antarmuka

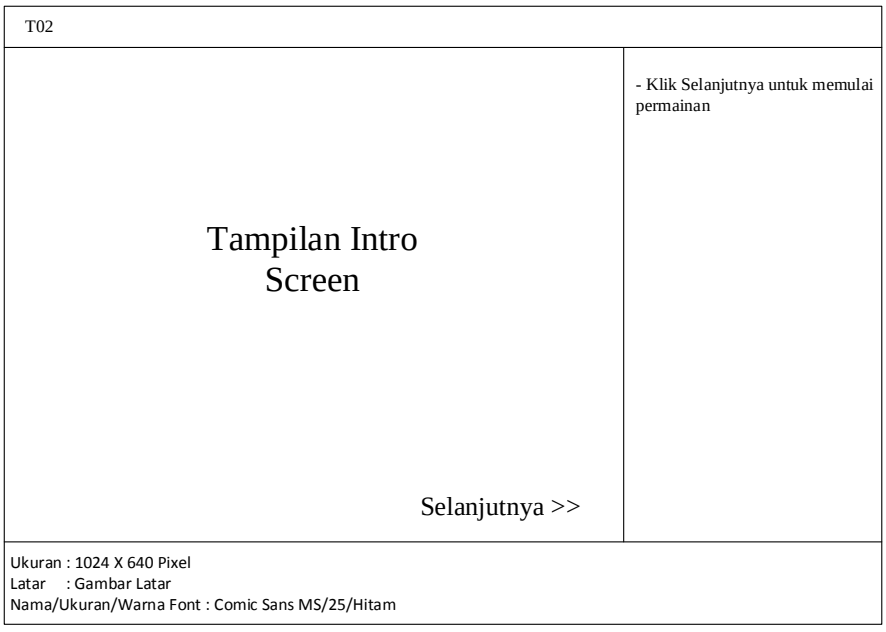
Antarmuka merupakan tampilan sistem yang disajikan untuk pengguna. Berikut antarmuka yang ada dalam game Legenda Gatotkaca :

1. Antarmuka Menu Utama



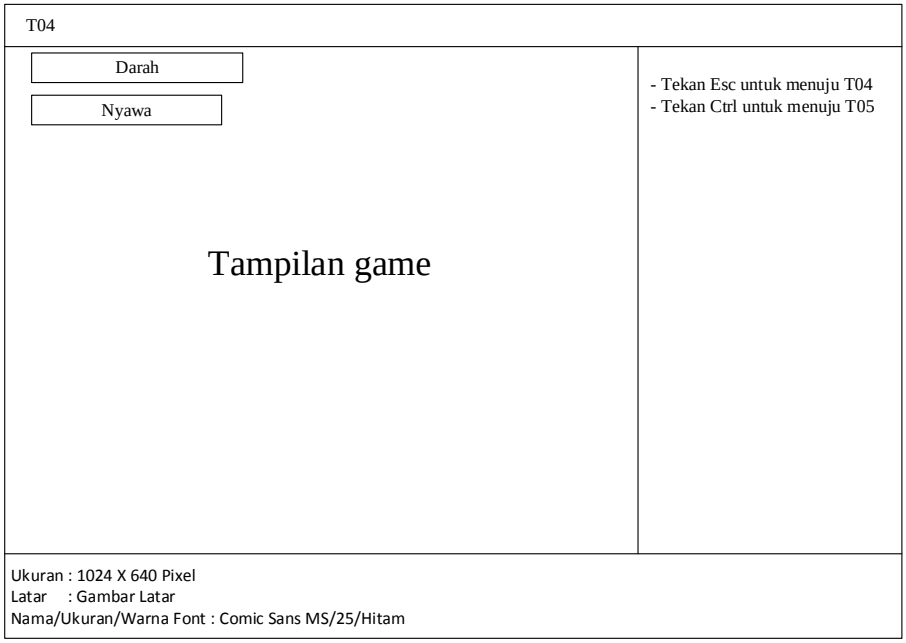
Gambar 3.33 Antarmuka Menu Utama

2. Antarmuka Intro Game



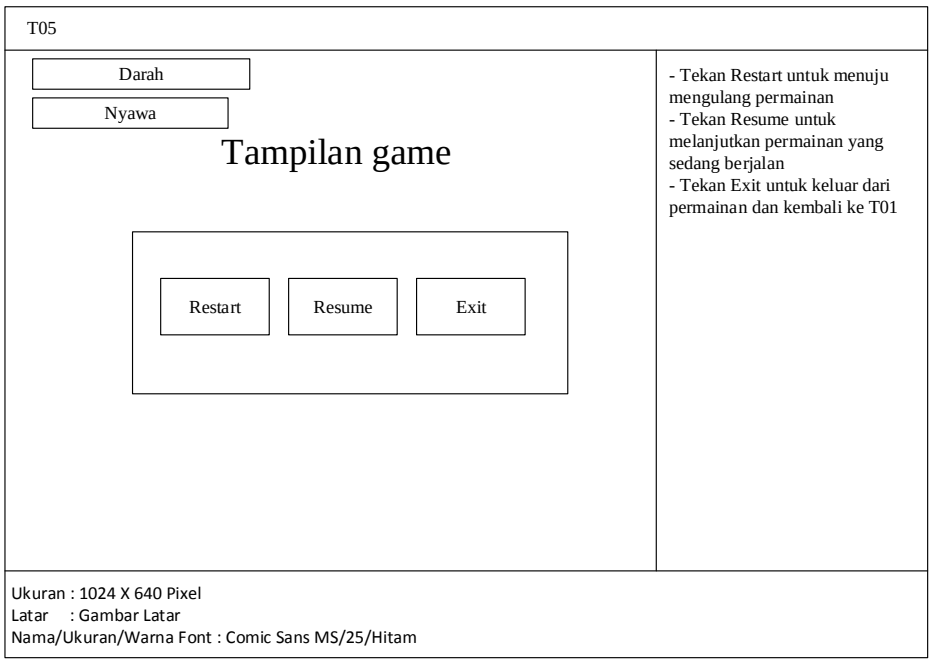
Gambar 3.34 Antarmuka Intro Game

3. Antarmuka Tampilan Game



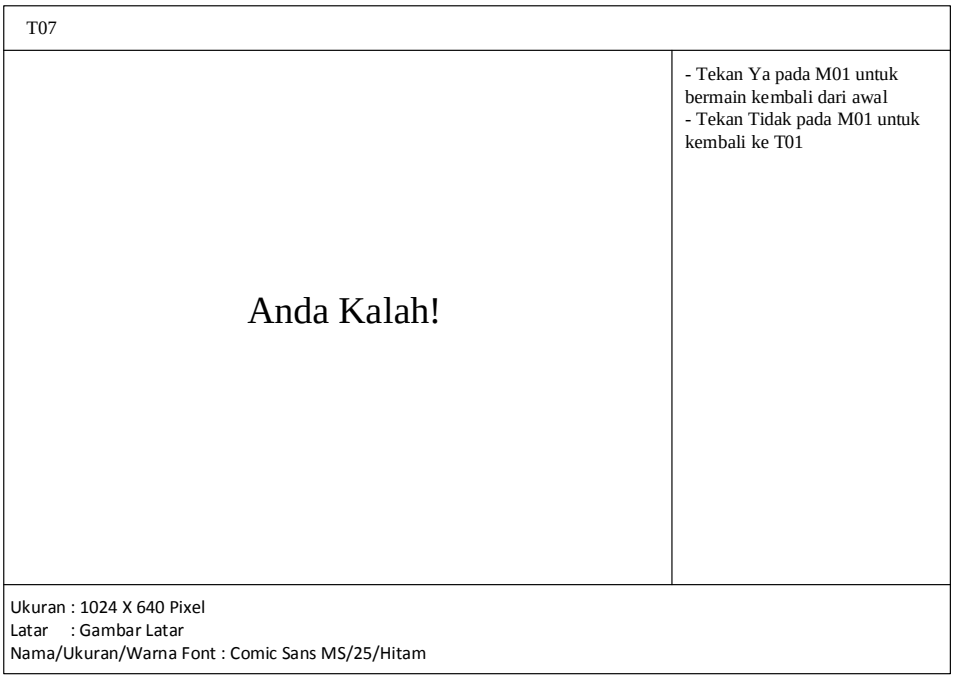
Gambar 3.35 Antarmuka Tampilan Game

4. Antarmuka Menu Permainan



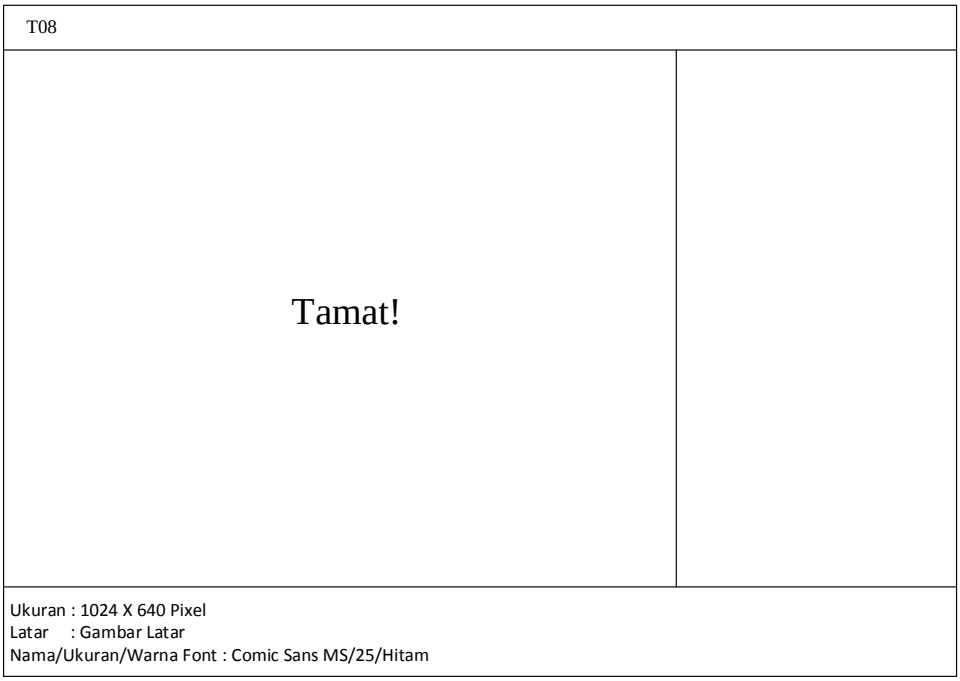
Gambar 3.36 Antarmuka Menu Permainan

5. **Antarmuka Game Over**



Gambar 3.37 Antarmuka Game Over

6. **Antarmuka Tamat**

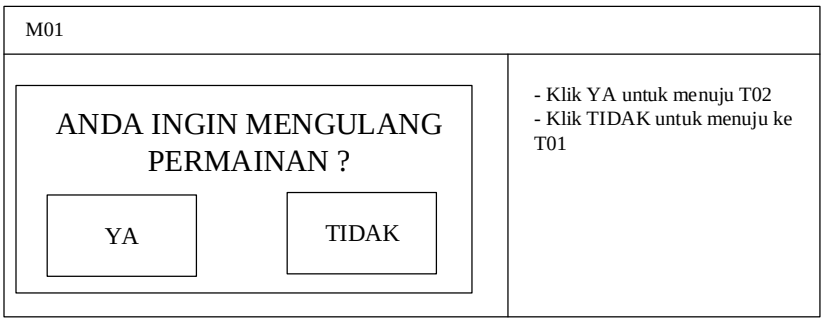


Gambar 3.38 Antarmuka Tamat

3.2.4. Perancangan Pesan

Pesan digunakan untuk memberi pemberitahuan, peringatan atau untuk meminta konfirmasi dari suatu aksi. Pesan dalam game Legenda Gatotkaca adalah sebagai berikut :

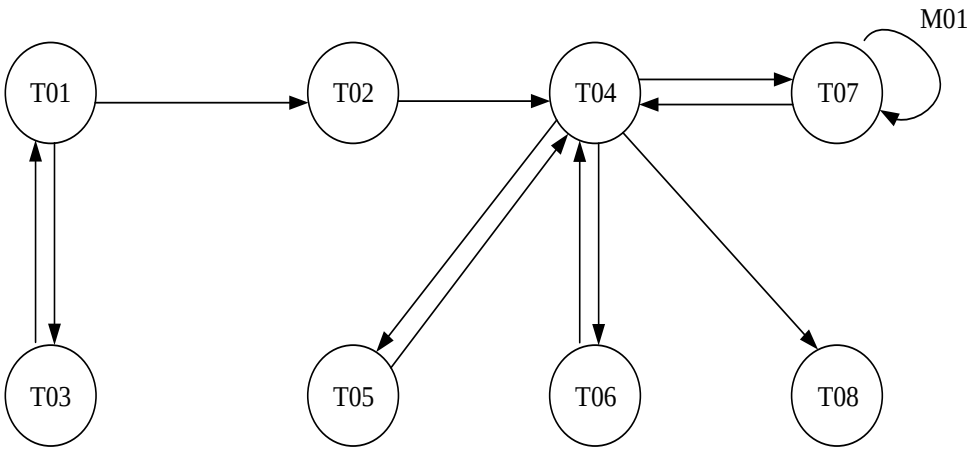
1. Pesan Konfirmasi Mengulang Permainan



Gambar 3.39 Pesan Konfirmasi Mengulang Permainan

3.2.5. Jaringan Semantik

Jaringan semantik merupakan jaringan data yang menunjukan hubungan antar berbagai objek. Gambar 3.39 menunjukkan jaringan semantik dari game yang dibangun.



Gambar 3.40 Jaringan Semantik