BAB 3

ANALISIS DAN PERANCANGAN SISTEM

3.1 Analisis Sistem

Analisis sistem ini merupakan cakupan dari analisis yang berhubungan dengan *game* yang akan dibangun diantaranya analisis masalah terhadap *game* yang sejenis, analisis kebutuhan fungsional, dan analisis non fungsional.

3.1.1 Analisis Masalah

Game dengan *genre turn base strategy* sudah tidak asing lagi dalam permainan game, perkembangannya tidak terlalu drastis namun *genre* ini tetap menarik untuk dimainkan karena dapat melatih otak untuk menentukan taktik atau strategi bermain. Asal mula *game* dengan *genre* ini mengadopsi *gameplay* dari permainan yang lama yaitu *game Chess* atau permainan catur. Saat ini *game turn based strategy* sudah banyak disebarluaskan, seperti *game Front mission, super robot wars, final fantasy tactics, heroes of might and magic, master of orion, tactics core.*

Dalam pembahasan, penelitian ini akan membahas *game* dengan *genre turn based strategy*. *Game* yang menjadi acuan disini ialah *game Tactic core*, secara dasar *gameplay game Tactic core* serupa dengan *gameplay game the kingdom of* pandawa yang akan dibangun, akan tetapi pada *game the kingdom of* pandawa akan ditambahkan elemen game yang dapat membuat skenario game menjadi lebih menarik, sedangkan dalam *story game* dari *game the kingdom of* pandawa akan diterapkan karakter-karakter pandawa dari cerita Mahabharata untuk memperjelas alur cerita game. Berdasarkan analisis masalah ini, *game Tactic core* terdapat beberapa masalah yang menjadi acuan untuk membangun *game the kingdom of* pandawa yaitu:

1. *Gameplay* pada permainan *game* kurang menarik karena murni mengadopsi *gameplay* permainan *game Chess* (catur) dari *battle system game*-nya.

2. Skenario *game* kurang menarik karena pada karakter tidak mempunyai karakter utama yang berasal dari cerita *game*-nya.
3. Karena mengadopsi *gameplay game Chess* di dalam permainan *game* karakter hanya dapat bergerak lalu menyerang, tidak terdapat *quest game* atau teka-teki di dalam *game*-nya.
4. Misi *game* kurang menarik karena hanya menyelesaikan satu level permainan.

Hal-hal ini dibuktikan dari situs www.gtds.net bahwa komentar untuk *game Tactic Core*.



Gambar 3.1 *Tactic Core Comment*[4].

3.1.1.1 Deskripsi Sistem pada *Game Tactic Core*

Game Tactic Core merupakan permainan ber-*genre* strategi yang dibangun menggunakan *flash.* Sesuai dengan namanya *game* ini memiliki permainan yang dimainkan menggunakan taktik untuk melawan karakter-karakter musuh. Permainan ini dimainkan dengan cara memindahkan karakter dari *base* awal hingga *base* yang jaraknya dapat menyerang karakter musuh secara bergiliran. Karakter yang terdapat di dalam permainan ini diantaranya adalah *archery* (pemanah), *magic* (penyihir) dan *swordmanship* (ahli pedang).



Gambar 3.2 *Screenshoo*t tampilan permainan *game Tactic Core*[30].

3.1.1.2 *Gameplay* pada *Game Tactic Core*

Pada *game Tactic Core* ini, pemain harus mengalahkan karakter-karakter musuh dengan karakter-karakter yang dimilikinya dengan cara memindahkan tiap karakter dari *base* awal sampai *base* yang jaraknya dapat menyerang musuh menggunakan alat *input* secara bergiliran. Dalam *game* ini terdapat arena permainan bentuk *base* yang berupa petak-petak. Karakter di dalam permainan mempunyai kemampuan kontrol seperti *move* (pergerakan), *attack* (serangan) dan *healthpoint* (batas kehidupan) sebagai acuan untuk karakter bertahan di dalam area permainan. Apabila pemain dapat mengalahkan seluruh karakter musuh dalam satu kali permainan maka pemain dinyatakan menang dan apabila sebaliknya pemain dapat memainkan kembali dari awal permainan.

3.1.1.3 *Elemen Game*

Elemen *game* yang terdapat dalam permainan game *Tactic Core* diantaranya adalah sebagai berikut :

1. Arena Permainan

   Pada permainan game *Tactic Core* mempunyai arena permainan yang dibentuk dengan *isometric map*. Arena permainan ini mempunyai gambaran layaknya di dalam sebuah labirin yang dipenuhi dinding-dinding batu.

2. Karakter pemain

   Karakter pemain adalah entitas yang akan dikendalikan oleh pengguna di dalam arena permainan. Karakter pemain mempunyai *state* yang akan digunakan dalam permainan diantaranya *move* (melangkah)yang akan digunakan karakter pemain untuk memindahkan karakternya dalam arena, *attack* (menyerang) yang akan digunakan karakter pemain untuk menyerang karakter musuh, *wait* (menunggu) yang akan digunakan karakter pemain untuk melewati atau mengakhiri giliran karakter. Selain mempunyai *state*, karakter pemain juga mempunyai *health point* (batas kehidupan) yang digunakan karakter pemain untuk bertahan hidup dalam arena permainan dan *attack point* (daya serang) yang digunakan karakter pemain untuk memberikan daya serang terhadap karakter musuh. Tujuan dari karakter pemain adalah mengalahkan semua karakter musuh dalam permainan.

3. Karakter Musuh

   Karakter musuh dalam *Tactic Core* adalah entitas yang akan melawan karakter pemain dalam arena permainan. Sama halnya dengan karakter pemain, karakter musuh pun mempunyai *state* yang sama, hanya saja karakter musuh mempunyai *attack point* (daya serang) dan *health point* (batas kehidupan) yang lebih tinggi.

4. Menu kontrol

   Menu kontrol adalah entitas yang berfungsi sebagai media yang menentukan setiap karakter untuk bergerak (*move*), menyerang (*attack*), melewati giliran (*wait*). Menu kontrol ini akan muncul ketika karakter mendapatkan giliran.

5. *Life Point*

   Life Point adalah entitas yang berisi *health point* (batas kehidupan) yang dimiliki oleh tiap karakter pemain dan karakter musuh. *Life point* pada tiap karakter berbeda-beda. *Life point* juga menjadi entitas terpenting dalam permainan, karena menentukan tiap karakter dapat bertahan dalam arena permainan.

3.1.1.4 Skenario *Game*

Skenario dalam *game Tactic Core* disusun oleh beberapa hal sebagai berikut:

A. Alur Permainan

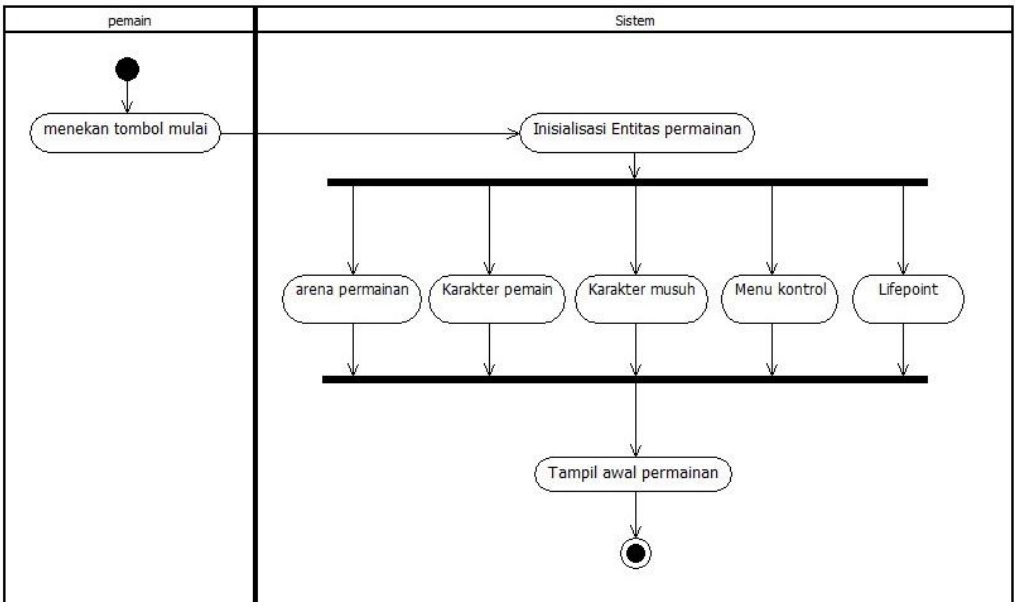
1. Alur Inisialisasi *Game Tactic Core*

Alur inisialisasi pada *game Tactic Core* dapat dilihat pada Tabel 3.1 berikut:

Tabel 3.1 Alur inisialisasi *Game Tactic Core*

No.	Alur Sistem
1	Pemain menekan tombol mulai permainan
2	Sistem akan menginisialisasi seluruh entitas permainan yang diantaranya adalah arena permainan, karakter pemain, menu kontrol, karakter musuh, *life point*.
3	Sistem akan menampilkan seluruh entitas permainan.

Adapun hasil alur inisialisasi permainan dari *game Tactic Core* disusun dari beberapa aktivitas yang akan digambarkan melalui *activity diagram* berikut ini :



Gambar 3.3 Alur inisialisasi permainan pada *game Tactic Core*

Pada gambar 3.3 menjelaskan aktivitas untuk menampilkan tampilan awal permainan, ketika pemain menekan tombol mulai maka permainan pun akan menginisialisasikan semua entitas permainan diantaranya arena permainan, karakter pemain, karakter musuh, menu kontrol, *lifepoint*, dan permainan pun akan menginisialisasikan posisi setiap entitas permainan sebelum ditampilkan.

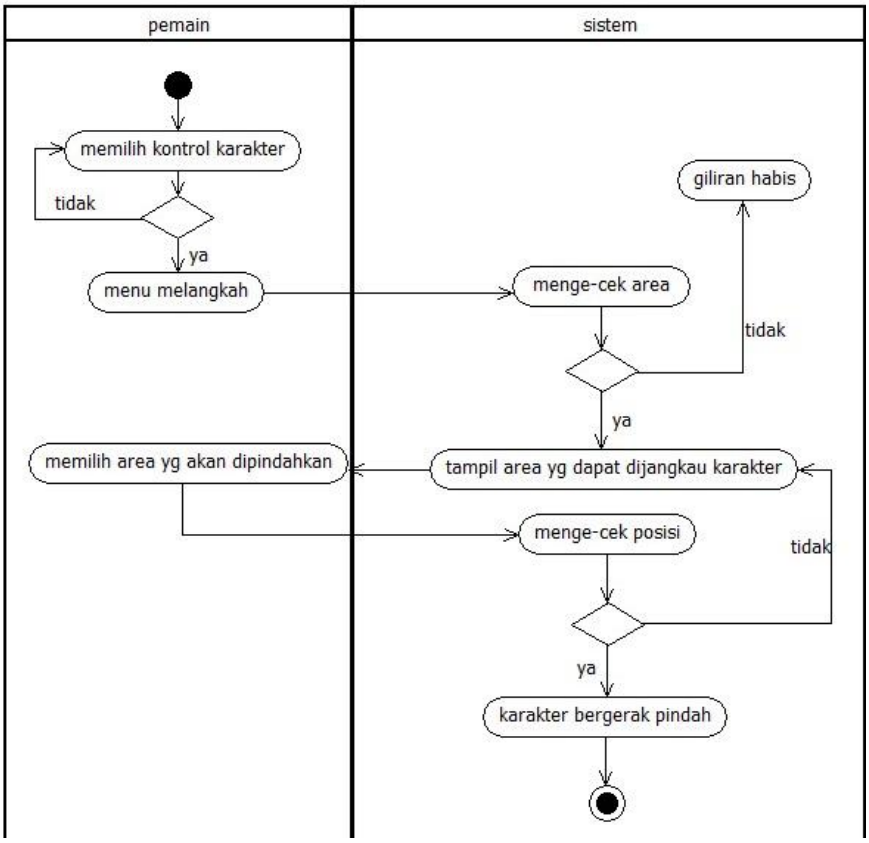
2. Alur kontrol melangkah *Game Tactic Core*

Alur kontrol melangkah pada *game Tactic Core* dapat dilihat pada Tabel 3.2 berikut:

Tabel 3.2 Alur kontrol melangkah *Game Tactic Core*

No.	Alur Sistem
1	Pemain dapat mengontrol permainan melalui input *mouse* sebagai penunjuk menu kontrol.
2	Pemain memilih menu kontrol *move* untuk menggerakkan karakter.
3	Sistem akan mengecek area.
4	Sistem akan menampilkan area yang dapat dijangkau
5	Pemain dapat memilih area untuk memindahkan karakter
6	Sistem akan memilih area yang akan dipindahkan.
7	Karakter akan bergerak pindah ke area yang dipilih.

Adapun hasil alur kontrol melangkah dari *game Tactic Core* disusun dari beberapa aktivitas yang akan digambarkan melalui *activity diagram* berikut ini :



Gambar 3.4 Alur kontrol melangkah pada *game Tactic Core*

Pada gambar 3.4 menjelaskan aktivitas untuk mengontrol pergerakan di dalam permainan. Ketika pemain ingin menggerakkan karakter maka pemain harus memilih menu kontrol melangkah terlebih dahulu. Setelah memilih menu kontrol melangkah maka permainan pun akan menge-cek area untuk menampilkan area yang dapat dijangkau oleh karakter. Setelah area ditampilkan pemain masih harus memilih area yang akan dipindahkan agar karakter bergerak pindah. Setelah posisi telah di cek maka akan dilakukan pemindahan posisi baru karakter.

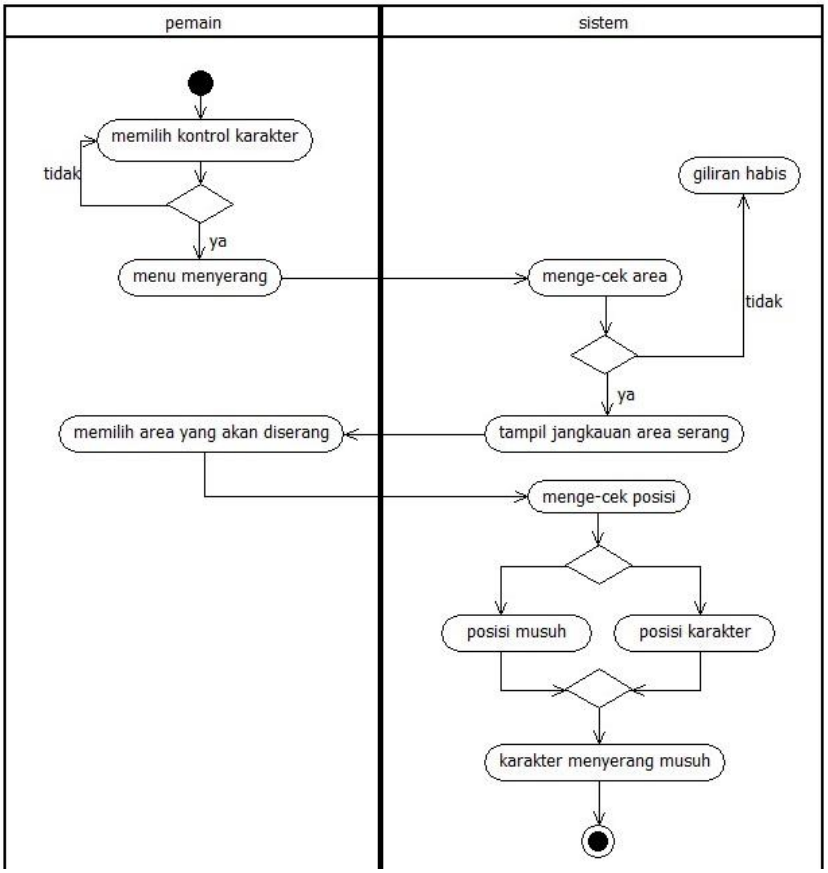
3. Alur kontrol menyerang *Game Tactic Core*

Alur kontrol menyerang pada *game Tactic Core* dapat dilihat pada Tabel 3.3 berikut:

Tabel 3.3 Alur kontrol menyerang *Game Tactic Core*

No.	Alur Sistem
1	Pemain dapat mengontrol permainan melalui input *mouse* sebagai penunjuk menu kontrol.
2	Pemain memilih menu kontrol *attack* untuk menggerakkan karakter.
3	Sistem akan mengecek area.
4	Sistem akan menampilkan jangkauan area serang
5	Pemain dapat memilih area untuk menyerang musuh
5	Sistem akan menge-cek posisi
6	Karakter akan menyerang musuh

Adapun hasil alur kontrol menyerang dari *game Tactic Core* disusun dari beberapa aktivitas yang akan digambarkan melalui *activity diagram* berikut ini :



Gambar 3.5 Alur kontrol menyerang pada *game Tactic Core*

Pada gambar 3.5 menjelaskan aktivitas untuk mengontrol serangan karakter kepada musuh di dalam permainan. Ketika pemain ingin menyerang musuh maka pemain harus memilih menu kontrol menyerang terlebih dahulu. Setelah pemain memilih menu kontrol menyerang, permainan pun akan menge-cek area, lalu permainan pun mulai menampilkan jangkauan area serang karakter. Setelah jangkauan area serang tampil maka pemain harus memilih area mana yang akan diserang, ketika area telah dipilih maka sistem akan menge-cek posisi serang yang ditempati oleh musuh maka karakter akan menyerang musuh dan ketika area serang tidak ditempati oleh musuh maka akan tetap menampilkan jangkauan area serang karakter.

4. Alur kontrol melewati giliran *Game Tactic Core*

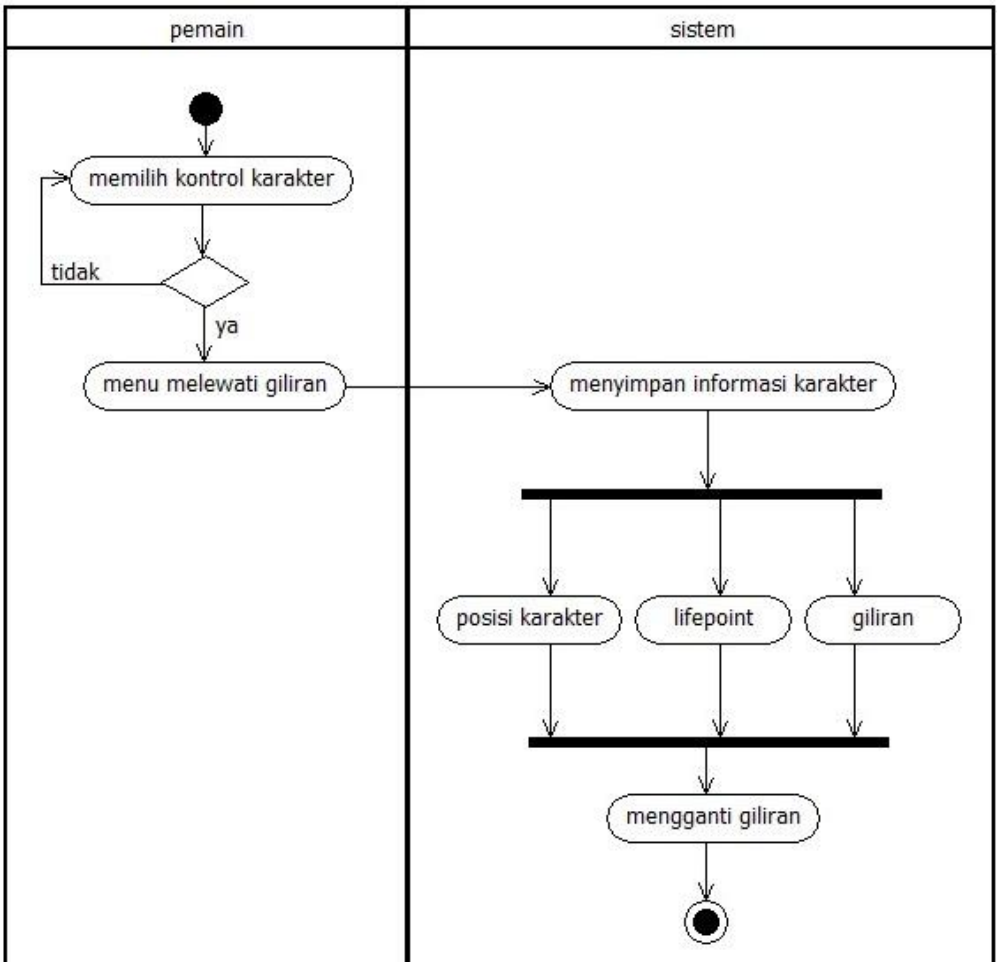
Alur kontrol melewati giliran *game Tactic Core* dapat dilihat pada Tabel 3.4 berikut:

Tabel 3.4 Alur kontrol melewati giliran *Game Tactic Core*

No.	Alur Sistem
1	Pemain menekan tombol mulai permainan
2	Sistem akan menginisialisasi seluruh entitas permainan yang diantaranya adalah arena permainan, karakter pemain, menu kontrol, karakter musuh, *life point*.
3	Sistem akan memposisikan seluruh entitas permainan.
4	Sistem akan menampilkan seluruh entitas permainan.
5	Sistem memberikan menu kontrol kepada setiap karakter.
6	Pemain dapat mengontrol permainan melalui input *mouse* sebagai penunjuk menu kontrol.
7	Tiap karakter dapat bergerak ketika pemain memilih menu kontrol *move*.
8	Tiap karakter dapat menyerang ketika pemain memilih menu kontrol *attack.*
9	Tiap karakter dapat melewati giliran ketika pemain memilih menu kontrol *wait*.
10	Tiap musuh dapat kehilangan batas kehidupannya dari entitas *life point*

	ketika dapat terkena serangan karakter pemain.
11	Sistem akan meng-*update life point* karakter ketika terjadi serangan dan ketika mengganti giliran main.
12	Sistem akan meng-*update* posisi karakter ketika terjadi pergerakan dan ketka mengganti giliran main.
13	Sistem akan meng-*update* giliran pemain dan musuh ketika seluruh karakter telah dimainkan.

Adapun hasil alur kontrol melewati giliran dari *game Tactic Core* disusun dari beberapa aktivitas yang akan digambarkan melalui *activity diagram* berikut ini :



Gambar 3.6 Alur kontrol melewati giliran pada *game Tactic Core*

Pada gambar 3.6 menjelaskan aktivitas untuk mengontrol melewati giliran karakter di dalam permainan. Ketika pemain ingin melewati giliran maka pemain harus memilih menu melewati giliran terlebih dahulu, kemudian permainan akan menyimpan informasi karakter yang diantaranya adalah posisi, *lifepoint*, giliran. Setelah permainan menyimpan informasi karakter maka permainan pun mengganti giliran karakter berikutnya.

B. Misi Permainan

Misi permainan pada *game Tactic Core* dapat dilihat pada Tabel 3.5 berikut:

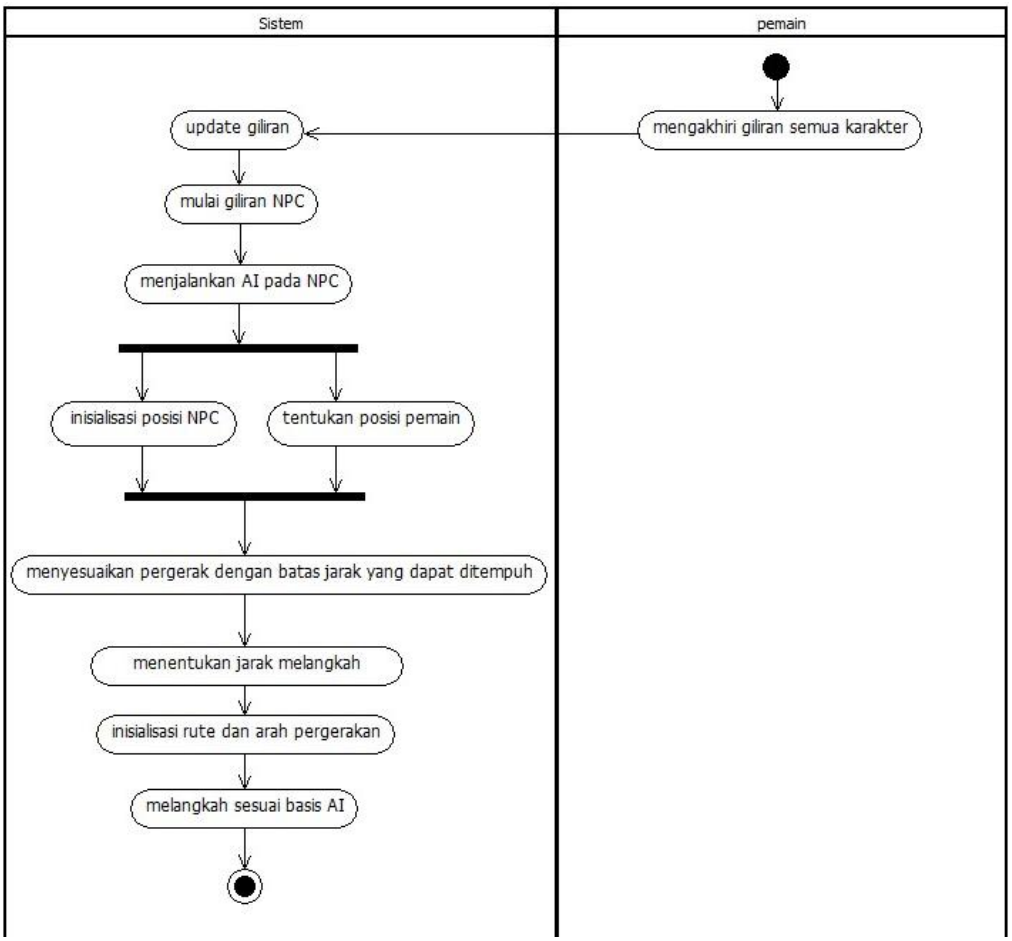
Tabel 3.5 Misi Permainan pada *game Tactic Core*

Misi Permainan
Pemain harus menyusun taktik dan mengalahkan semua karakter musuh untuk dapat menang dari musuh, dengan menggerakan karakter pemain dan menyerang semua karakter musuh. Ketika semua karakter musuh sudah terkalahkan maka permainan akan selesai karena permainan hanya mempunyai satu level.

3.1.1.5 Analisis Metode pada *Game Tactic Core*

Pada *game Tactic Core* terdapat objek yang diterapkan suatu metode tertentu sehingga objek dapat bergerak sendiri di arena permainan. Objek yang diterapkan metode tersebut dikenal juga dengan NPC (*Non-Playable Character*). NPC ditanamkan *artificial intelligence* melalui algoritma yang diberikan di dalam programnya sehingga NPC akan bereaksi mendekati pemain atau ketika pemain mendekati daerah musuh. Dengan menggunakan *game engine* dalam pembuatannya, AI dapat membuat NPC melacak posisi objek pemain dari posisi awal NPC menuju objek pemain.

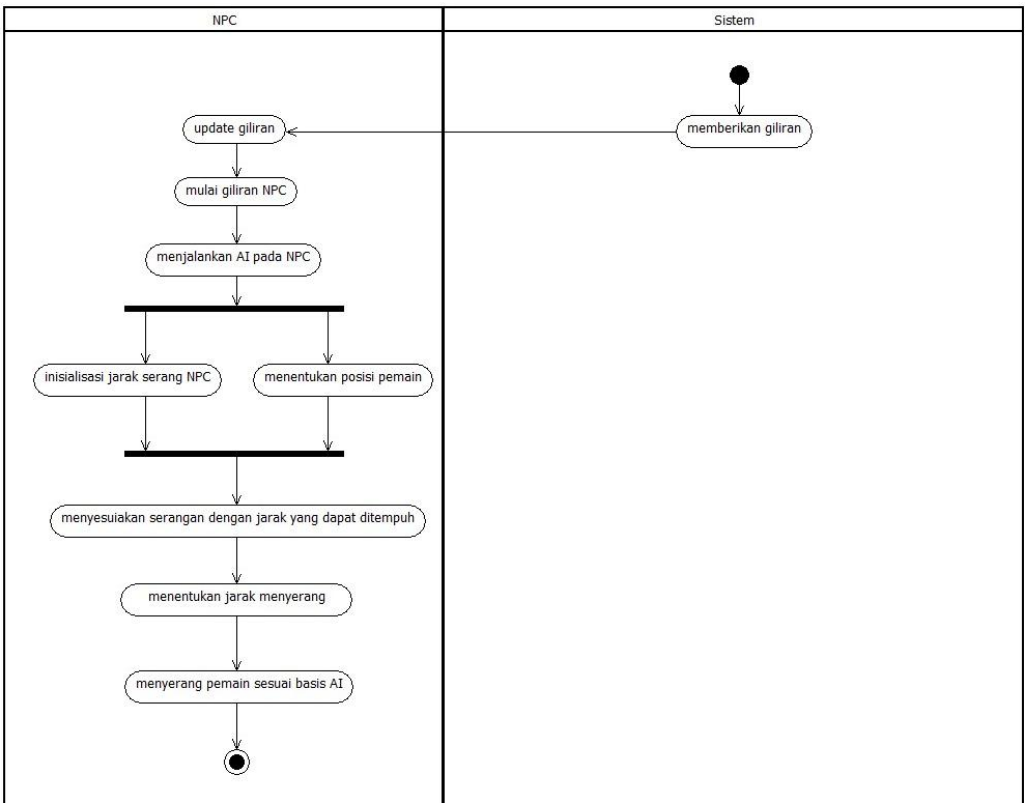
Adapun aktivitas metode AI pada melangkah NPC (*Non-Playable Character*) dapat dilihat sebagai berikut :



Gambar 3.7 Aktivitas melangkah NPC (*Non-Playable Character*)

Pada gambar 3.7 menjelaskan aktivitas melangkah NPC (*Non-Playable Character*) di dalam permainan. Aktivitas akan terjadi ketika giliran pemain sudah berakhir dan NPC sudah memulai gilirannya, AI *game* akan menginisialisasi posisi NPC yang merupakan titik awal dan menentukan posisi pemain sebagai titik tujuan. Setelah AI *game* melacak maka NPC akan mulai melangkah sesuai jarak yang dapat ditempuh NPC untuk mendekati pemain.

Adapun aktivitas metode AI pada menyerang NPC (*Non-Playable Character*) dapat dilihat sebagai berikut:



Gambar 3.8 Aktivitas menyerang NPC (*Non-Playable Character*)

Pada gambar 3.8 menjelaskan aktivitas menyerang NPC (*Non-Playable Character*) di dalam permainan. Aktivitas akan terjadi ketika giliran pemain sudah berakhir dan NPC sudah memulai gilirannya, AI *game* akan menginisialisasi jarak serangnya sebagai titik awal dan menentukan posisi pemain sebagai titik tujuan. Setelah menentukan jarak menyerang maka NPC akan mulai menyerang.

3.1.2 Analisis *Game The Kingdom of* Pandawa

Analisis *game* merupakan cakupan analisis dari *game* yang akan dibangun yang diantaranya adalah sebagai berikut :

3.1.2.1 Deskripsi *Game*

Game yang akan dibangun merupakan *game* ber-*genre strategy*. Karakter pada *game* ini diambil dari salah satu cerita heroik yaitu pandawa. Pada *game* ini terdapat 5 karakter pandawa sebagai karakter pemain yang diantaranya yudhistira, bima, arjuna, nakula, sadewa, dan karakter kurawa sebagai musuh yang

diantaranya abimanyu, drestadyumna, seta, kresna, adipati karna. *Game the kingdom of* pandawa merupakan *game* dengan *system single player*.

3.1.2.2 *Story Line*

Kerajaan hastinapura adalah kerajaan yang makmur dan sejahtera, hastinapura pun mempunyai raja yang gagah bernama Pandu Dewanata. Pandu dewanata memiliki seorang istri yang bernama Dewi kunti dan 5 orang anak yaitu yudhistira, bima, arjuna, nakula, dan sadewa. Menurut pewarisan jaman dahulu, penurunan tahta kerajaan akan diberikan kepada anak yang paling tua, dan pada saat pandawa masih kecil mereka ditinggal oleh ayahnya meninggal. Melihat kesempatan itu kurawa yang terlahir menjadi anak dari adik Pandu dewanata yaitu Drestaresta, iri akan penurunan kerajaan terhadap yudhistira. Setelah mendengar kabar seperti itu kurawa dan penasehat yang liciknya yaitu sengkuni menyiasati agar pandawa tidak dapat tahta kerajaan dan berusaha untuk mengusirnya dari hastinapura. Karena curangnya kurawa, pandawa pun berhasil diusir dari hastinapura. Pandawa pun terus berkelana sampai dimana yudhistira Mendengar hastinapura telah di pimpin oleh kurawa yang sangat licik dan tamak, perlahan-lahan pandawa pun bangkit dan bertekad untuk mengalahkan kurawa mengambil kerajaan hastinapura kembali.

3.1.2.3 *Gameplay*

Pada *game* ini pemain harus mengalahkan karakter kurawa sebagai musuh dengan memainkan karakter utama yaitu yudhistira. Dalam *game* ini tiap karakter dapat melangkah dan menyerang secara bergiliran, akan tetapi berbeda dengan karakter pemain yang dapat memunculkan karakter (*summon*) untuk membantu melawan karakter musuh di tengah permainan. Kemampuan memunculkan karakter (*summon*) pun terbatas karena pemain harus menempati dahulu kerajaan atau kastil dimana jumlah dari kastil yang ditempati menjadi batas karakter utama untuk memunculkan karakter (*summon*). Dalam permainan ini pemain mempunyai kendali untuk mempertahankan karakter utama di arena permainan yaitu *move* (melangkah), *attack* (menyerang), *summon* (memunculkan karakter) dan

healthpoint (batas kehidupan). Pada permainan ini terdapat elemen-elemen pembentuk *gameplay* permainan seperti berikut :

3.1.2.4 *Elemen Game*

Elemen permainan yang terdapat dalam permainan *game the kingdom of* pandawa diantaranya adalah sebagai berikut:

1. Arena permainan

   Pada arena permainan *game the kingdom of* pandawa terdapat 3 arena permainan sebagai medan pertempurannya. Arena pertempuran berbentuk kotak *hexagonal* yang digunakan sebagai tempat tiap karakter menempati dalam satu kotaknya. Di arena ini entitas pemain harus melawan musuh dari Kerajaan Kurawa.

2. Kastil

   Pada game ini terdapat kastil yang berfungsi sebagai jumlah *summon* karakter yang dapat pemain munculkan. Kastil menjadi media yang harus dikuasai, karena dapat membantu jumlah karakter yang akan bertempur.

3. Pemain (Pandawa)

   Pada awal *game* ini karakter utama pemain dari pandwa yang bernama Yudhistira. Ia dapat mengontrol pergerakan dari *base* ke *base* yang lain dan dapat memunculkan karakter Pandawa yang lain ke dalam arena permain. Pemain memiliki misi yaitu mengalahkan musuh dengan jumlah yang lebih banyak dari pemain. Pemain dapat menyeimbangkan jumlah dengan memunculkan karakter Pandawa yang lain ke dalam arena.

4. Musuh (*Non-Playable Character*)

   Pada *game* ini musuh terdiri dari 2 jenis yaitu prajurit Kurawa dan para Kurawa. Prajurit Kurawa akan menjaga *base* 1 dan 2 dan *base* 3 akan dihuni oleh *boss*Kurawa. *Boss* kurawa berbeda dengan prajurit Kurawa, mereka lebih sulit dikalahkan dikarenakan memiliki status *life point* yang lebih besar.

5. Menu Kontrol

   Pada game ini terdapat Menu kontrol yang berfungsi sebagai kendali permainan. Menu kontrol adalah media yang menentukan setiap karakter untuk bergerak (*move*), menyerang (*attack*), melewati giliran (*wait*). Menu kontrol ini akan muncul ketika karakter mendapatkan giliran.

6. *Life Point*

   Pada *game* ini terdapat *Life Point* yang berisi *health point* (batas kehidupan) yang dimiliki oleh tiap karakter pemain dan karakter musuh. *Life point* pada tiap karakter berbeda-beda. *Life point*menjadi entitas terpenting dalam permainan, karena menentukan karakter pemain dan musuh dapat bertahan dalam arena permainan.

3.1.2.5 Skenario *Game*

Skenario pada *game the kingdom of* pandawa tidak jauh berbeda dengan yang terdapat *game Tactics Core* namun dilakukan beberapa pengembangan skenario diantaranya :

A. Alur Permainan

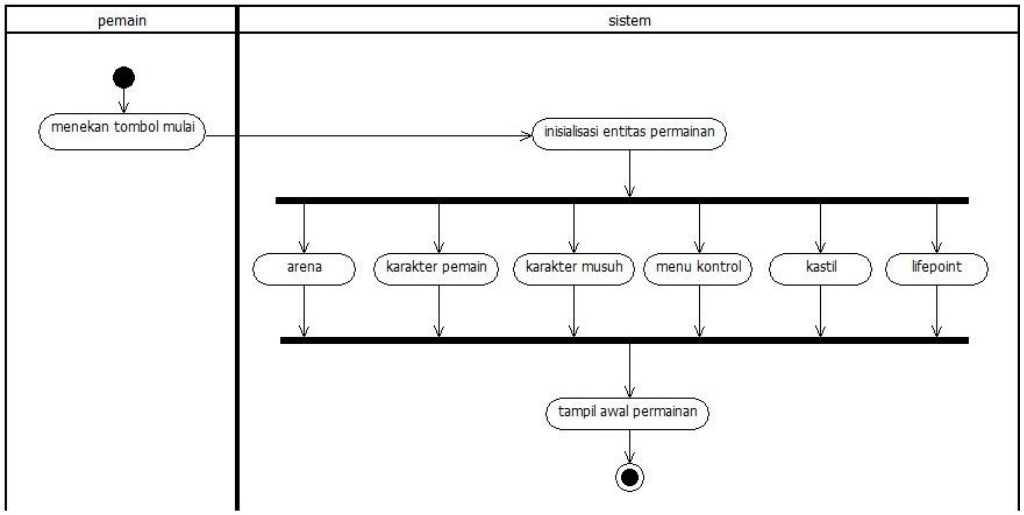
1. Alur inisialisasi *Game The Kingdom of* Pandawa

Alur inisialisasi pada *game the kingdom of* pandawa dapat dilihat pada Tabel 3.6 berikut :

Tabel 3.6 Alur inisialisasi *game the kingdom of* pandawa

No	Alur Sistem
1	Pemain akan milih menu permainan yang akan dimainkan.
2	Sistem menginisialisasi seluruh entitas yang ada di dalam arena permainan seperti arena permainan, pemain, musuh, menu kontrol, *life point*.
3	Menampilkan entitas permainan yang akan ditentukan oleh sistem.

Adapun hasil alur inisialisasi dari *game the kingdom of* pandawa disusun dari beberapa aktivitas yang akan digambarkan melalui *activity diagram* berikut ini :



Gambar 3.9 Alur Inisialisasi Permainan Pada *game the kingdom of* pandawa

Pada gambar 3.9 menjelaskan aktivitas untuk menampilkan tampilan awal permainan. Ketika pemain mulai menekan tombol mulai maka permainan akan menginisialisasi semua entitas permainan yang diantaranya arena permainan, karakter pemain, karakter musuh, menu kontrol, kastil, lifepoint. Setelah semua entitas permainan di inisialisasi maka permainan akan menampilkan tampilan awal permainan.

2. Alur kontrol melangkah *Game The Kingdom of* Pandawa

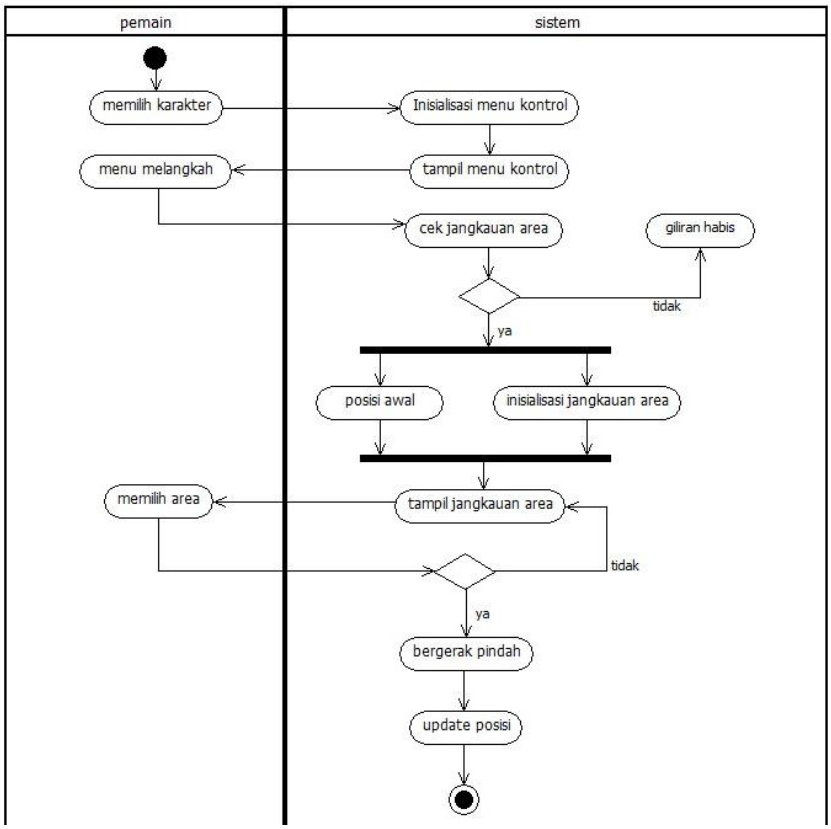
Alur kontrol melangkah pada *game the kingdom of* pandawa dapat dilihat pada Tabel 3.7 berikut :

Tabel 3.7 Alur kontrol melangkah *game the kingdom of* pandawa

No	Alur Sistem
1	Pemain memilih karakter yang akan dikontrol melalui menu kontrol yang dipilih oleh alat input *keyboard*.
2	Pemain memilih menu kontrol *move* untuk melangkah karakter mendekati musuh.
3	Sistem menge-*cek* jangkauan area melangkah karakter.
4	Sistem menampilkan hasil jangkauan area melangkah karakter.
5	Pemain memilih area yang akan dituju oleh karakter.

6	Sistem melakukan *update* posisi karakter.

Adapun hasil alur kontrol melangkah dari *game the kingdom of* pandawa disusun dari beberapa aktivitas yang akan digambarkan melalui *activity diagram* berikut ini :



Gambar 3.10 Alur Kontrol Melangkah Pada *Game the kingdom of* pandawa

Pada gambar 3.10 menjelaskan aktivitas melangkah di dalam permainan. Pemain harus memilih karakter mana yang akan dikendalikan terlebih dahulu, kemudian permainan akan menampilkan menu kontrol. Setelah itu, pemain harus memilih lagi menu melangkah, kemudian permainan akan menge-cek dan menginisialisasi jangkauan area yang dapat ditempuh. Setelah jangkauan di inisialisasi permainan pun mulai menampilkan area yang dapat di jangkau oleh karakter. Setelah tampil jangkauan area maka pemain harus memilih salah satu area yang akan ditempuh karakter, kemudian permainan akan memindahkan karakter jika pemain memilih area yang dapat ditempuh, apabila pemain memilih

area yang tidak dapat ditempuh maka permainan tidak akan ada perubahan. Setelah karakter bergerak pindah maka permainan akan meng-*update* posisi karakter.

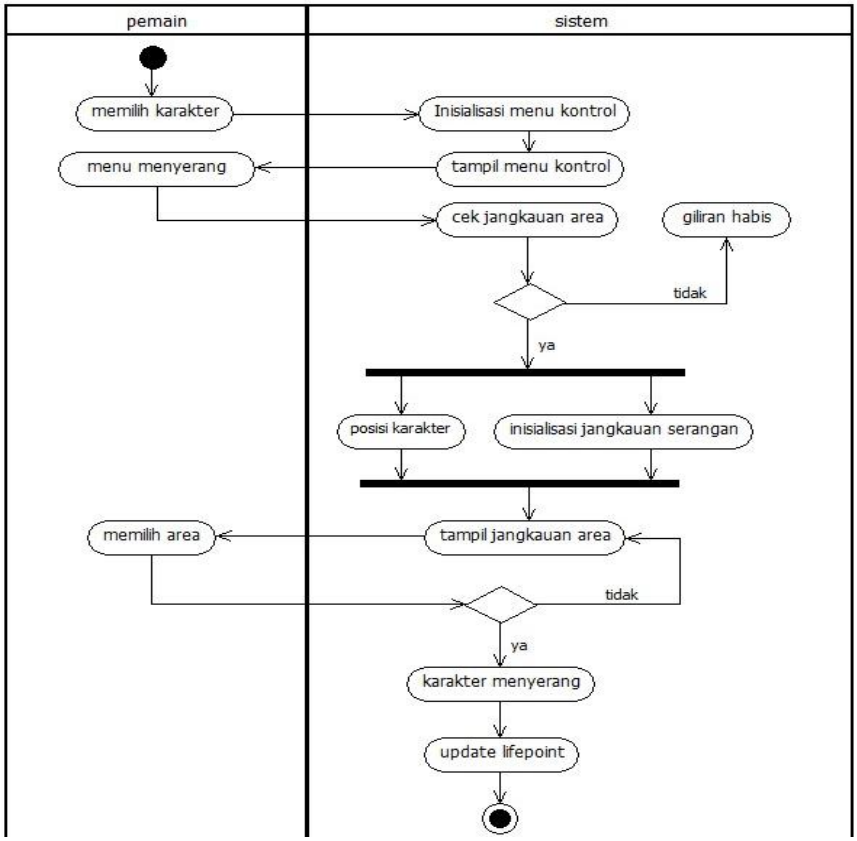
3. Alur kontrol menyerang *Game The Kingdom of* Pandawa

Alur kontrol menyerang pada *game the kingdom of* pandawa dapat dilihat pada Tabel 3.8 berikut :

Tabel 3.8 Alur kontrol menyerang *game the kingdom of* pandawa

No	Alur Sistem
1	Pemain memilih karakter yang akan dikontrol melalui menu kontrol yang dipilih oleh alat input *keyboard*.
2	Pemain memilih menu kontrol *attack* untuk menyerang musuh oleh karakter.
3	Sistem menge-*cek* jangkauan area menyerang karakter.
4	Sistem menampilkan jangkauan area menyerang karakter.
5	Pemain memilih area yang akan diserang oleh karakter.
6	Sistem melakukan *update lifepoint* karakter yang diserang oleh karakter.

Adapun hasil alur kontrol menyerang dari *game the kingdom of* pandawa disusun dari beberapa aktivitas yang akan digambarkan melalui *activity diagram* berikut ini :



Gambar 3.11 Alur kontrol menyerang pada *game the kingdom of* pandawa

Pada gambar 3.11 menjelaskan aktivitas untuk menyerang di dalam permainan. Pemain harus memilih karakter terlebih dahulu, kemudian permainan akan menampilkan menu kontrol. Setelah menu kontrol tampil pemain harus memilih salah satu menu kontrol yaitu menu menyerang untuk membuat karakter menyerang. Kemudian permainan akan menge-cek posisi dan menginisialisasi jangkauan serangan karakter, lalu permainan akan menampilkan jangkauan serangan dari posisi karakter. Setelah jangkauan serangan tampil maka pemain harus memilih salah satu area jangkauan serang karakter. Karakter akan menyerang apabila pemain memilih yang termasuk jangkauan area dan di dalam area terdapat karakter musuh, jika pemain memilih diluar jangkauan area atau tidak terdapat karakter musuh maka karakter tidak akan melakukan serangan. Setelah karakter melakukan serangan terhadap musuh permainan akan meng-*update lifepoint* yang terkena serangan.

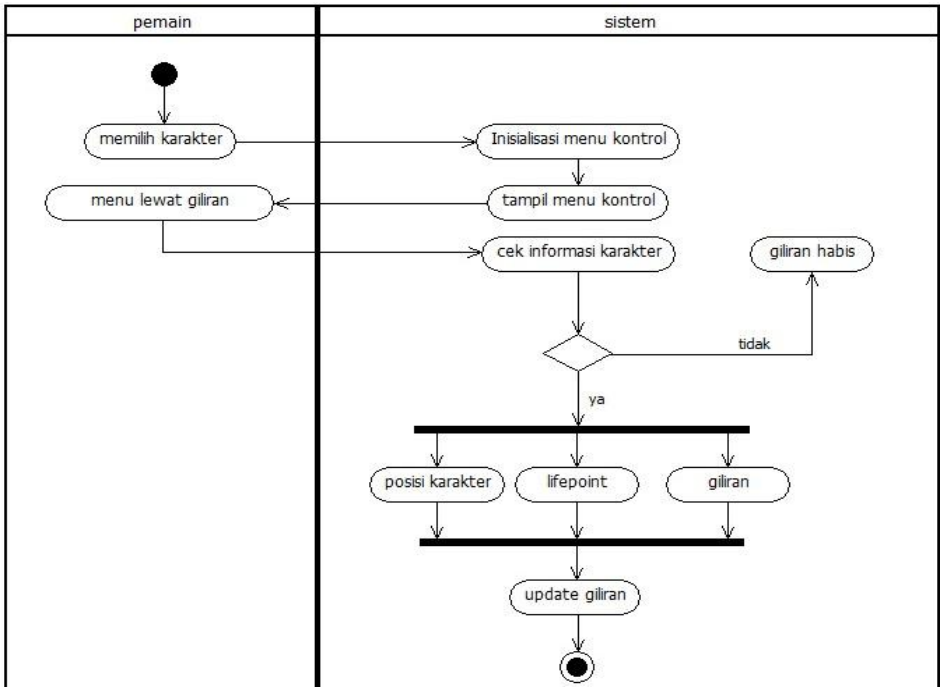
4. Alur kontrol melewati giliran *Game The Kingdom of* Pandawa

Alur kontrol melewati giliran pada *game the kingdom of* pandawa dapat dilihat pada Tabel 3.9 berikut :

Tabel 3.9 Alur kontrol melewati giliran *game the kingdom of* pandawa

No	Alur Sistem
1	Pemain memilih karakter yang akan dikontrol melalui menu kontrol yang dipilih oleh alat input *keyboard*.
2	Pemain memilih menu kontrol *wait* untuk melewati giliran karakter.
3	Sistem menge-*cek* informasi karakter.
4	Sistem melakukan *update* giliran karakter.

Adapun hasil alur kontrol melewati giliran dari *game the kingdom of* pandawa disusun dari beberapa aktivitas yang akan digambarkan melalui *activity diagram* berikut ini :



Gambar 3.12 Alur kontrol lewati giliran pada *game the kingdom of* pandawa

Pada gambar 3.12 menjelaskan aktivitas untuk melewati giliran di dalam permainan. Pemain harus memilih karakter untuk dikontrol, lalu permainan akan

merespon dengan menampilkan menu kontrol. Setelah itu pemain harus memilih menu lewat giliran, kemudian permainan akan menge-cek posisi, lifepoint, dan giliran karakter tersebut. Permainan akan meng-*update* giliran untuk mengakhiri dan menentukan giliran selanjutnya.

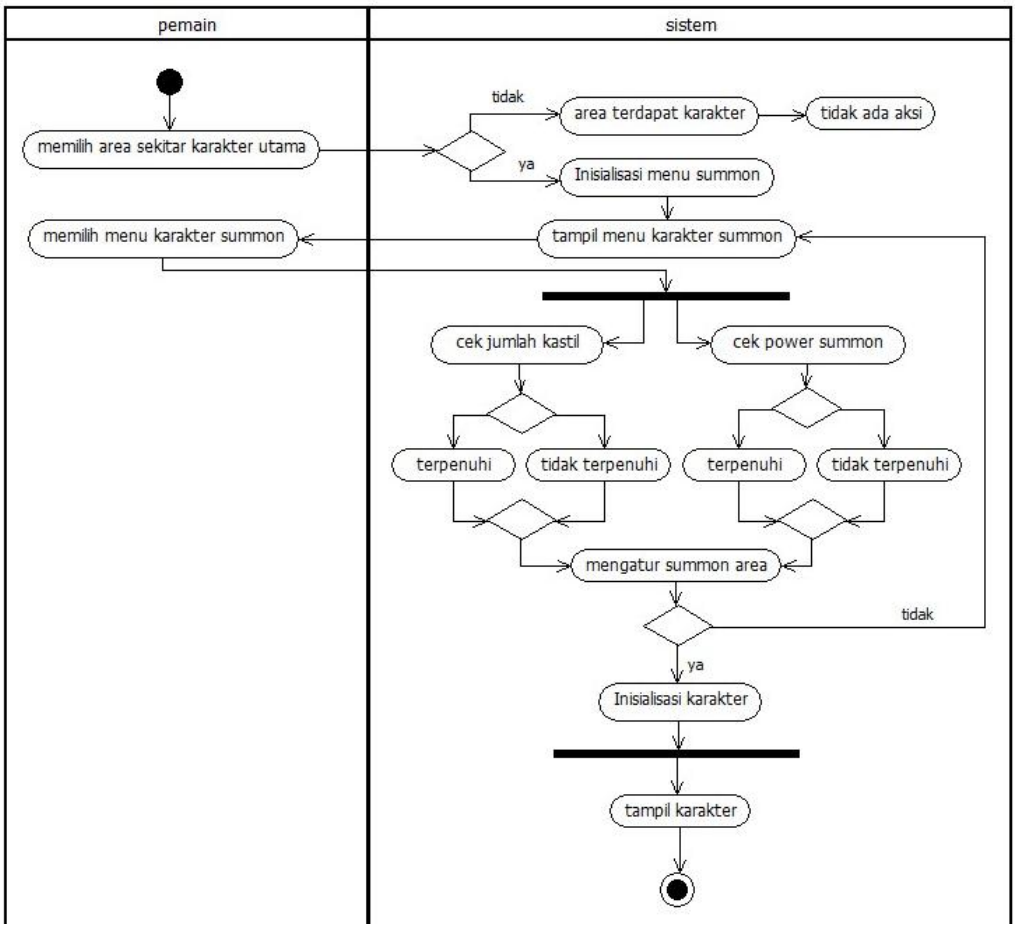
5. Alur kontrol *summon Game The Kingdom of* Pandawa

Alur kontrol *summon* pada *game the kingdom of* pandawa dapat dilihat pada Tabel 3.10 berikut :

Tabel 3.10 Alur kontrol *summon game the kingdom of* pandawa

No	Alur Sistem
1	Pemain memilih karakter yang akan dikontrol melalui menu kontrol yang dipilih oleh alat input *keyboard*.
2	Pemain dapat menempati kastil agar dapat memunculkan karakter lain.
3	Pemain dapat memunculkan karakter lain dari karakter utama.
4	Pemain memilih area di sekitar karakter utama untuk memunculkan karakter lain.
5	Sistem menampilkan *list* karakter lain yang akan dimunculkan.
6	Pemain memilih karakter lain yang akan dimunculkan.
7	Sistem menge-*cek* jumlah kastil yang telah dimiliki.
8	Sistem menge-*cek* jumlah *power summon* yang dibutuhkan.
9	Sistem melakukan inisialisasi untuk memunculkan karakter di area yang telah dipilih di sekitar karakter utama.

Adapun hasil alur *summon* dari *game the kingdom of* pandawa disusun dari beberapa aktivitas yang akan digambarkan melalui *activity diagram* berikut ini :



Gambar 3.13 Alur Kontrol *Summon* Pada *Game the kingdom of* pandawa

Pada gambar 3.13 menjelaskan aktivitas untuk memunculkan karakter di dalam permainan. Untuk memunculkan karakter lain hanya dapat dilakukan oleh karakter utama. Pemain harus menekan atau memilih salah satu area sekitar karakter utama, kemudian permainan akan menampilkan menu summon yang berisi daftar karakter yang dapat dimunculkan. Setelah itu pemain harus memilih salah satu karakter untuk dimunculkan, lalu sistem akan memnge-cek syarat memunculkan karakter yaitu harus menempati kasti dan mempunyai *power summon* yang cukup untuk karakter yang akan dimunculkan. Apabila syarat untuk memunculkan terpenuhi maka permainan akan memunculkan karakter berdasarakan area yang telah dipilih sebelumnya, jika syarat tidak terpenuhi maka permainan tidak akan merespon atau karakter tidak akan muncul. Permainan akan meng-*update* posisi karakter yang baru muncul.

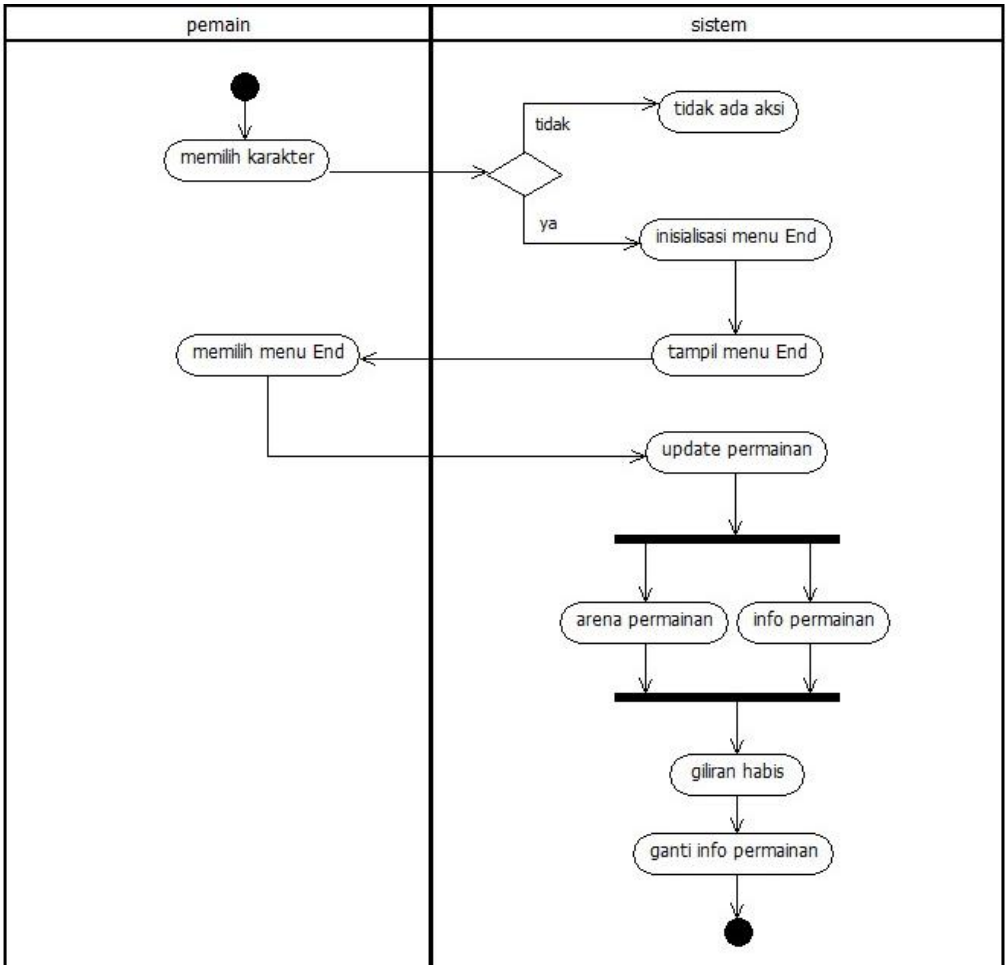
6. Alur kontrol mengakhiri giliran *Game The Kingdom of* Pandawa

Alur kontrol mengakhiri giliran pada *game the kingdom of* pandawa dapat dilihat pada Tabel 3.11 berikut :

Tabel 3.11 Alur kontrol mengakhiri giliran *game the kingdom of* pandawa

No	Alur Sistem
1	Pemain memilih karakter yang akan dikontrol melalui menu kontrol yang dipilih oleh alat input *keyboard*.
2	Pemain memilih menu mengakhiri giliran.
3	Sistem menyimpan informasi permainan.
4	Sistem mengakhiri giliran
5	Sistem mengganti info permainan.

Adapun hasil alur kontrol mengakhiri giliran dari *game the kingdom of* pandawa disusun dari beberapa aktivitas yang akan digambarkan melalui *activity diagram* berikut ini :



Gambar 3.14 Alur Mengakhiri Giliran Pada *Game the kingdom of* pandawa

Pada gambar 3.14 menjelaskan aktivitas untuk mengakhiri giliran di dalam permainan. Pemain harus memilih karakter untuk dikontrol, lalu permainan akan merespon dengan menampilkan menu kontrol mengakhiri giliran. Setelah itu pemain harus memilih menu mengakhiri giliran, kemudian permainan akan menge-cek posisi, lifepoint, dan giliran karakter tersebut. Permainan akan menyimpan info permainan untuk mengakhiri dan mengganti info permainan dengan informasi permainan selanjutnya.

B. Misi Permainan

Misi permainan pada *game the kingdom of* pandawa dapat dilihat dari Tabel 3.7 berikut:

Table 3.12 Misi permainan pada *game the kingdom of* pandawa

Misi Permainan
Pemain harus mengambil base dari kurawa, dengan mengalahkan prajurit Kurawa. Ketika prajurit Kurawa dikalahkan semua, maka pemain akan melanjutkan ke *level* berikutnya. Dalam *level* terakhir pemain harus mengalahkan *boss* dalam *base* tersebut yaitu Kurawa.

3.1.2.6 Analisis Tingkat Kesulitan

Pada setiap level, musuh akan diterapkan algoritma A* (A-Star) untuk tingkat kesulitan pemain. Pemain dapat bergerak dan menyerang sesuai gilirannya, kemudian pemain harus menempati kastil untuk memunculkan teman di arena permainan dan secara bersamaan pun musuh akan menggunakan gilirannya untuk menyerang. Dalam permainan ini dibagi menjadi 3 tingkat kesulitan yaitu, jumlah musuh yang berperang dengan pemain dalam setiap level berbeda-beda jumlahnya, musuh yang mempunyai status berperang seperti *Lifepoint*, *Attack Power*, *Move Range* dapat berpengaruh ketika menyerang pemain, jumlah kastil yang dapat pemain tempati tiap level akan berkurang.

3.1.3 Analisis Metode pada *Game The Kingdom of* Pandawa

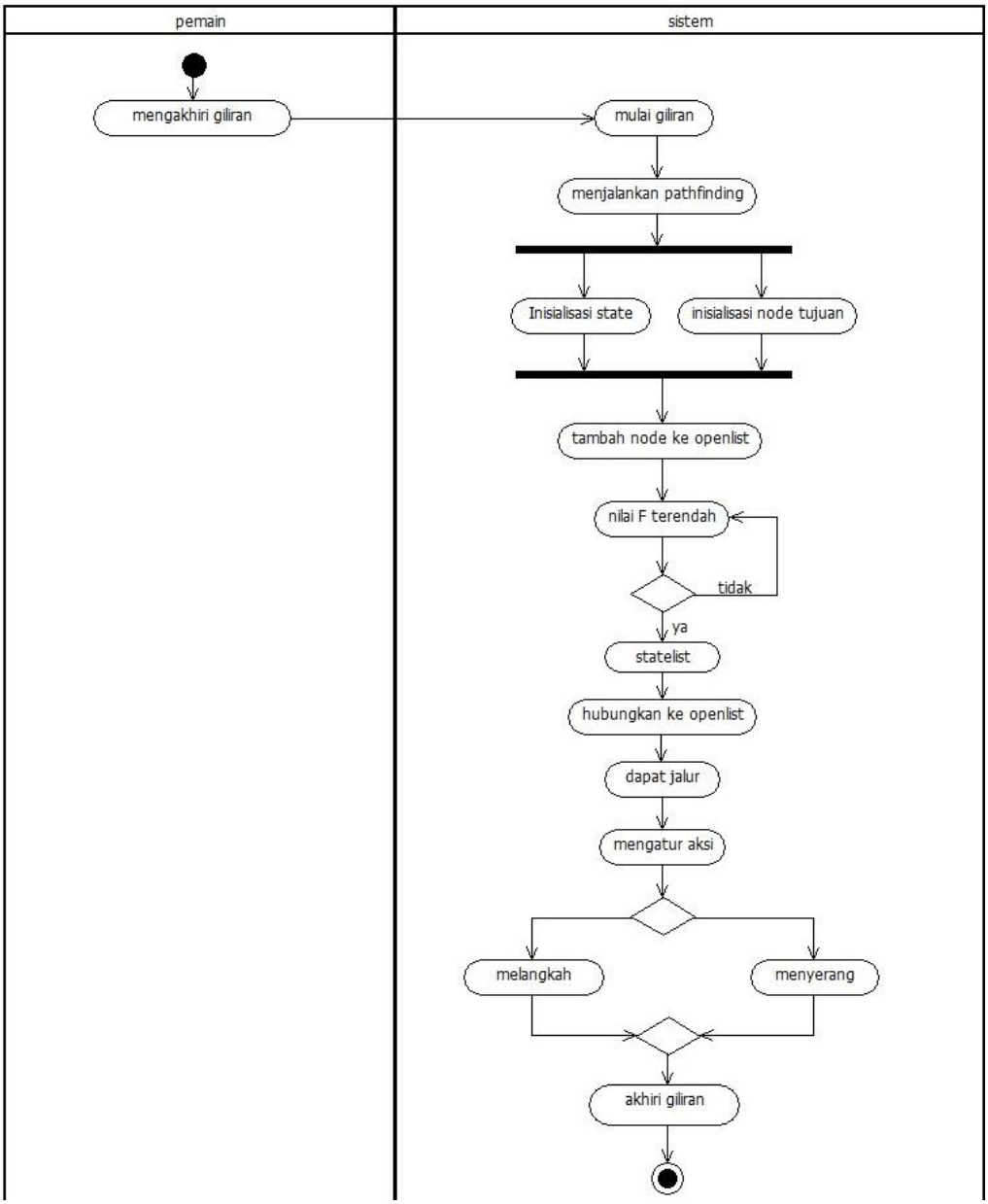
Untuk membuat *game* menjadi lebih menarik dapat dilakukan dengan meningkatkan kualitas *gameplay*. *Artificial Intellegence* (AI) disebut dapat meningkatkan *gameplay* pada semua jenis *game*, termasuk pada *game the kingdom of* pandawa yang akan dibangun. Beberapa hal yang dapat menentukan keberhasilan suatu *game* yang menerapkan AI yaitu kesuksesan dalan melakukan pencarian dan pelacakan. Dengan adanya AI dapat membuat musuh atau *non-playable characters* (NPC) mengatasi masalah yang ada. Salah satu masalah

dalam *game* secara umum dikenal dengan nama *pathfinding*. Yang merupakan suatu masalah yang digunakan untuk menentukan pergerakan suatu objek berdasarkan keadaan lokasi atau objek yang lainnya. Masalah yang timbul membutuhkan teknik pencarian sehingga dapat melacak dan menentukannya. A* merupakan teknik pencarian yang sudah dikenal sebagai salah satu teknik pencarian yang dapat mencari solusi optimum secara cepat, dikarenakan A* merupakan pencarian terbimbing. Pada dasarnya ada dua teknik pencarian dan pelacakan yang digunakan, yaitu pencarian buta (*blind search*) dan pencarian terbimbing (*heuristic search*).

Algoritma A* diterapkan pada NPC dalam *game the kingdom of* pandawa untuk melakukan *pathfinding*, dalam proses pencariannya akan dimodelkan dengan funsi *heuristic*. Algoritma A* akan bekerja dari titik awal ke titik tujuan dengan mencari jalan dari akumulasi bobot terendah (dilambangkan dengan notasi g) dan dijumlahkan dengan perkiraan jalan terendah dengan bobotnya untuk mencapai tujuan (dilambangkan dengan notasi h). Rute yang memiliki node dengan nilai f(n) = g(n) + h(n), node yang terkecil akan ditelusuri terlebih dahulu. Dari fungsi *heuristic* dapat digambarkan kemungkinan titik-titik yang dapat dikunjungi oleh musuh untuk mencapai titik tujuan dalam melakukan pencarian lokasi pemain. Fungsi ini pun akan meminimumkan total biaya lintasan yang akan ditempuh musuh.

Dengan adanya fungsi *heuristic* dari algoritma A*, maka musuh dapat mencari lokasi pemain di dalam arena permainan walaupun lokasinya berada jauh dari posisi musuh.

Adapun aktivitas dari NPC pada *game the kingdom of* pandawa adalah sebagai berikut :

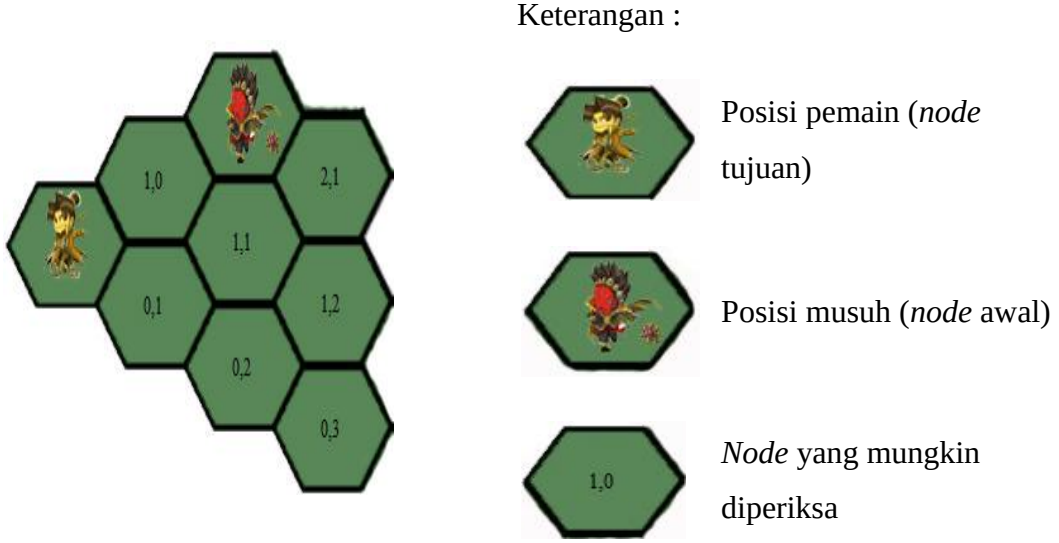


Gambar 3.15 *Activity* NPC pada *game the kingdom of* pandawa

Pada gambar 3.15 menjelaskan bahwa *game the kingdom of* pandawa akan dibentuk *pathfinding* yang akan dijalankan ketika permainan dimulai. AI *game* akan menginisialisasi state NPC untuk menemukan jalur dari posisi awal dan menginisialisasi node tujuan untuk menemukan jalur sampai tujuan. Kemudian node awal akan ditambahkan ke openlist, setelah menambahkan *node* maka *node* akan dicari nilai F terendahnya untuk ditambahkan ke calon jalur yang akan ditempuh yaitu statelist. Kemudian node openlist yang sebelumnya akan di

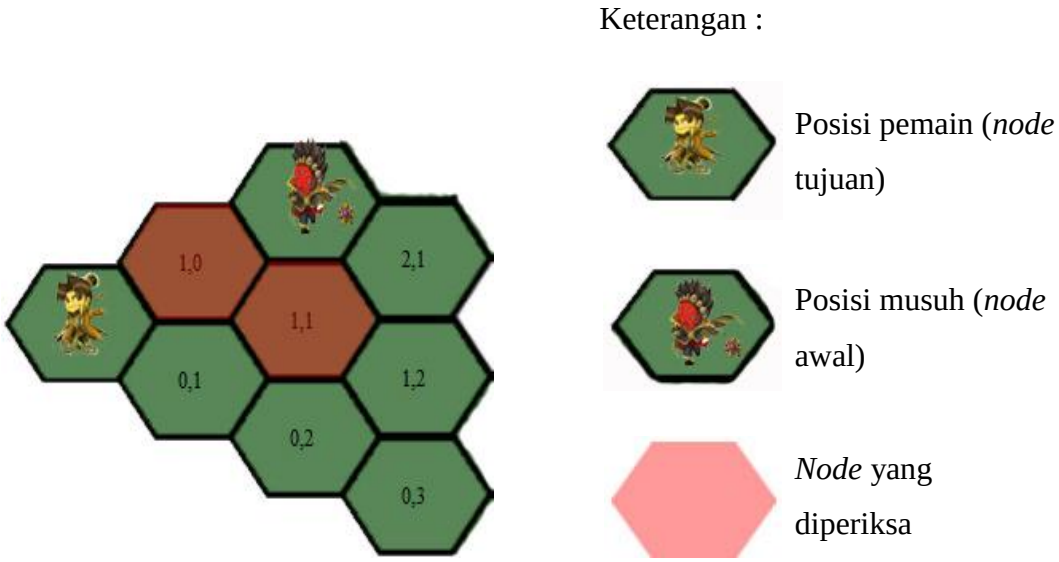
hubungkan dengan *statelist* yang telah ditambahkan sebelumnya, setelah itu jalur bisa didapat. Kemudian NPC akan mengatur aksi apa yang akan dilakukan berdasarkan *decision making* yaitu bila jarak tempuh masih kurang dengan jarak jarak tempuh yang dimiliki karakter maka NPC akan mengatur aksi melangkah dan apabila jarak tempuh yang ditempuh dapat menemukan *node* berdasarkan jarak tempuh yang dimiliki karakter maka NPC akan mengatur aksi menyerang.

Pada metode *pathfinding* yang terdapat dalam *game the kingdom of* pandawa dapat diilustrasikan pada gambar 3.16 sebagai berikut :



Gambar 3.16 ilustrasi awal *pathfinding* pada *game the kingdom of* pandawa

Pada ilustrasi ini terdapat *node* awal atau posisi musuh yang berada pada koordinat (2,0), sedangkan *node* tujuan atau posisi pemain berada pada koordinat (0,0). Pada langkah selanjutnya akan dilakukan pemeriksaan terhadap *node-node* yang berdekatan dengan *node* musuh. Node yang akan dipilih oleh musuh adalah *node* yang memiliki *cost* atau biaya terkecil sebagai *Best Node*, dengan melakukan perhitungan sebagai berikut :



Gambar 3.17 pemeriksaan jalur pada *node* (1,0) dan *node* (1,1)

a. Node (1,0)

Nilai n.x = 1
Nilai n.y = 0

Node tujuan (0,0) sehingga goal.x = 0, goal.y = 0

g(1,0) = 1

h_orthogonal(n) = (abs(n.x – goal.x) + abs(n.y – goal.y))
h_orthogonal(1,0) = (abs(1 – 0) + abs(0 – 0))
= (abs(1) + abs(0))
= 1

h_diagonal(n) = min(abs(n.x – goal.x) + abs(n.y – goal.y))
h_diagonal(1,0) = min(abs(1 – 0) + abs(0 – 0))
= min(abs(1) + abs(0))
= min 1

h(n) = h_diagonal + (h_orthogonal(n) – (2*h_diagonal(n)))
h(1,0) = (-1) + (1 – (2*(-1)))
= -1 + 3

= 2

f(1,0) = g(1,0) + h(1,0)
= 1 + 2
= 3

b. Node (1,1)

Nilai n.x = 1
Nilai n.y = 1

Node tujuan (0,0) sehingga goal.x = 0, goal.y = 0

g(1,1) = 1

h_orthogonal(n) = (abs(n.x – goal.x) + abs(n.y – goal.y))
h_orthogonal(1,1) = (abs(1 – 0) + abs(1 – 0))
= (abs(1) + abs(1))
= 2

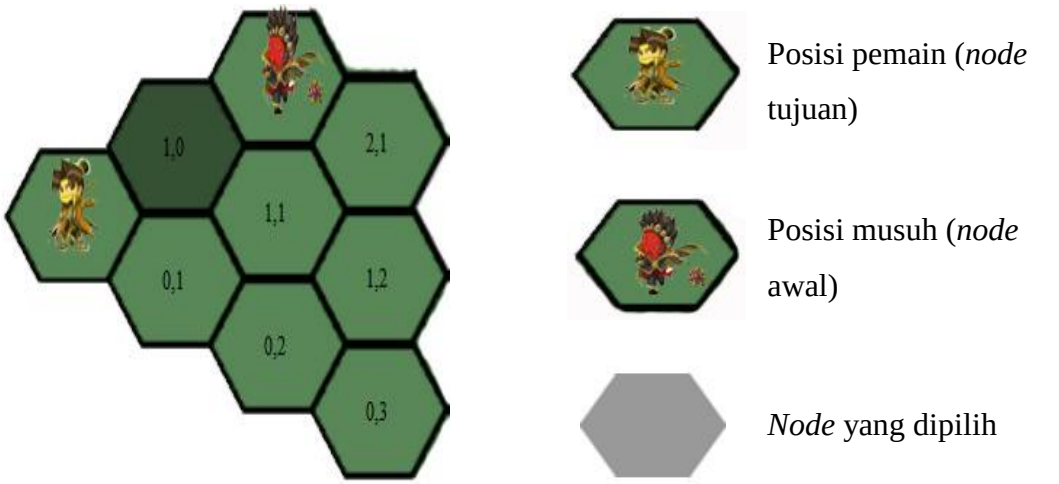
h_diagonal(n) = min(abs(n.x – goal.x) + abs(n.y – goal.y))
h_diagonal(1,1) = min(abs(1 – 0) + abs(1 – 0))
= min(abs(1) + abs(1))
= min 2

h(n) = h_diagonal + (h_orthogonal(n) – (2*h_diagonal(n)))
h(1,1) = (-2) + (2 – (2*(-2)))
= -2 + 6
= 4

f(1,0) = g(1,0) + h(1,0)
= 1 + 4
= 5

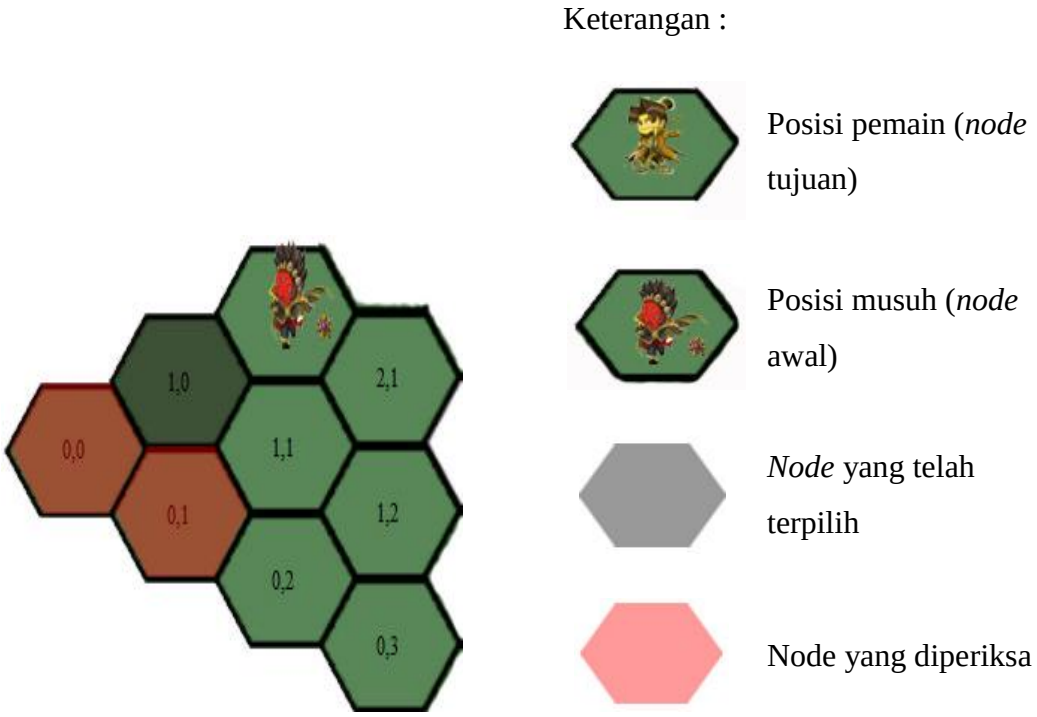
Dari hasil perhitungan diatas terlihat terdapat 2 *node* yang menjadi calon *BestNode* yaitu (1,0) dengan f(n) = 3, dan (1,1) dengan f(n) = 5. Dari 2 *node*

tersebut maka dipilih *BestNode* yang memiliki *cost* biaya terkecil yaitu *node* (1,0) dengan f(n) = 3.



Gambar 3.18 *Node* yang terpilih sebagai *Best Node*

Setelah mendapatkan *BestNode* dilangkah pertama, dilakukan pencarian jalur lanjutan untuk mendapatkan *BestNode* selanjutnya seperti langkah sebelumnya. Pencarian jalur terdapat pada koordinat (0,0) dan koordinat (0,1) sebagai berikut :



Gambar 3.19 pemeriksaan jalur pada *node* (0,0) dan *node* (0,1)

a. Node (0,0)

Nilai n.x = 0
Nilai n.y = 0

Node tujuan (0,0) sehingga goal.x = 0, goal.y = 0

g(n) = g(0,0) = 0

h_orthogonal(n) = (abs(n.x – goal.x) + abs(n.y – goal.y))
h_orthogonal(0,0) = (abs(0 – 0) + abs(0 – 0))
= (abs(0) + abs(0))
= 0

h_diagonal(n) = min(abs(n.x – goal.x) + abs(n.y – goal.y))
h_diagonal(0,0) = min(abs(0 – 0) + abs(0 – 0))
= min(abs(0) + abs(0))
= min 0

h(n) = h_diagonal(n) + (h_orthogonal(n) – (2* h_diagonal(n)))
h(0,0) = (-0) + (0 – (2*(-0)))
= 0 + 0
= 0

f(0,0) = g(0,0) + h(0,0)
= 0 + 0
= 0

b. Node (0,1)

Nilai n.x = 0
Nilai n.y = 1

Node tujuan (0,0) sehingga goal.x = 0, goal.y = 0

g(n) = g(0,1) = 0

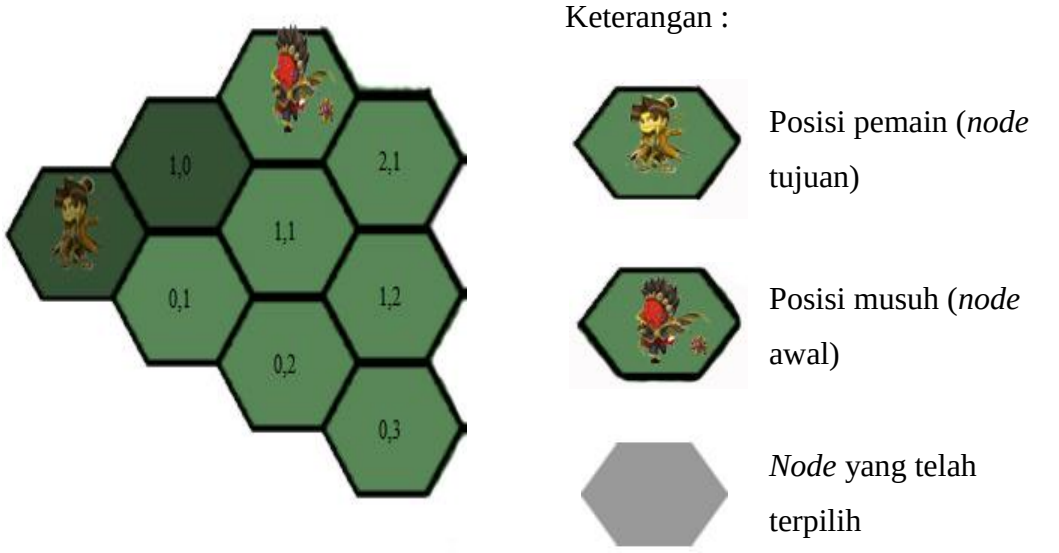
h_orthogonal(n) = (abs(n.x – goal.x) + abs(n.y – goal.y))
h_orthogonal(0,1) = (abs(0 – 0) + abs(1 – 0))
= (abs(0) + abs(1))
= 1

h_diagonal(n) = min(abs(n.x – goal.x) + abs(n.y – goal.y))
h_diagonal(0,1) = min(abs(0 – 0) + abs(1 – 0))
= min(abs(0) + abs(1))
= min 1

h(n) = h_diagonal(n) + (h_orthogonal(n) – (2* h_diagonal(n)))
h(0,0) = (-1) + (1 – (2*(-1)))
= -1 + 3
= 2

f(0,0) = g(0,0) + h(0,0)
= 0 + 2
= 2

Dari hasil perhitungan diatas terlihat terdapat 2 *node* yang menjadi calon *BestNode* yaitu (0,0) dengan f(n) = 0, dan (0,1) dengan f(n) = 2. Dari 2 *node* tersebut maka dipilih *BestNode* yang memiliki *cost* biaya terkecil yaitu *node* (0,0) dengan f(n) = 0.



Gambar 3.20 Hasil *pathfinding*

Dari seluruh perhitungan yang telah dilakukan di pilihlah biaya *cost* terkecil pada setiap langkahnya sehingga jalur terpendek yang akan dilalui musuh untuk melakukan pencarian posisi pemain.

3.1.4 Analisis Kebutuhan Non Fungsional

Analisis kebutuhan non-fungsional merupakan analisis yang dibutuhkan untuk menentukan spesifikasi kebutuhan sistem[8]. Spesifikasi ini juga meliputi elemen atau komponen yang dibutuhkan untuk sistem yang akan dibangun sampai dengan sistem tersebut dapat diimplementasikan. Analisis kebutuhan ini juga menentukan spesifikasi masukan yang diperlukan sistem, keluaran yang akan dihasilkan, dan proses yang dibutuhkan untuk mengolah masukan sehingga menghasilkan suatu keluaran yang diinginkan.

3.1.4.1 Analisis Kebutuhan Perangkat Lunak (Penulis)

Perangkat lunak digunakan dalam sebuah sistem merupakan perintah yang diberikan kepada perangkat keras agar dapat saling berinteraksi diantara keduanya. Perangkat lunak yang dibutuhkan untuk membangun *game the kingdom of* pandawa ini adalah sebagai berikut :

Tabel 3.13 Spesifikasi Perangkat Lunak (Penulis)

Sistem operasi	Windows 7 *Ultimate* 32-bit
Perangkat lunak (*compiler*)	Eclipse Hellios
Perangkat lunak (*development kit*)	JDK 6 (*Java Development Kit*)
Perangkat lunak (desain tools)	Adobe Photoshop CS 2
Perangkat lunak (desain analisis)	StarUML

3.1.4.2 Analisis Kebutuhan Perangkat Keras (Penulis)

Komputer terdiri dari perangkat keras dan perangkat lunak yang saling berinteraksi. Perangkat lunak memberikan instruksi kepada perangkat keras untuk melakukan tugas tertentu. Adapun perangkat keras yang digunakan dalam membangun *game the kingdom of* pandawa adalah sebagai berikut :

Tabel 3.14 Spesifikasi Perangkat Keras (Penulis)

Nama Perangkat	Spesifikasi
Prosesor	Dual Core
RAM	1024MB
Harddisk	40G
Display	Resolusi 800 x 600
Keyboard	Standar
Mouse	Standar

3.1.4.3 Analisis Pengguna

Spesifikasi minimal bagi pengguna yang dapat menggunakan *game* ini adalah sebagai berikut :

Tabel 3.15 Spesifikasi Pengguna

Peran	Pemain
Usia	7 Tahun ke atas
Tingkat Keterampilan	Dapat menggunakan komputer
Pengalaman	Pernah memainkan aplikasi permainan lainnya seperti *garden defense, warcraft*, dan lain-lain.

3.1.4.4 Analisis Perangkat Lunak (Pengguna)

Pada sisi pengguna, perangkat keras yang dibutuhkan untuk menjalankan *game* ini adalah sebagai berikut :

Tabel 3.16 Spesifikasi Perangkat Lunak (Pengguna)

Sistem operasi	*Windows* 7
Perangkat lunak	JDK 6 (*Java Development Kit*)

3.1.4.5 Analisis Perangkat Keras (Pengguna)

Pada sisi pengguna, perangkat lunak yang dibutuhkan untuk menjalankan *game* ini adalah sebagai berikut :

Tabel 3.17 Spesifikasi Perangkat Keras (Pengguna)

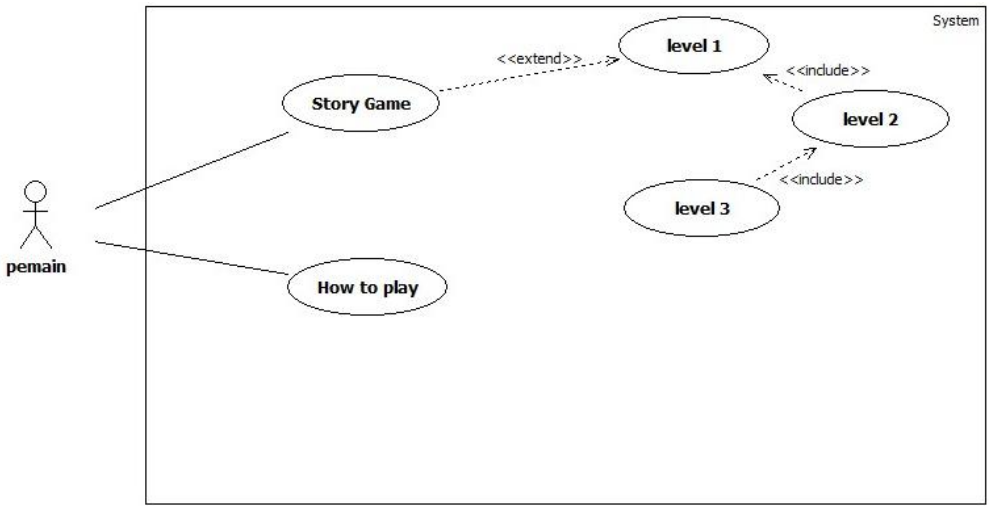
Nama Perangkat	Spesifikasi
Prosesor	Dual Core
RAM	Minimum 256 MB
Harddisk	Minimum 500 MB
Display	Resolusi 800 x 600
Keyboard	Standar
Mouse	Standar

3.1.5 Analisis Kebutuhan Fungsional

3.1.5.1 *Use case* Diagram

Use case atau diagram *use case* merupakan gambaran fungsionalitas dari suatu sistem sehingga pengguna sistem mengerti mengenai kegunaan sistem yang akan dibangun.

Berikut ini adalah interaksi antara pengguna dengan *game the kingdom of* pandawa yang digambarkan pada diagram *use case* berikut ini :



Gambar 3.21 Diagram *Use Case Game the kingdom of* pandawa.

3.1.5.2 Definisi *Use case*

Definisi *use case* berfungsi untuk menjelaskan fungsi dari *use case* yang terdapat pada diagram *use case*. Definisi *use case* dijelaskan pada tabel berikut:

Tabel 3.18 Definisi *Use case*

No	*Use case*	Keterangan
1	*Story Game*	Proses untuk memulai permainan sesuai cerita.
2	*How to play*	Proses untuk menampilkan informasi petunjuk permainan
3	*Level* 1	Proses permainan pada *level* 1
4	*Level* 2	Proses permainan pada *level* 2
5	*Level* 3	Proses permainan pada level 3

3.1.5.3 Skenario *Use case*

Skenario pada diagram *use case* merupakan penjelasan mengenai alur kejadian yang di dalamnya menjelaskan urutan interaksi antara *actor* dengan *use*

case tersebut. Adapun skenario *use case* yang terdapat pada *game the kingdom of* pandawa adalah sebagai berikut :

1. **Skenario *Use case Story Game***

Skenario *use case* ini menjelaskan interaksi antara aktor, yaitu pemain dengan *use case story game* yang dijelaskan pada tabel berikut ini :

Tabel 3.19 Skenario *Use case Story Game*

Identifikasi	
Nama	*Story Game*
Tujuan	Masuk ke permainan kemudian mulai menyelesaikan permainan.
Aktor	Pemain
Kondisi Awal	Menu utama ditampilkan
Skenario Normal	
Aksi Aktor	**Reaksi Sistem**
1. Tekan menu *Story Game*	
	2. Masuk ke permainan level 1
3. Menyelesaikan permainan level 1	
	4. Masuk ke permainan level 2
5. Menyelesaikan permainan level 2	
	6. Masuk ke permainan level 3
Skenario Alternatif	
Aksi Aktor	**Reaksi Sistem**
1. Tekan menu *Story game*	
	2. Inisialisasi entitas permainan level 1
	3. Tampil permainan level 1
4. Mulai permainan level 1	
5. Menyelesaikan misi	
	6. Permainan selesai
	7. Inisialisasi entitas permainan level 2

	8. Tampil permainan level 2
9. Mulai permainan level 2	
10. Menyelesaikan misi	
	11. Permainan level 2 selesai
	12. Inisialisasi entitas permainan
	13. Tampil permainan level 3
14. Mulai permainan level 3	
15. Menyelesaikan misi	
	16. Permainan level 3 selesai
Kondisi Akhir	Pemain masuk ke dalam permainan.

2. Skenario *Use case How to play*

Skenario *use case* ini menjelaskan interaksi antar aktor, yaitu pemain dengan *use case* petunjuk yang dijelaskan pada tabel berikut ini :

Tabel 3.20 Skenario *Use case How to play*

Identifikasi	
Nama	*How to play*
Tujuan	Untuk dapat mengetahui bagaimana cara permainan ini dimainkan
Aktor	Pemain
Skenario Normal	
Kondisi Awal	
Aksi Aktor	**Reaksi Sistem**
1. Pilih menu *How to play*	
	2. Menampilkan informasi dari cara bermain
Skenario Alternatif	
1. Pilih menu *How to play*	
	2. Inisialisasi petunjuk permainan
	3. Tampil informasi petunjuk

	permainan
4. Tekan tombol lanjut	
	5. Tampil informasi petunjuk permainan yang lainnya
Kondisi Akhir	Menampilkan informasi mengenai petunjuk bermain

3. Skenario *Use case Level* 1

Skenario *use case* ini menjelaskan interaksi antar aktor, yaitu pemain dengan *use case Level* 1 yang dijelaskan pada tabel berikut ini :

Tabel 3.21 Skenario *Use case Level* 1

Identifikasi	
Nama	*Level* 1
Tujuan	Untuk dapat memulai permainan dari *level* 1
Aktor	Pemain
Skenario Utama	
Kondisi Awal	Karakter utama pandawa berada di tepi kiri atas layar dengan musuh yang berada di kanan layar.
Aksi Aktor	**Reaksi Sistem**
1. Memilih *story game*	
	2. Masuk ke level 1
	3. Inisialisasi seluruh konten permainan yang dibutuhkan untuk *level* 1
4. Mulai permainan *level* 1	
5. Kontrol permainan	
	6. Inisialisasi kontrol permainan
	7. Tampil kursor dalam permainan
8. Pilih karakter	
	9. Inisialisasi menu karakter
	10. Tampil menu karakter

11. Pilih menu karakter	
	12. Karakter bereaksi sesuai menu karakter
	13. Update kontrol karakter
14. Tekan area dekat karakter utama	
	15. Inisialisasi menu summon
	16. tampil menu summon
17. pilih menu summon	
	18. inisialisasi karakter summon
	19. tampil karakter summon di dalam area
20. Menyelesaikan misi *level* 1	
	21. Permainan *level* 1 selesai
Skenario Alternatif	
1. Tekan multiplayer	
	2. Inisialisasi entitas permainan
	3. Tampil permainan
4. Mulai permainan	
5. Kontrol permainan	
	6. Inisialisasi kontrol permainan
	7. Tampil kursor di permainan
8. Tekan karakter	
	9. Inisialisasi menu karakter
	10. Tampil menu karakter
11. Pilih menu karakter	
	12. Jika sudah pernah memilih menu karakter dalam satu giliran, maka sistem tidak akan merespon kembali
	13. Karakter bereaksi sesuai menu

	karakter
	14. Update kontrol permainan
15. Tekan area dekat karakter utama	
	16. Jika area terdapat karakter lain, maka sistem tidak dapat menginisialisasi menu summon
	17. Inisialisasi menu summon
	18. Tampil menu summon
19. Pilih menu summon	
	20. Jika jumlah power summon tidak terpenuhi, maka sistem tidak akan merespon
	21. Jika jumlah kastil tidak terpenuhi, maka sistem tidak akan merespon
	22. Jika sudah pernah memilih menu summon dalam satu giliran, maka sistem tidak akan merespon kembali
	23. Inisialisasi karakter summon
	24. Tampil karakter summon di dalam area permainan
25. Pilih menu End	
	26. Sistem akan menyimpan informasi permainan yaitu area permainan dan posisi karakter
	27. Sistem akan menyudahi giliran dengan mengganti info permainan
	28. Sistem menjalankan AI pada

	karakter musuh
	29. Sistem akan menggerakan dan membuat karakter musuh menyerang
	30. Sistem menyimpan informasi permainan yaitu arena permainan dan posisi karakter
	31. Sistem mengganti menyudahi giliran dan mengganti info permainan
32. Menyelesaikan misi permainan	
	33. Jika permainan *level* tidak selesai, maka sistem tidak akan bias melanjutkan ke permainan *level* 2
	34. Permainan *level* 1 selesai
Kondisi Akhir	Mengalahkan seluruh karakter musuh dalam submenu *level* 1

4. Skenario *Use case Level* 2

Skenario *use case* ini menjelaskan interaksi antar aktor, yaitu pemain dengan *use case Level* 2 yang dijelaskan pada tabel berikut ini :

Tabel 3.22 Skenario *Use case Level* 2

Identifikasi	
Nama	*Level* 2
Tujuan	Untuk dapat memulai permainan dari *level* 2
Aktor	Pemain
Skenario Utama	
Kondisi Awal	Karakter utama pandawa berada di tepi kiri atas layar dengan musuh yang berada di kanan layar.

Aksi Aktor	**Reaksi Sistem**
1. Memilih *story game*	
	2. Masuk ke level 2
	3. Inisialisasi seluruh konten permainan yang dibutuhkan untuk *level* 2
4. Mulai permainan *level* 2	
5. Kontrol permainan	
	6. Inisialisasi kontrol permainan
	7. Tampil kursor dalam permainan
8. Pilih karakter	
	9. Inisialisasi menu karakter
	10. Tampil menu karakter
11. Pilih menu karakter	
	12. Karakter bereaksi sesuai menu karakter
	13. Update kontrol karakter
14. Tekan area dekat karakter utama	
	15. Inisialisasi menu summon
	16. tampil menu summon
17. pilih menu summon	
	18. inisialisasi karakter summon
	19. tampil karakter summon di dalam area
20. Menyelesaikan misi *level* 2	
	21. Permainan *level* 2 selesai
Skenario Alternatif	
1. Tekan multiplayer	
	2. Inisialisasi entitas permainan

	3. Tampil permainan
4. Mulai permainan	
5. Kontrol permainan	
	6. Inisialisasi kontrol permainan
	7. Tampil kursor di permainan
8. Tekan karakter	
	9. Inisialisasi menu karakter
	10. Tampil menu karakter
11. Pilih menu karakter	
	12. Jika sudah pernah memilih menu karakter dalam satu giliran, maka sistem tidak akan merespon kembali
	13. Karakter bereaksi sesuai menu karakter
	14. Update kontrol permainan
15. Tekan area dekat karakter utama	
	16. Jika area terdapat karakter lain, maka sistem tidak dapat menginisialisasi menu summon
	17. Inisialisasi menu summon
	18. Tampil menu summon
19. Pilih menu summon	
	20. Jika jumlah power summon tidak terpenuhi, maka sistem tidak akan merespon
	21. Jika jumlah kastil tidak terpenuhi, maka sistem tidak akan merespon
	22. Jika sudah pernah memilih

	menu summon dalam satu giliran, maka sistem tidak akan merespon kembali
	23. Inisialisasi karakter summon
	24. Tampil karakter summon di dalam area permainan
25. Pilih menu End	
	26. Sistem akan menyimpan informasi permainan yaitu area permainan dan posisi karakter
	27. Sistem akan menyudahi giliran dengan mengganti info permainan
	28. Sistem menjalankan AI pada karakter musuh
	29. Sistem akan menggerakan dan membuat karakter musuh menyerang
	30. Sistem menyimpan informasi permainan yaitu arena permainan dan posisi karakter
	31. Sistem mengganti menyudahi giliran dan mengganti info permainan
32. Menyelesaikan misi permainan	
	33. Jika permainan *level* 2 tidak selesai, maka sistem tidak akan bias melanjutkan ke permainan *level* 3
	34. Permainan *level* 2 selesai

Kondisi Akhir	Mengalahkan seluruh karakter musuh dalam submenu *level* 2

6. Skenario *Use case Level* 3

Skenario *use case* ini menjelaskan interaksi antar aktor, yaitu pemain dengan *use case Level* 3 yang dijelaskan pada tabel berikut ini :

Tabel 3.23 Skenario *Use case Level* 3

Identifikasi		
Nama	*Level* 3	
Tujuan	Untuk dapat memulai permainan dari *level* 3	
Aktor	Pemain	
Skenario Utama		
Kondisi Awal	Karakter utama pandawa berada di tepi kiri atas layar dengan musuh yang berada di kanan layar.	
Aksi Aktor		**Reaksi Sistem**
1. Memilih *story game*		
		2. Masuk ke level 3
		3. Inisialisasi seluruh konten permainan yang dibutuhkan untuk *level* 3
4. Mulai permainan *level* 3		
5. Kontrol permainan		
		6. Inisialisasi kontrol permainan
		7. Tampil kursor dalam permainan
8. Pilih karakter		
		9. Inisialisasi menu karakter
		10. Tampil menu karakter
11. Pilih menu karakter		
		12. Karakter bereaksi sesuai menu

	karakter
	13. Update kontrol karakter
14. Tekan area dekat karakter utama	
	15. Inisialisasi menu summon
	16. tampil menu summon
17. pilih menu summon	
	18. inisialisasi karakter summon
	19. tampil karakter summon di dalam area
20. Menyelesaikan misi *level* 3	
	21. Permainan selesai
Skenario Alternatif	
1. Tekan multiplayer	
	2. Inisialisasi entitas permainan
	3. Tampil permainan
4. Mulai permainan	
5. Kontrol permainan	
	6. Inisialisasi kontrol permainan
	7. Tampil kursor di permainan
8. Tekan karakter	
	9. Inisialisasi menu karakter
	10. Tampil menu karakter
11. Pilih menu karakter	
	12. Jika sudah pernah memilih menu karakter dalam satu giliran, maka sistem tidak akan merespon kembali
	13. Karakter bereaksi sesuai menu karakter
	14. Update kontrol permainan

15. Tekan area dekat karakter utama	
	16. Jika area terdapat karakter lain, maka sistem tidak dapat menginisialisasi menu summon
	17. Inisialisasi menu summon
	18. Tampil menu summon
19. Pilih menu summon	
	20. Jika jumlah power summon tidak terpenuhi, maka sistem tidak akan merespon
	21. Jika jumlah kastil tidak terpenuhi, maka sistem tidak akan merespon
	22. Jika sudah pernah memilih menu summon dalam satu giliran, maka sistem tidak akan merespon kembali
	23. Inisialisasi karakter summon
	24. Tampil karakter summon di dalam area permainan
25. Pilih menu End	
	26. Sistem akan menyimpan informasi permainan yaitu area permainan dan posisi karakter
	27. Sistem akan menyudahi giliran dengan mengganti info permainan
	28. Sistem menjalankan AI pada karakter musuh
	29. Sistem akan menggerakan dan

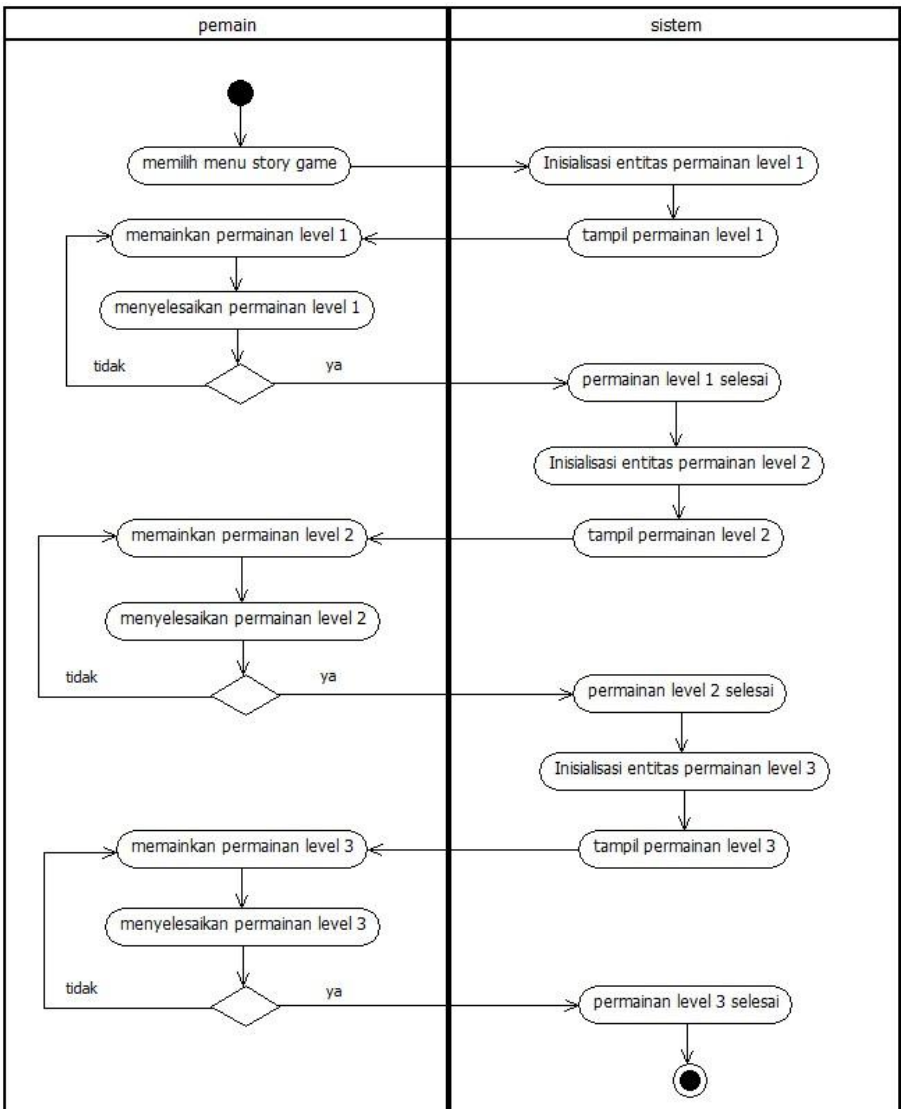
	membuat karakter musuh menyerang
	30. Sistem menyimpan informasi permainan yaitu arena permainan dan posisi karakter
	31. Sistem mengganti menyudahi giliran dan mengganti info permainan
32. Menyelesaikan misi permainan	
	33. Jika permainan *level* 3 tidak selesai, maka sistem tidak akan bias melanjutkan ke permainan selanjutnya
	34. Permainan selesai
Kondisi Akhir	Mengalahkan seluruh karakter musuh dalam submenu *level* 3

3.1.5.4 *Activity* Diagram

Activity diagram mempunyai penjelasan lebih mengenai urutan suatu proses yang terjadi antara aktor dan sistem. Aktivitas yang pertama dimulai dari aktor, kemudian akan mendapat respon dari sistem yang nantinya akan menampilkan hasil dari proses tersebut.

1. ***Activity* Diagram *Story Game***

Activity diagram ini menjelaskan alur dari aktivitas *story game* antar aktor dan sistem yang dijelaskan pada gambar berikut ini:

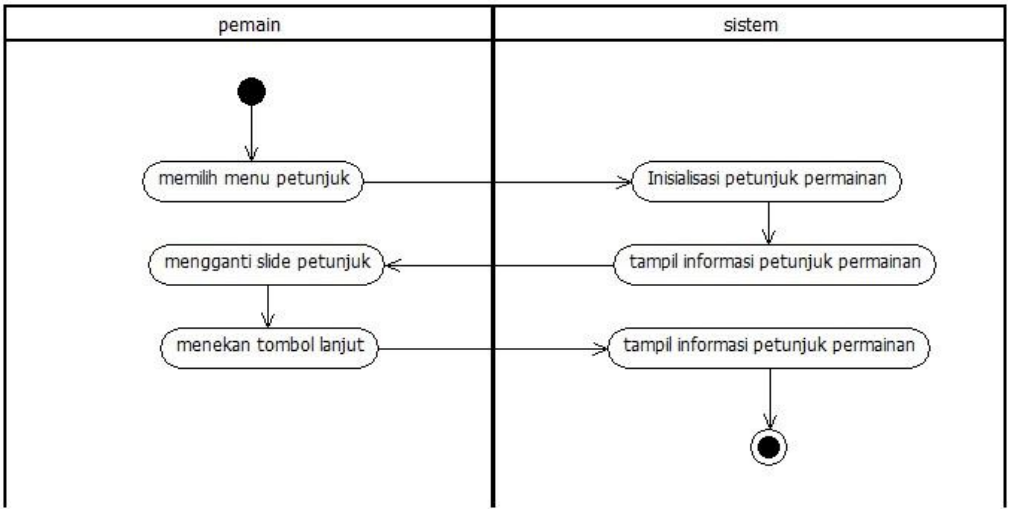


Gambar 3.22 Diagram *Activity Story Game*

Aktivitas ini terjadi saat pemain memililih menu *story game* pada halaman utama permainan *game the kingdom of* pandawa. Sistem akan menampilkan arena permainan, kemudian pemain dapat mulai memainkan permainan level 1 dan menyelesaikan permainan level 1. Jika permainan level 1 telah selesai, sistem akan langsung menampilkan permainan level 2, kemudian pemain dapat memainkan dan menyelesaikan permainan level 2. Jika permainan level 2 telah selesai, maka sistem akan langsung menampilkan permainan yang terakhir yaitu level 3. Setelah permainan level 3 tampil pemain dapat menyelesaikan permainan ini.

2. *Activity* Diagram *How to play*

Activity diagram ini menjelaskan alur dari aktivitas *How to play* antar aktor dan sistem yang dijelaskan pada gambar berikut ini :

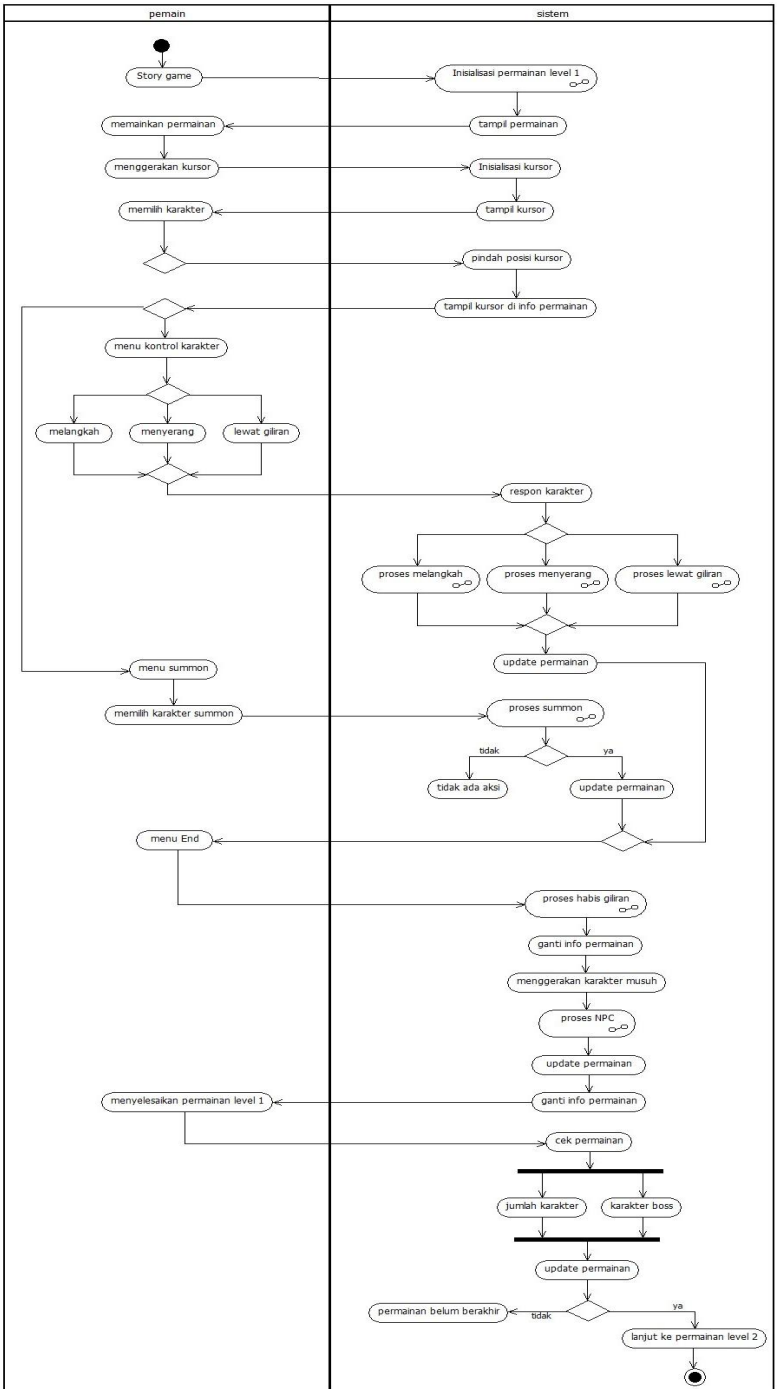


Gambar 3.23 Diagram *Activity How to play*

Aktivitas ini terjadi saat pemain memilih menu *how to play* pada halaman utama permainan *the kingdom of* pandawa. Sistem akan menampilkan petunjuk permainan, kemudian pemain dapat mengganti slide petunjuk dengan menekan tombol lanjut. Setelah tombol lanjut di tekan maka sistem akan menampilkan petunjuk yang lainnya.

3. *Activity Diagram level* 1

Activity diagram ini menjelaskan alur dari aktivitas *level* 1 antar aktor dan sistem yang dijelaskan pada gambar berikut ini:



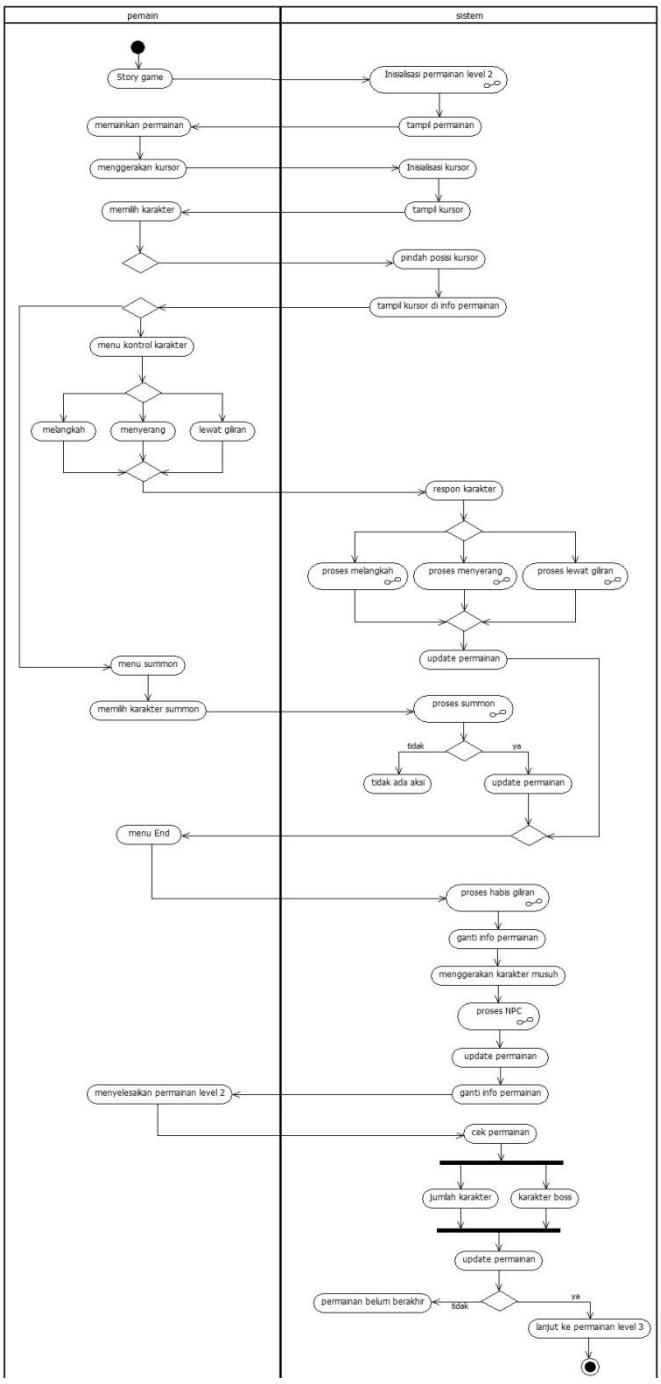
Gambar 3.24 Diagram *Activity level* 1

Aktivitas ini terjadi saat pemain masuk ke permainan dan tampil pada halaman pertama pada permainan *story game*. Sistem akan menampilkan arena permainan, kemudian pemain dapat memainkannya dengan cara mengerakan kursor di dalam permainan. Ketika kursor memilih karakter maka kursor akan

pindah ke mengarah ke info permainan yang di dalamnya terdapat menu kontrol dan menu summon. Ketika pemain memilih menu kontrol maka pemain dapat memilih kontrol karakter tersebut yang diantaranya melangkah, menyerang, dan melewati giliran. Setelah kontrol karakter sudah dipilih maka karakter utama akan melakukan aksi yang sesuai dengan kontrol karakter yang dipilih. Dan ketika pemain memilih menu summon maka sistem akan memproses inisialisasi karakter untuk muncul dan permainan akan update summon karakter. Setelah proses kontrol dan summon dilakukan maka sistem akan melakukan update giliran karakter tersebut. Setelah giliran kelompok karakter pemain berakhir maka pada giliran karakter musuh akan menjalankan AI untuk melakukan perpindahan dan melakukan penyerangan, kemudian setelah semua kelompok karakter musuh sudah menggunakan gilirannya, maka sistem akan mengganti giliran kepada karakter pemain kembali, pemain dapat melanjutkan permainan ke level 2 setelah pemain mengalahkan seluruh karakter musuh dan karakter boss.

4. *Activity Diagram level* 2

Activity diagram ini menjelaskan alur dari aktivitas *level* 2 antar aktor dan sistem yang dijelaskan pada gambar berikut ini :



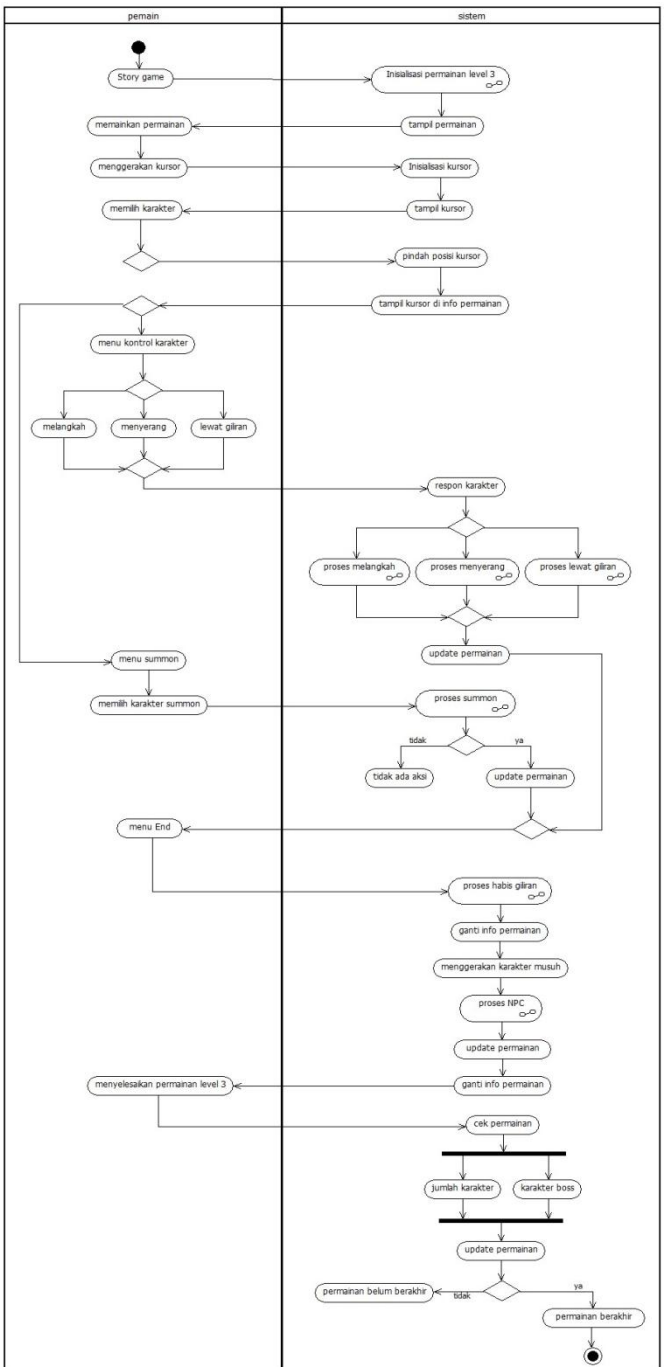
Gambar 3.25 Diagram *Activity level* 2

Aktivitas ini terjadi saat pemain masuk ke permainan dan tampil pada halaman kedua pada permainan *story game*. Sistem akan menampilkan arena permainan, kemudian pemain dapat memainkannya dengan cara mengerakan kursor di dalam permainan. Ketika kursor memilih karakter maka kursor akan

pindah ke mengarah ke info permainan yang di dalamnya terdapat menu kontrol dan menu summon. Ketika pemain memilih menu kontrol maka pemain dapat memilih kontrol karakter tersebut yang diantaranya melangkah, menyerang, dan melewati giliran. Setelah kontrol karakter sudah dipilih maka karakter utama akan melakukan aksi yang sesuai dengan kontrol karakter yang dipilih. Dan ketika pemain memilih menu summon maka sistem akan memproses inisialisasi karakter untuk muncul dan permainan akan update summon karakter. Setelah proses kontrol dan summon dilakukan maka sistem akan melakukan update giliran karakter tersebut. Setelah giliran kelompok karakter pemain berakhir maka pada giliran karakter musuh akan menjalankan AI untuk melakukan perpindahan dan melakukan penyerangan, kemudian setelah semua kelompok karakter musuh sudah menggunakan gilirannya, maka sistem akan mengganti giliran kepada karakter pemain kembali, pemain dapat melanjutkan permainan ke level 3 setelah pemain mengalahkan seluruh karakter musuh dan karakter boss.

5. *Activity Diagram level* 3

Activity diagram ini menjelaskan alur dari aktivitas *level* 3 antar aktor dan sistem yang dijelaskan pada gambar berikut ini :



Gambar 3.26 Diagram *Activity level* 3

Aktivitas ini terjadi saat pemain masuk ke permainan dan tampil pada halaman ketiga atau halaman terakhir pada permainan *story game*. Sistem akan menampilkan arena permainan, kemudian pemain dapat memainkannya dengan cara mengerakan kursor di dalam permainan. Ketika kursor memilih karakter

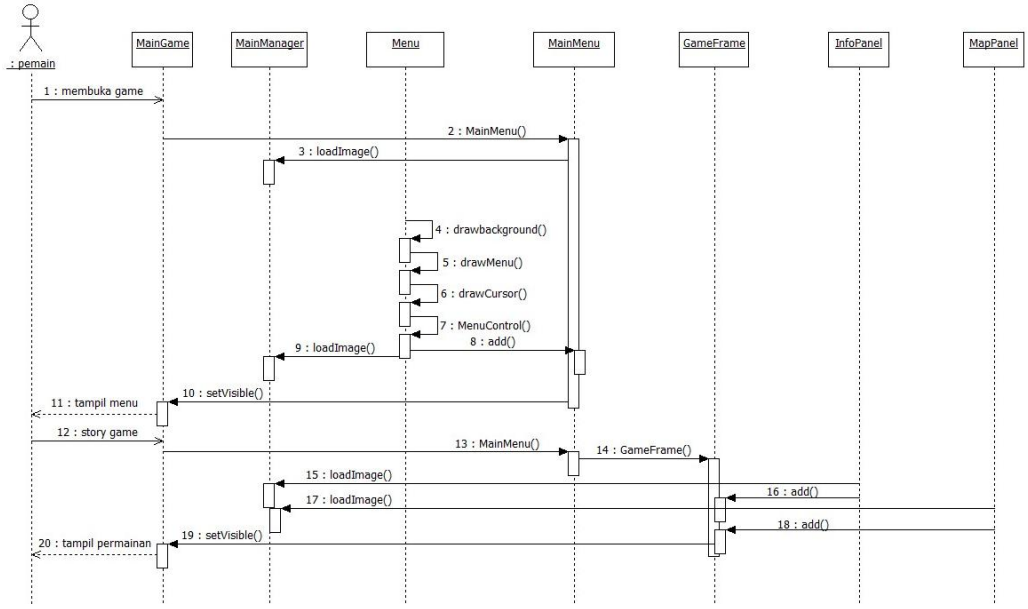
maka kursor akan pindah ke mengarah ke info permainan yang di dalamnya terdapat menu kontrol dan menu summon. Ketika pemain memilih menu kontrol maka pemain dapat memilih kontrol karakter tersebut yang diantaranya melangkah, menyerang, dan melewati giliran. Setelah kontrol karakter sudah dipilih maka karakter utama akan melakukan aksi yang sesuai dengan kontrol karakter yang dipilih. Dan ketika pemain memilih menu summon maka sistem akan memproses inisialisasi karakter untuk muncul dan permainan akan update summon karakter. Setelah proses kontrol dan summon dilakukan maka sistem akan melakukan update giliran karakter tersebut. Setelah giliran kelompok karakter pemain berakhir maka pada giliran karakter musuh akan menjalankan AI untuk melakukan perpindahan dan melakukan penyerangan, kemudian setelah semua kelompok karakter musuh sudah menggunakan gilirannya, maka sistem akan mengganti giliran kepada karakter pemain kembali, pemain dapat menyelesaikan permainan setelah pemain mengalahkan seluruh karakter musuh dan karakter boss.

3.1.5.5 *Sequence Diagram*

Sequence diagram menjelaskan kelakuan objek pada *use case* dengan mendeskripsikan berdasarkan waktu hidup dan pesan yang dikirim dan diterima antar objek.

1. ***Sequence* Diagram *Story Game***

   Sequence diagram ini menjelaskan urutan interaksi dari *use case story game* antar objek dan pesan yang dijelaskan pada gambar berikut ini :

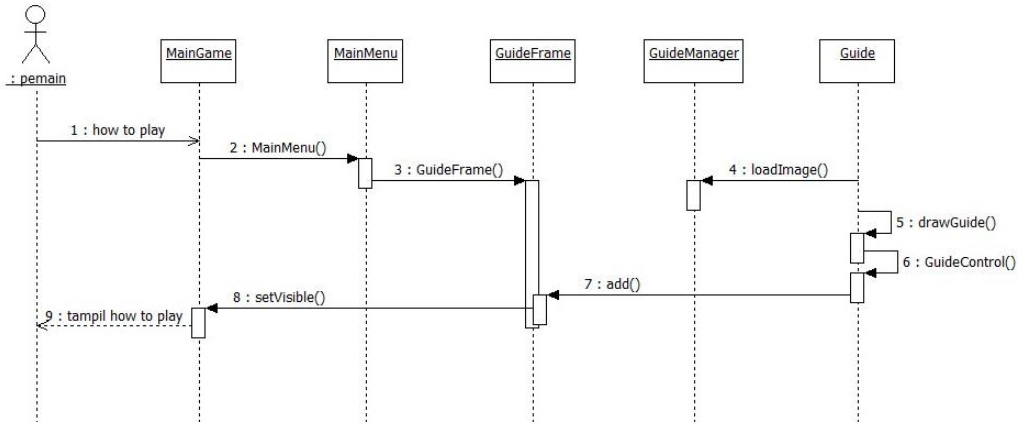


Gambar 3.27 *Sequence* Diagram *Story Game*

Urutan interaksi ini terjadi ketika pemain membuka permainan dan mengaktifkan main() dalam MainGame. Setelah MainGame aktif, MainGame akan mengirmkan sinyal kepada Loading untuk mengaktifkan showSplash(), lalu mengirimkan sinyal yang sama ke SplashScreen untuk mengaktifkan showSplash() yang berbeda. Kemudian MainGame pun mengirimkan sinyal kepada MainMenu yang ditujukan untuk mengaktifkan Menu dan MenuControl. Setelah itu MainMenu pun dikirim kembali ke MainGame dengan mengaktifkan MainMenu() dan tampilan utama pun tampil. Setelah tampil pemain pun memilih menu *Story Game* yang direspon oleh MainGame kembali, MainGame pun mengirimkan kembali respon tersebut ke MainMenu yang ditujukan untuk mengaktifkan Frame. Setelah aktif Frame pun mengirimkan Frame() kembali ke MainMenu yang akan dipakai kembali oleh MainGame, dan menu permainan *Story Game* pun akan tampil.

2. *Sequence* Diagram *How to Play*

Sequence diagram ini menjelaskan urutan interaksi dari *use case How to play* antar objek dan pesan yang dijelaskan pada gambar berikut ini :

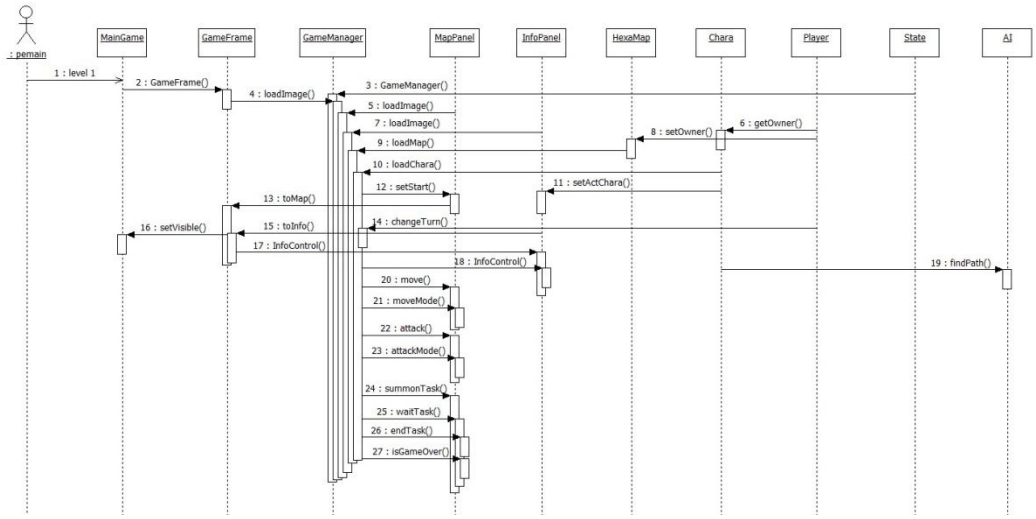


Gambar 3.28 *Sequence* Diagram *How to Play*

Urutan interaksi ini terjadi ketika pemain membuka menu *how to play* dan mengaktifkan main() dalam MainGame. Kemudian MainGame pun mengirimkan sinyal kepada MainMenu yang ditujukan untuk mengaktifkan MainMenu. Setelah itu MainMenu pun mengirim kembali ke GuideFrame dan tampilan utama pun tampil. Setelah tampil pemain pun tampilan dapat berubah dengan mengaktifkan *method* GuideControl() yang akan mememunculkan gambar yang berbeda setelah terjadi peristiwa tekan tombol. Kemudian GameFrame akan menangkap semua respon yang dikerjakan oleh Guide dan di tampilkan di layar.

3. *Sequence* Diagram *level* 1

Sequence diagram ini menjelaskan urutan interaksi dari *use case level* 1 antar objek dan pesan yang dijelaskan pada gambar berikut ini :

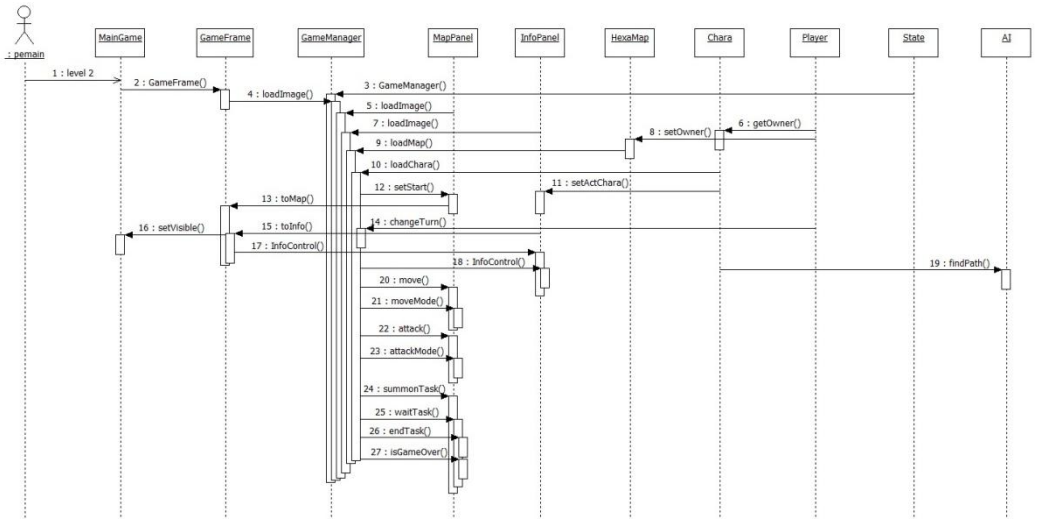


Gambar 3.29 *Sequence* Diagram *level* 1

Urutan interaksi ini terjadi dari MainGame yang langsung direspon dari GameFrame. Pertama-tama Game Frame akan menginisialisasi gambar melewati GameManager dan inisialisasi itu diantaranya, loadImage(), loadMap, loadChara. Disini GameManager adalah tempat inisialisasi yang di pakai untuk mengatur MapPanel, InfoPanel, Chara, State. Posisi arena permainan diatur oleh GameManager yang diaktifkan dari MapPanel dan InfoPanel, setelah itu pengaturan posisi karakter akan dilakukan oleh GameManager yang diaktifkan dari Chara dan HexaMap. Kemudian GameManager dapat mengatur giliran karakter yang dilakukan oleh State. Dalam MapPanel informasi karakter dan aksi karakter dapat aktif setelah Chara mengaktifkan setActChara yang diatur di InfoPanel. Aksi yang terjadi kemudian di tampilkan oleh GameFrame oleh method kepemilikannya yaitu toMap, toInfo, toUpdate.

4. *Sequence* Diagram *level* 2

Sequence diagram ini menjelaskan urutan interaksi dari *use case level* 2 antar objek dan pesan yang dijelaskan pada gambar berikut ini :

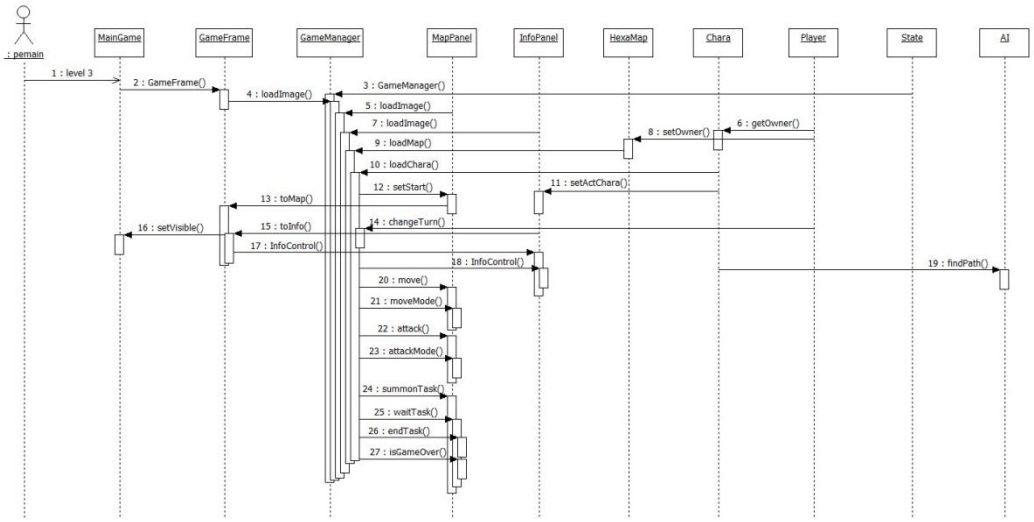


Gambar 3.30 *Sequence* Diagram *level* 2

Urutan interaksi ini terjadi dari MainGame yang langsung direspon dari GameFrame. Pertama-tama Game Frame akan menginisialisasi gambar melewati GameManager dan inisialisasi itu diantaranya, loadImage(), loadMap, loadChara. Disini GameManager adalah tempat inisialisasi yang di pakai untuk mengatur MapPanel, InfoPanel, Chara, State. Posisi arena permainan diatur oleh GameManager yang diaktifkan dari MapPanel dan InfoPanel, setelah itu pengaturan posisi karakter akan dilakukan oleh GameManager yang diaktifkan dari Chara dan HexaMap. Kemudian GameManager dapat mengatur giliran karakter yang dilakukan oleh State. Dalam MapPanel informasi karakter dan aksi karakter dapat aktif setelah Chara mengaktifkan setActChara yang diatur di InfoPanel. Aksi yang terjadi kemudian di tampilkan oleh GameFrame oleh method kepemilikannya yaitu toMap, toInfo, toUpdate.

5. Sequence Diagram *level* 3

Sequence diagram ini menjelaskan urutan interaksi dari *use case level* 3 antar objek dan pesan yang dijelaskan pada gambar berikut ini :



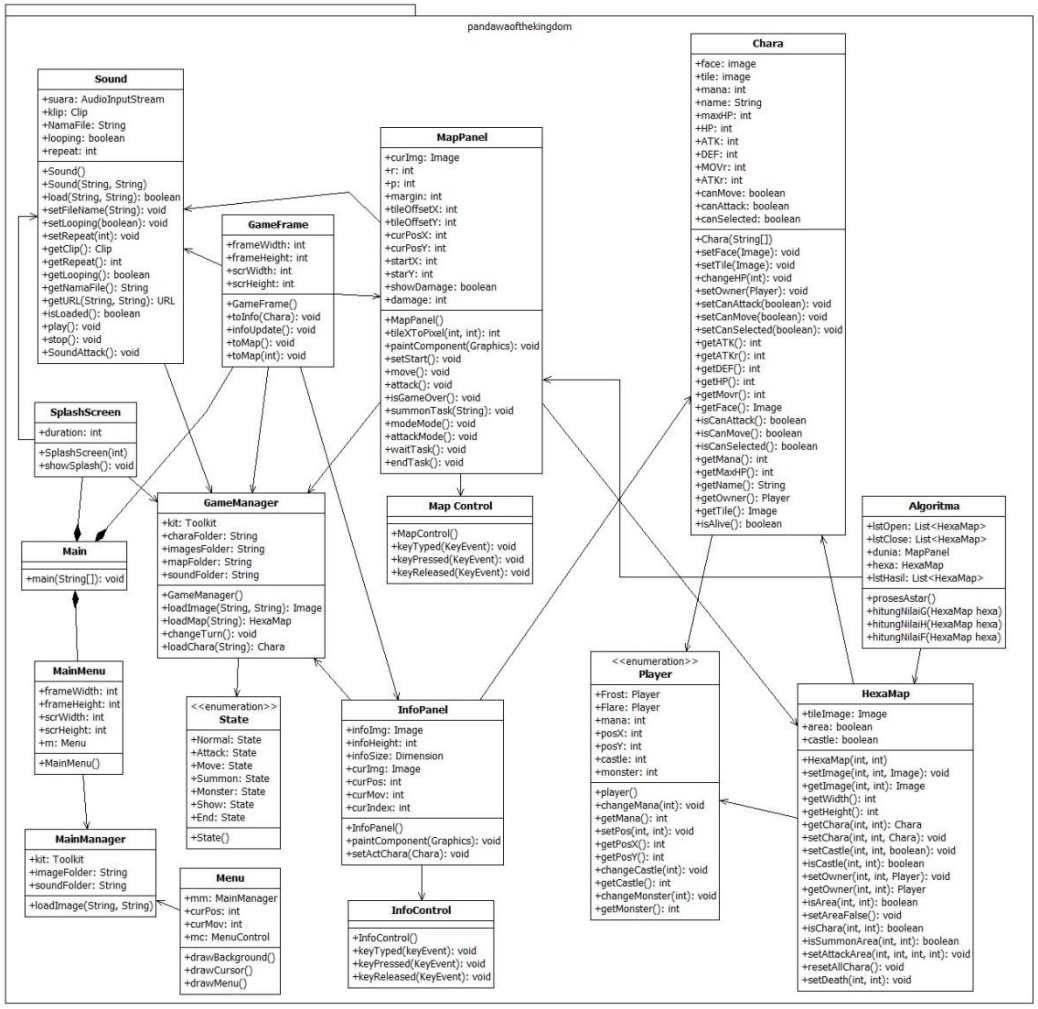
Gambar 3.31 *Sequence* Diagram *level* 3

Urutan interaksi ini terjadi dari MainGame yang langsung direspon dari GameFrame. Pertama-tama Game Frame akan menginisialisasi gambar melewati GameManager dan inisialisasi itu diantaranya, loadImage(), loadMap, loadChara. Disini GameManager adalah tempat inisialisasi yang di pakai untuk mengatur MapPanel, InfoPanel, Chara, State. Posisi arena permainan diatur oleh GameManager yang diaktifkan dari MapPanel dan InfoPanel, setelah itu pengaturan posisi karakter akan dilakukan oleh GameManager yang diaktifkan dari Chara dan HexaMap. Kemudian GameManager dapat mengatur giliran karakter yang dilakukan oleh State. Dalam MapPanel informasi karakter dan aksi karakter dapat aktif setelah Chara mengaktifkan setActChara yang diatur di InfoPanel. Aksi yang terjadi kemudian di tampilkan oleh GameFrame oleh method kepemilikannya yaitu toMap, toInfo, toUpdate.

3.1.5.6 *Class Diagram*

Class diagram menjelaskan relasi di dalam sistem dengan mendeskripsikan relasi antar kelas-kelas atau paket-paket.

Adapun *Class* Diagram pada *game the kingdom of* pandawa akan dijelaskan pada gambar berikut ini :



Gambar 3.32 Diagram *Class the kingdom of* Pandawa

Relasi antar kelas ini terjadi dari 16 kelas yang saling berelasi. Relasi yang terdapat disini terdapat 13 relasi asosiasi dan 3 relasi komposisi.

3.2 Perancangan *Game The Kingdom of* Pandawa

3.2.1 Perancangan Entitas *Game The Kingdom of* Pandawa

Beberapa entitas yang terlibat dalam *game the kingdom of* pandawa adalah sebagai berikut :

3.2.1.1 Entitas Karakter Pemain

Table 3.24 Entitas karakter pemain

No	Nama	Gambar	Keterangan
1	Yudhistira		Ini adalah karakter utama atau sekaligus karakter pemimpin di pandawa yang bernama Yudhistira. Karakter ini mempunyai *health point* 1200, *attack power* 60, *defend power* 60, *move range* 2, *attack range* 6.
2	Bima		Ini adalah salah satu karakter *summon* dalam kelompok pandawa. Karakter ini mempunyai *health point* 700, *attack power* 60, *defend power* 55, *move range* 4, *attack range* 2.
3	Nakula		Ini adalah salah satu karakter *summon* dalam kelompok Pandawa. Karakter ini mempunyai *health point* 1200, *attack power* 80, *defend power* 60, *move range* 4, *attack range* 1.
4	Sadewa		Ini adalah salah satu karakter *summon* dalam kelompok Pandawa. Karakter ini mempunyai *health point* 900, *attack power* 50, *defend power* 90, *move range* 4, *attack range* 1.
5	Arjuna		Ini adalah salah satu karakter *summon* dalam kelompok pandawa. Karakter ini mempunyai *health point* 850, *attack power* 110, *defend power* 65, *move range* 5, *attack range* 4.

3.2.1.2 Entitas karakter Musuh

Table 3.25 Entitas karakter musuh

No	Nama	Gambar	Keterangan
1	Abimanyu		Ini adalah karakter musuh dalam kelompok kurawa. karakter ini akan dilawan oleh pemain di level terakhir. Karakter ini mempunyai *health point* 1000, *attack power* 70, *defend power* 60, *move range* 2, *attack range* 6.
2	Destradyumna		Ini adalah karakter musuh dalam kelompok kurawa. karakter ini akan di lawan oleh pemain di level dua. Karakter ini mempunyai *health point* 500, *attack power* 40, *defend power* 55, *move range* 6, *attack range* 4.
3	Seta		Ini adalah karakter musuh dalam kelompok kurawa. karakter ini akan di lawan oleh pemain di level dua. Karakter ini mempunyai *health point* 900, *attack power* 100, *defend power* 70, *move range* 5, *attack range* 4.
4	Kresna		Ini adalah karakter musuh dalam kelompok kurawa. karakter ini akan di lawan oleh pemain di level pertama. Karakter ini mempunyai *health point* 700, *attack power* 70, *defend power* 80, *move range* 5, *attack range* 3.
5	Adipati Karna		Ini adalah karakter musuh dalam kelompok kurawa. karakter ini akan mendampingi Abimanyu di level

			terakhir. Karakter ini mempunyai *health point* 800, *attack power* 60, *defend power* 50, *move range* 4, *attack range* 2.
6	Soldier		Ini adalah karakter musuh yang menjadi prajurit kurawa. karakter ini akan dilawan di level pertama hingga akhir. Karakter ini mempunyai *health point* 100, *attack power* 20, *defend power* 20, *move range* 3, *attack range* 3.

3.2.1.3 Entitas Menu Kontrol

Table 3.26 Entitas menu kontrol

No	Nama	Gambar	Keterangan
1	Move		Ini adalah navigasi yang berfungsi untuk memindahkan posisi karakter dalam arena permainan.
2	Attack		Ini adalah navigasi yang berfungsi untuk membuat karakter pemain menyerang karakter musuh dalam arena permainan.
3	Wait		Ini adalah navigasi yang berfungsi untuk melewat giliran karakter pemain.
4	End Turn		Ini adalah navigasi yang berfungsi untuk mengakhiri giliran karakter pemain.

3.2.1.4 Entitas Arena Permainan

Table 3.27 Entitas arena permainan

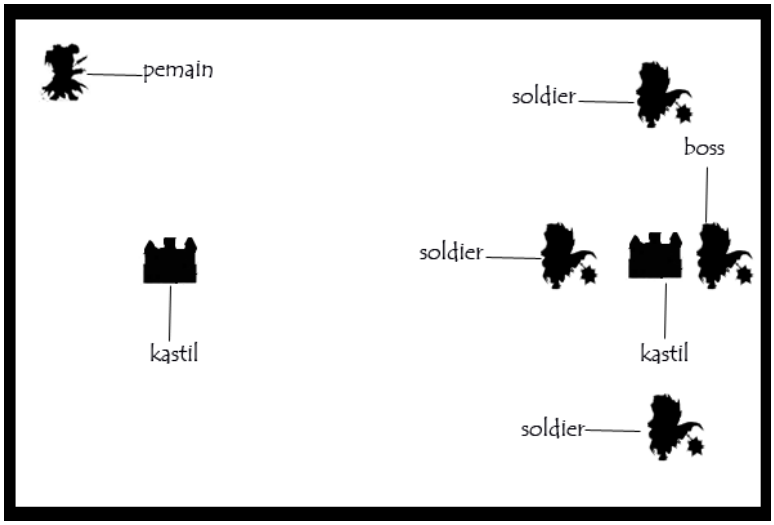
No	Nama	Gambar	Keterangan
1	*Castil*		Ini adalah *castil* yang harus di rebut oleh Pandawa, castil ini pun berfungsi sebagai syarat dari berapa banyak pemain dapat summon (dimunculkan) karakter lain dalam permainan.
2	arena		Ini adalah arena yang berfungsi untuk menjadi tempat dimana seluruh karakter bermain.

3.2.1.5 Perancangan *Story Board*

Story board merupakan sketsa gambar yang akan disusun berurutan sesuai dengan naskah dan tingkat kesulitan, dengan *story board* kita dapat menggambarkan ide cerita kita kepada orang lain dengan mudah. Karena kita dapat menggiring pemahaman seseorang mengikuti gambar-gambar yang ada, sehingga menghasilkan pemahaman yang sama pada ide cerita kita. Berikut merupakan *story board* dari *the kingdom of* pandawa:

1. *Story Board* level 1

Story board untuk level 1 pada *game* ini dapat dilihat sebagai berikut :

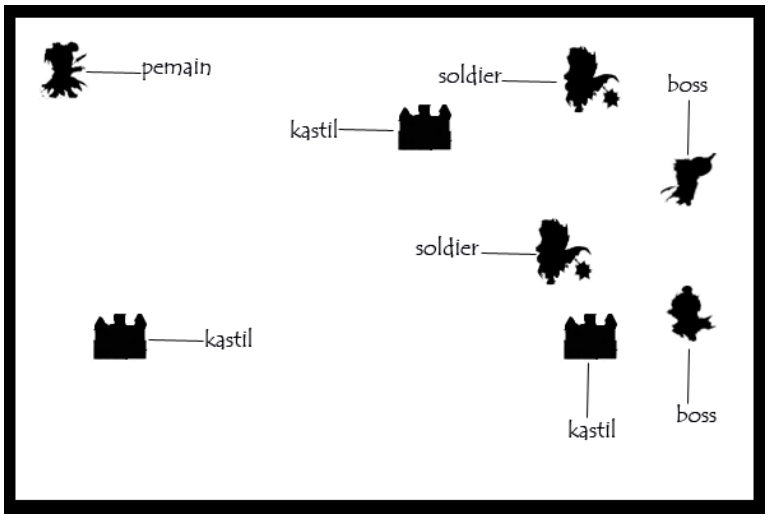


Gambar 3.33 *Story Board* level 1

Pada *story board* level 1 musuh sudah diterapkan algoritma A*. pemain akan tampil di pojok kiri atas kemudian musuh akan tampil pada sebelah kanan layar. Tingkat kesulitan pada level 1 ini dipengaruhi oleh jumlah musuh yang harus dikalahkan oleh pemain yaitu berjumlah 3 prajurit musuh dan 1 *boss* musuh dan posisi kerajaan yang harus dimiliki pemain.

2. *Story Board* level 2

Story board untuk level 2 pada *game* ini dapat dilihat sebagai berikut :

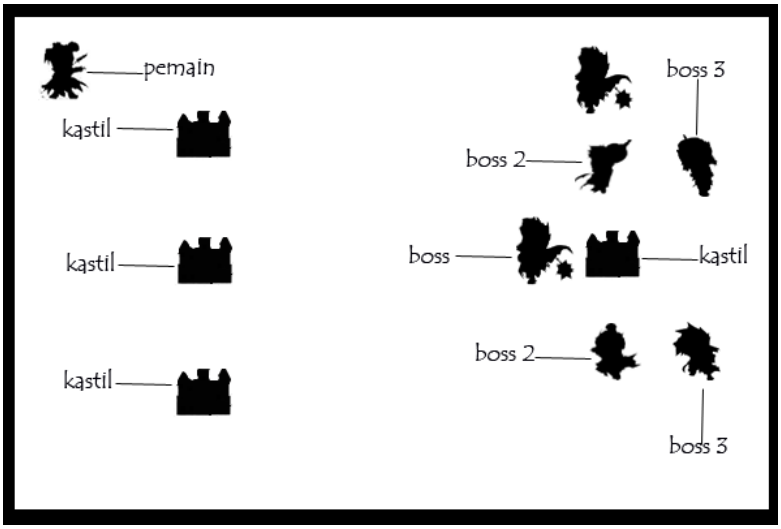


Gambar 3.34 *Story Board* level 2

Pada *story board* level 1 musuh sudah diterapkan algoritma A*. pemain akan tampil di pojok kiri atas kemudian musuh akan tampil pada sebelah kanan layar. Tingkat kesulitan pada level 1 ini dipengaruhi oleh jumlah musuh yang harus dikalahkan oleh pemain yaitu berjumlah 2 prajurit musuh dan 2 *boss* musuh dan posisi kerajaan yang harus dimiliki pemain.

3. *Story Board* level 3

Story board untuk level 2 pada *game* ini dapat dilihat sebagai berikut :



Gambar 3.35 *Story Board* level 3

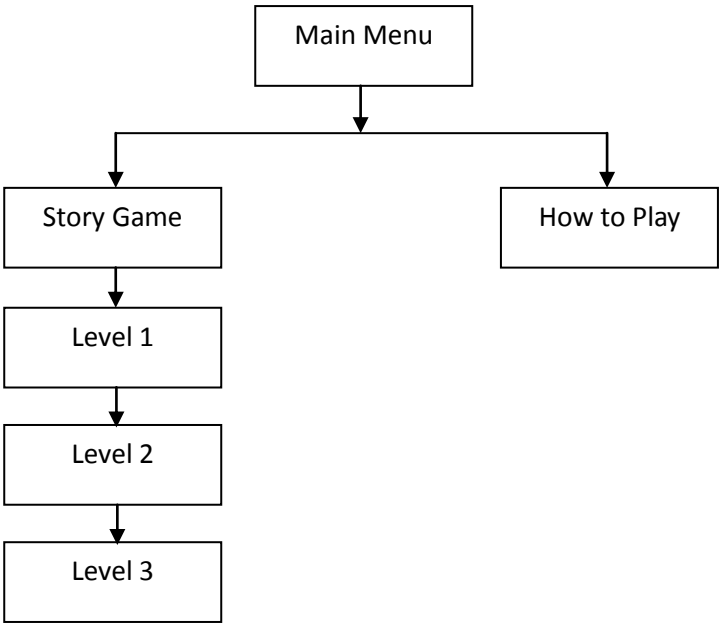
Pada *story board* level 1 musuh sudah diterapkan algoritma A*. pemain akan tampil di pojok kiri atas kemudian musuh akan tampil pada sebelah kanan layar. Tingkat kesulitan pada level 1 ini dipengaruhi oleh jumlah musuh yang harus dikalahkan oleh pemain yaitu berjumlah 5 *boss* musuh dan posisi kerajaan yang harus dimiliki pemain.

3.2.2 Perancangan Perangkat Lunak

Perancangan perangkat lunak merupakan kegiatan untuk merancang aplikasi yang akan di bangun, tahapannya dimulai dari perancangan arsitektur menu, kemudian berlanjut ke perancangan antar muka dan jaringan semantik.

3.2.3 Perancangan Arsitektur Menu

Untuk mempermudah dalam pembuatan sistem, diperlukan perancangan struktur menu program yang akan dibangun. Perancangan struktur menu program ini membantu dalam merancang bagian-bagian dari sistem yang sebenarnya dan untuk mengetahui bagian mana yang nantinya akan diakses terlebih dahulu setelah program selesai dibuat. Perancangan arsitektur menu pada *game the kingdom of* pandawa dapat dilihat pada gambar berikut:



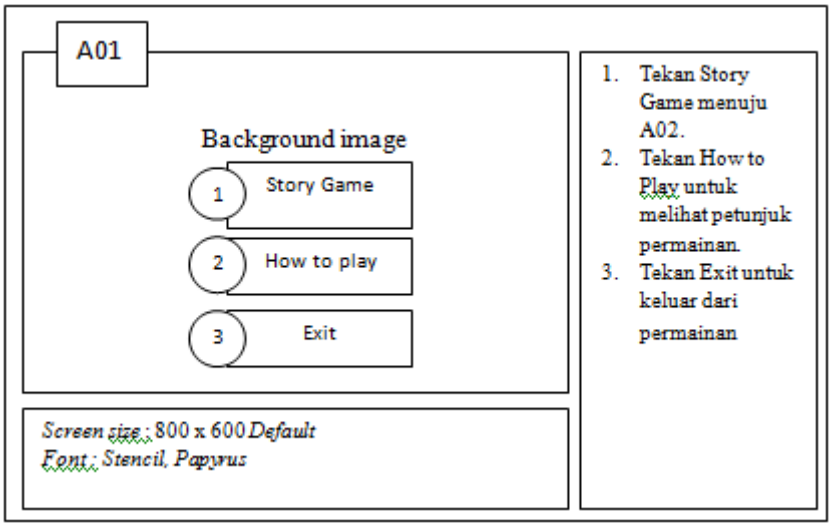
Gambar 3.36 Arsitektur Menu *Game The Kingdom of* Pandawa

3.2.4 Perancangan Antar Muka

Perancangan antar muka bertujuan untuk memberikan gambaran tentang aplikasi yang akan dibangun, sehingga akan mempermudah dalam mengimplementasikan aplikasi serta akan mempermudah pembuatan aplikasi. Berikut ini merupakan perancangan antar muka pada *game The Kingdom of* Pandawa :

1. **Perancangan Antar Muka Menu Utama**

   Perancangan antar muka Menu Utama pada *game* ini dapat dilihat sebagai berikut:

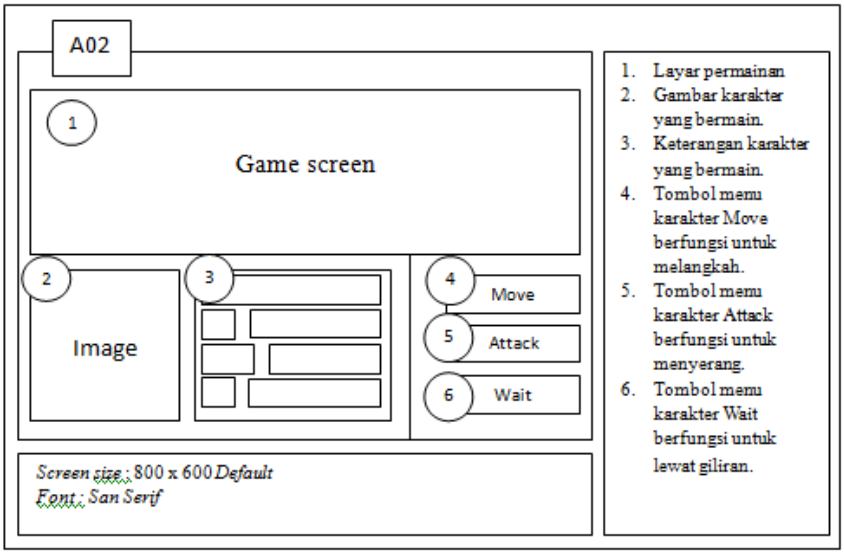


Gambar 3.37 Perancangan Antar Muka Menu Utama

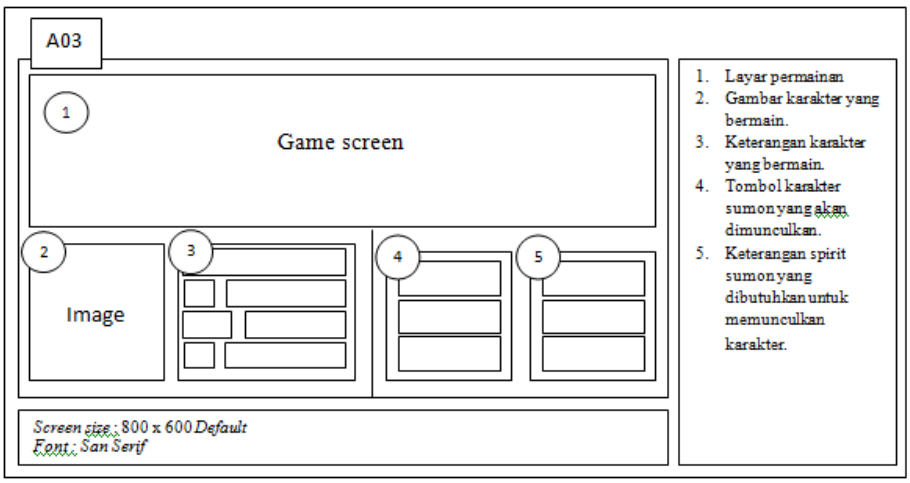
2. Perancangan Antar Muka Menu Level

Perancangan antar muka Menu Level pada *game* ini dapat dilihat sebagai berikut:

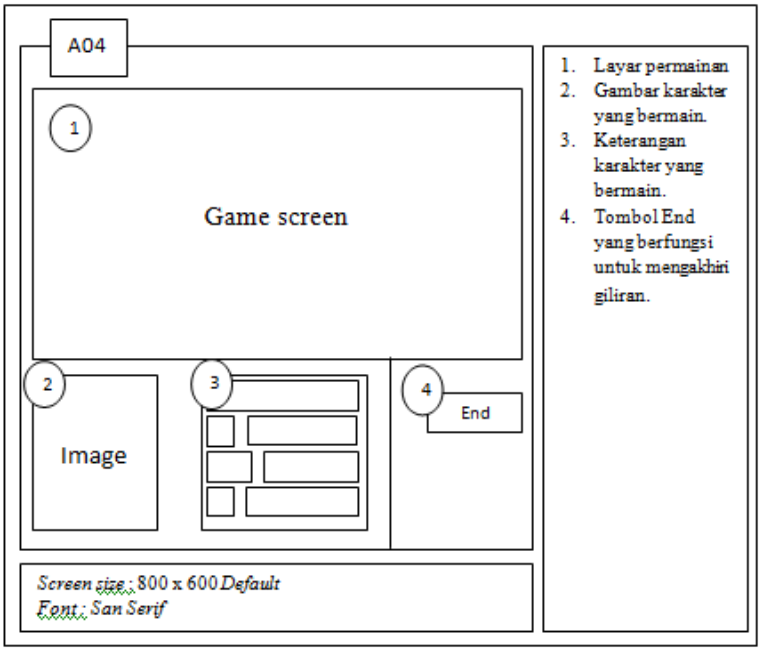
1. Perancangan Antar Muka Level 1



Gambar 3.38 Perancangan Antar Muka Level 1

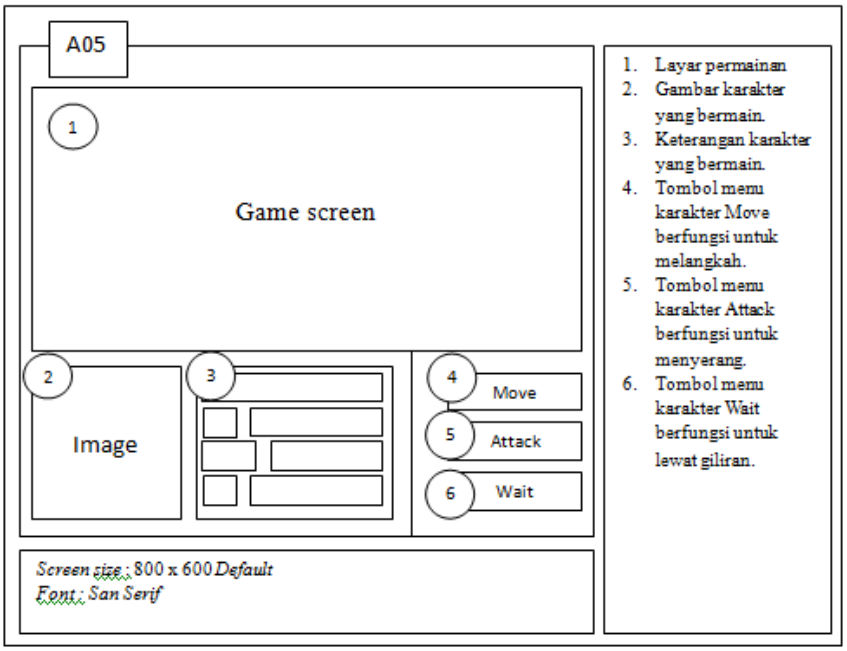


Gambar 3.39 Perancangan Antar Muka Level 1

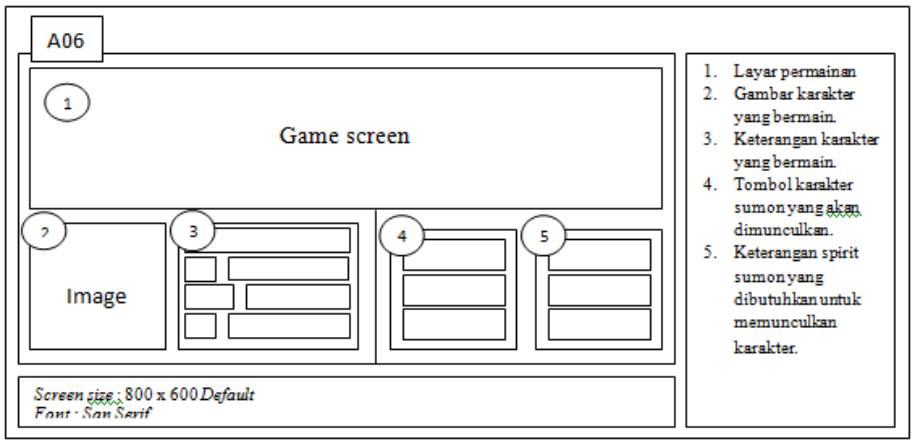


Gambar 3.40 Perancangan Antar Muka Level 1

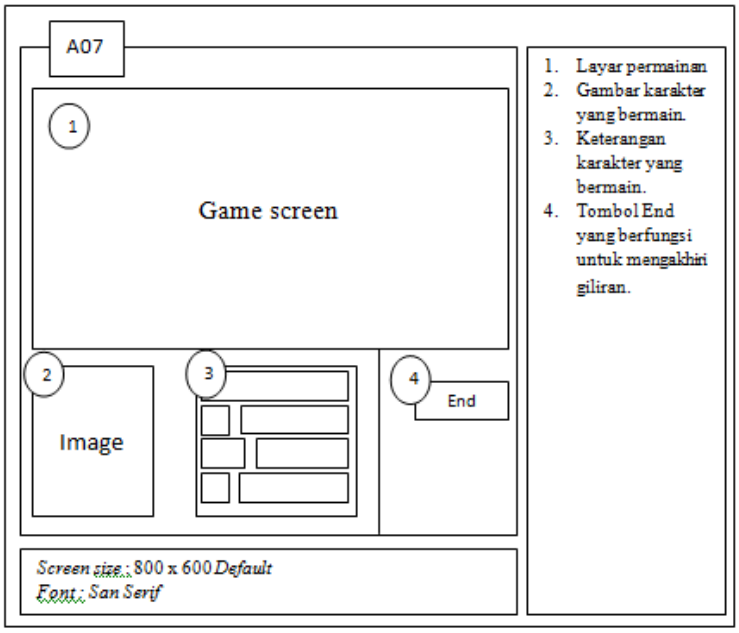
2. Perancangan Antar Muka Level 2



Gambar 3.41 Perancangan Antar Muka Level 2

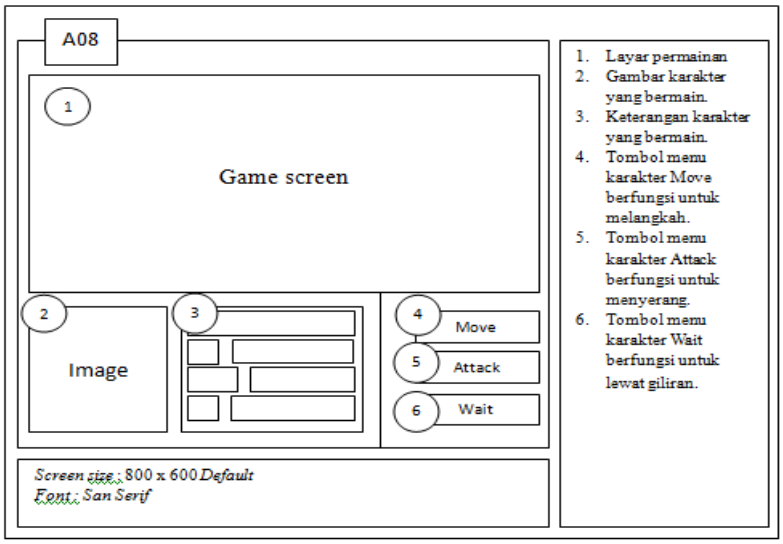


Gambar 3.42 Perancangan Antar Muka Level 2

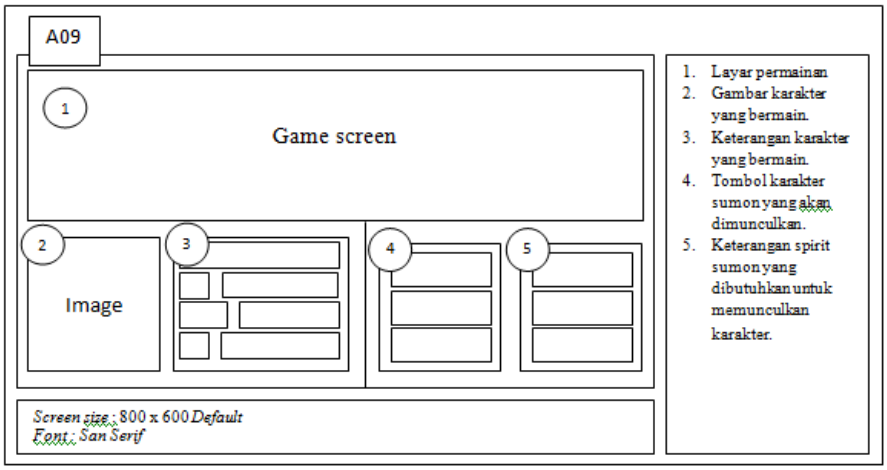


Gambar 3.43 Perancangan Antar Muka Level 2

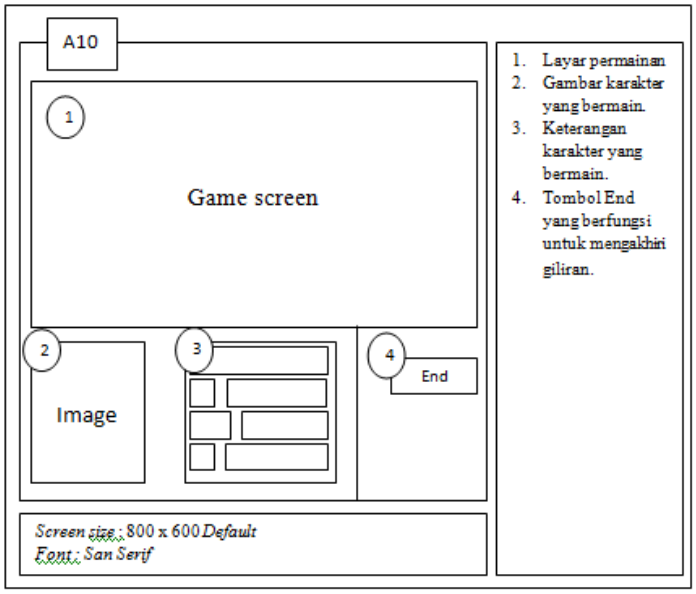
3. Perancangan Antar Muka Level 3



Gambar 3.44 Perancangan Antar Muka Level 3



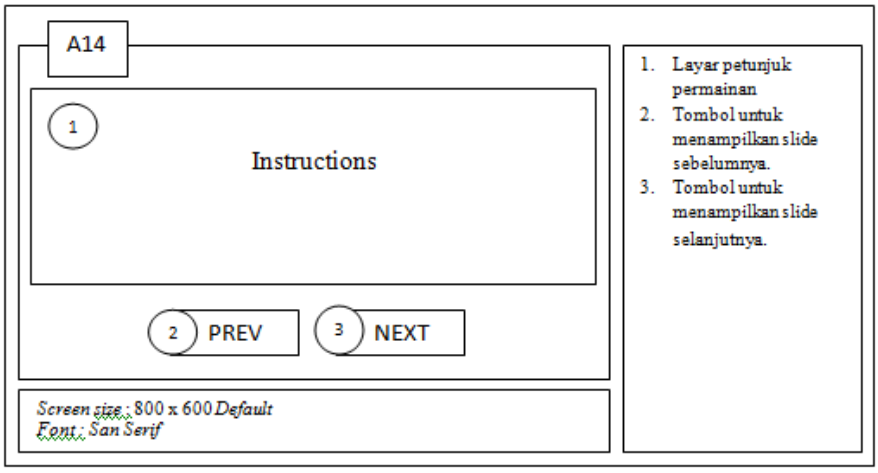
Gambar 3.45 Perancangan Antar Muka Level 3



Gambar 3.46 Perancangan Antar Muka Level 3

3. Perancangan Antar Muka *How to play*

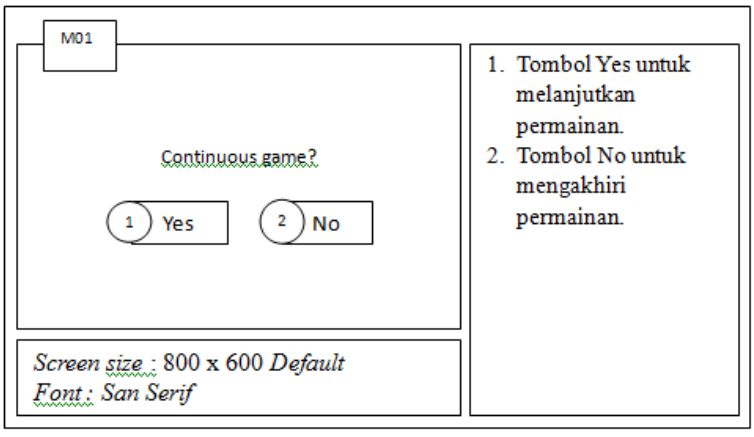
Perancangan antar muka *How to play* pada aplikasi ini dapat dilihat sebagai berikut :



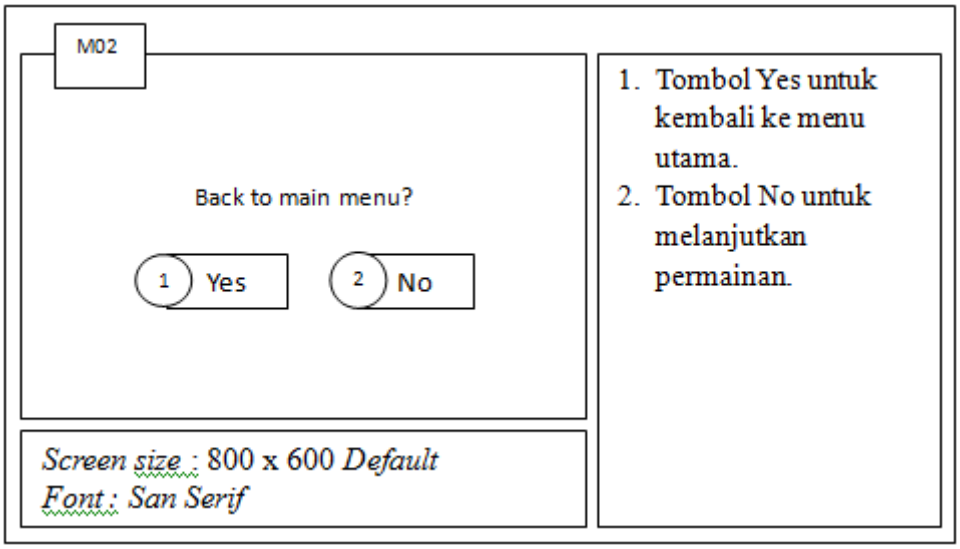
Gambar 3.47 Perancangan Antar Muka *How to play*

4. Perancangan Antar Muka Pesan

Perancangan antar muka pesan pada aplikasi ini dapat dilihat sebagai berikut :



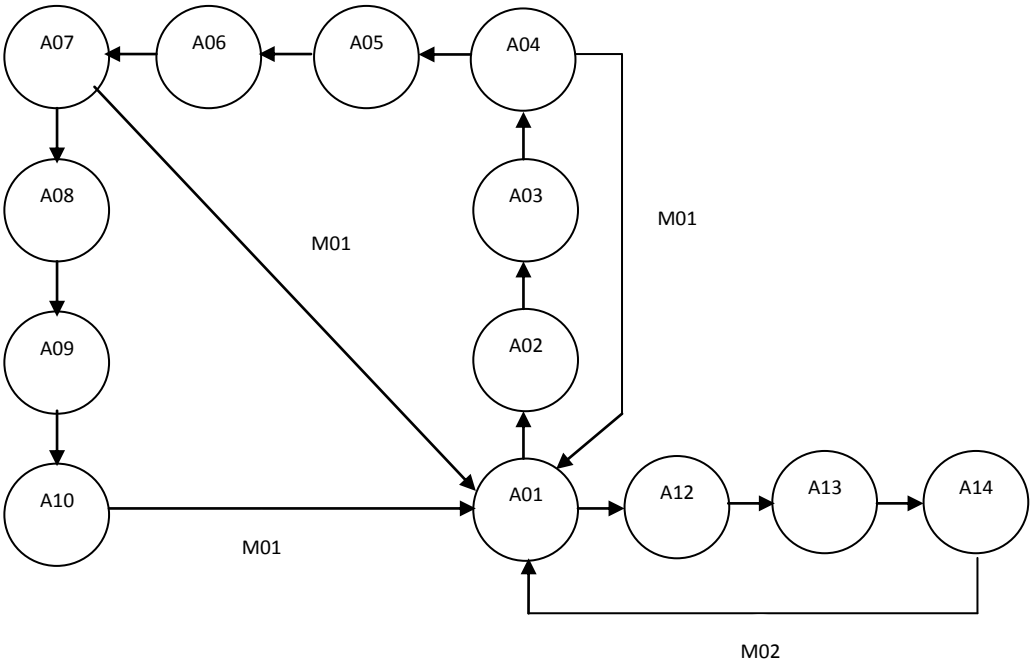
Gambar 3.48 Perancangan Antar Muka Peringatan



Gambar 3.49 Perancangan Antar Muka Peringatan

3.2.5 Jaringan Semantik

Jaringan semantik adalah gambaran tentang relasi dari masing- masing halaman yang bisa diakses oleh seorang pengguna. Jaringan semantik ini, digunakan sebagai panduan penggunaan menu-menu dalam sebuah aplikasi.



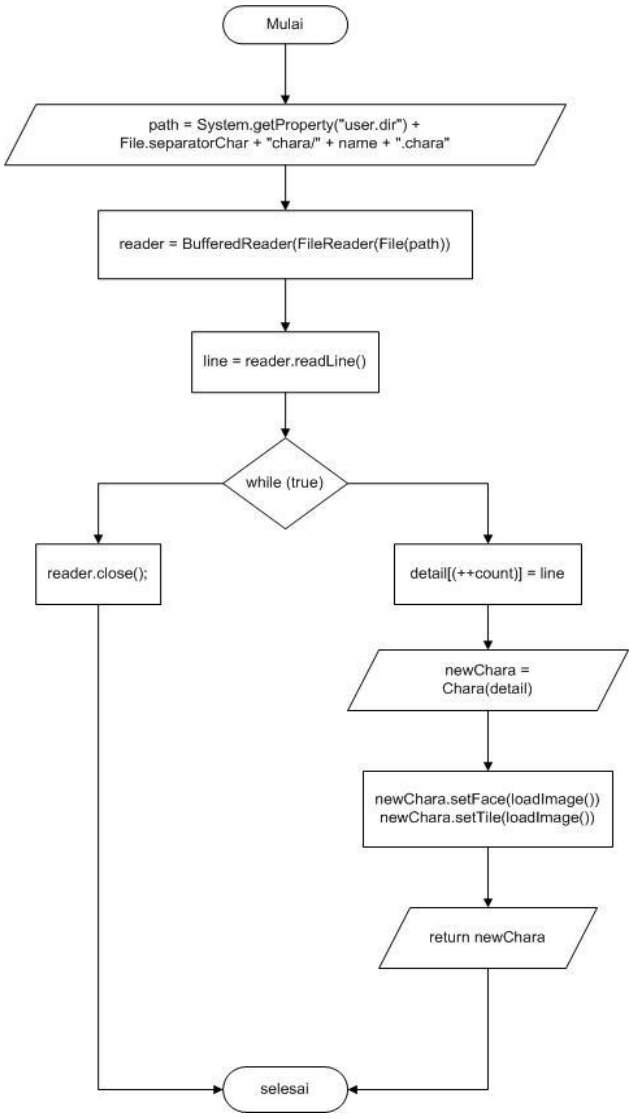
Gambar 3.50 Jaringan Semantik

3.2.6 Perancangan *Method*

Perancangan *method* merupakan perncangan yang berfungsi untuk mendeskripsikan *method-method* yang berada di dalam aplikasi. *Method* dapat dipanggil dengan menyertakan variabel, baik hanya satu variabel, banyak variabel atau bahkan tidak ada sama sekali. Adapun *method-method* yang terdapat dalam *game the kingdom of* pandawa yang akan dibangun adalah sebagai berikut :

1. **Method Inisialisai Karakter**

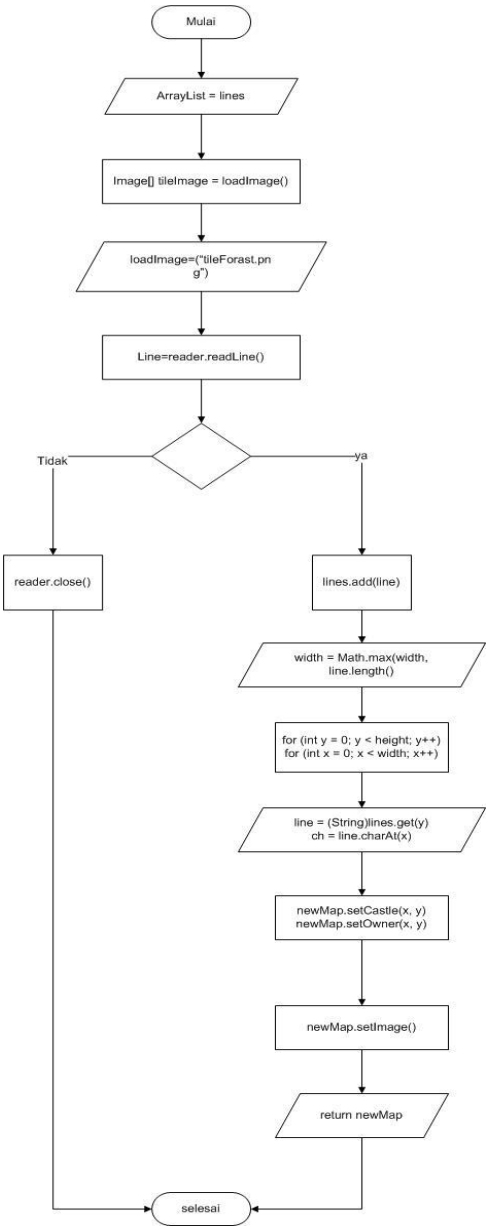
   Method ini digunakan untuk menginisialisasi karakter.



Gambar 3.51 *Flowchart Method Inisialisasi Karakter*

2. Method Inisialisasi *Map*

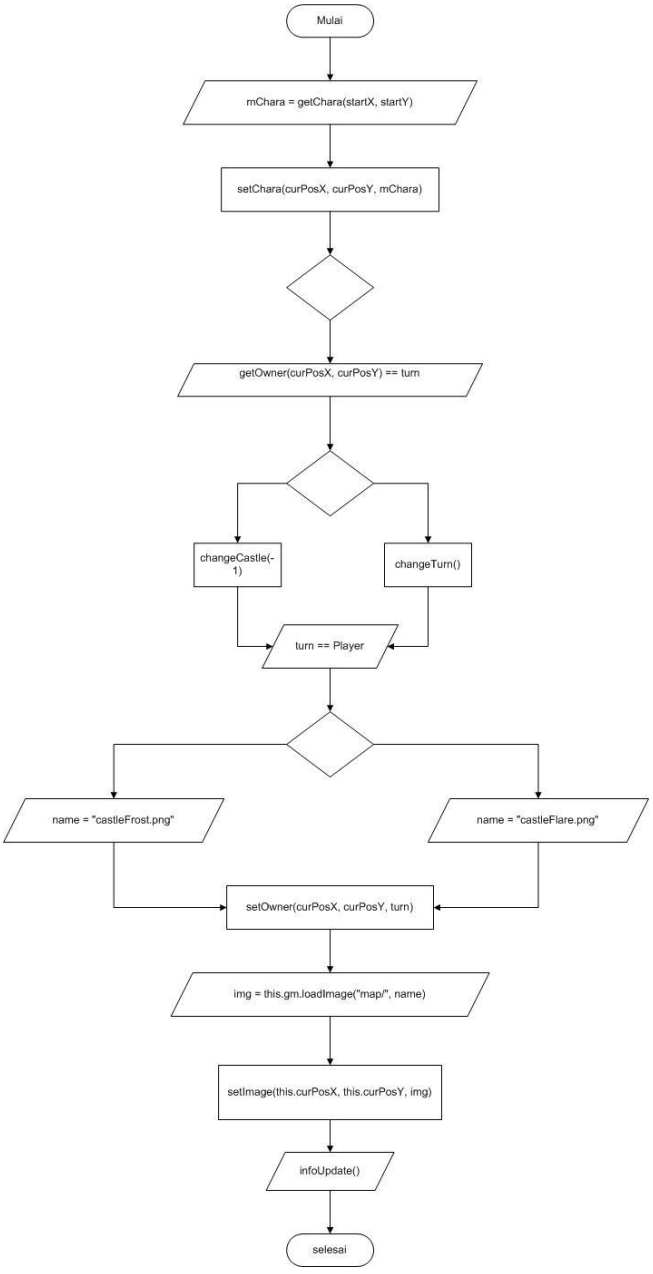
Method ini digunakan untuk menginisialisasi *map*.



Gambar 3. 52 *Flowchart Method Inisialisasi Map*

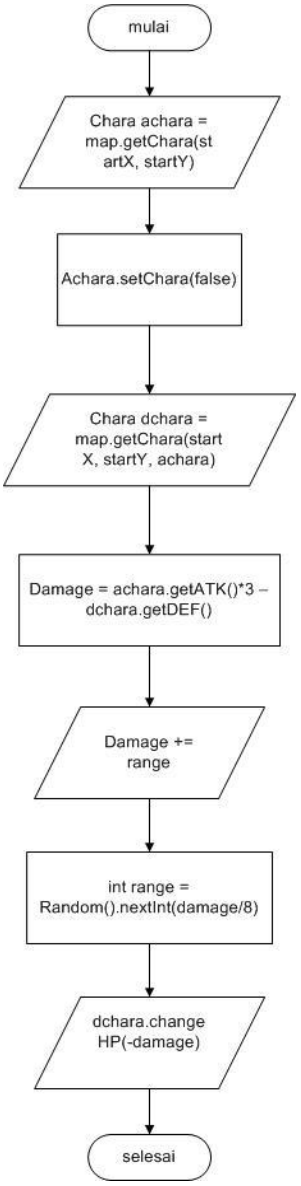
3. Method *Move*

Method ini digunakan untuk menggerakkan karakter.



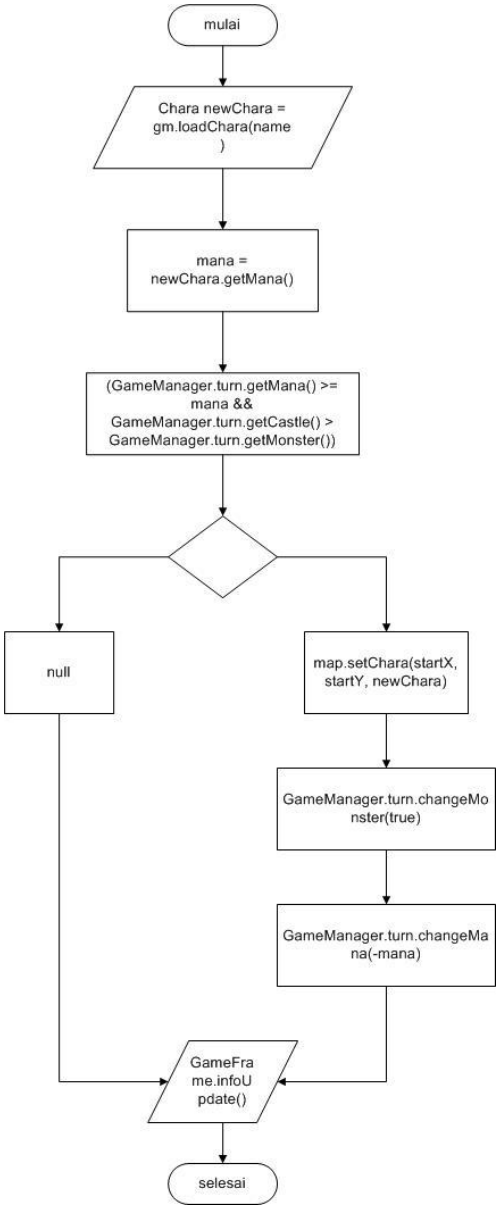
Gambar 3.53 *Flowchart Method Move*

4. Method *Attack*



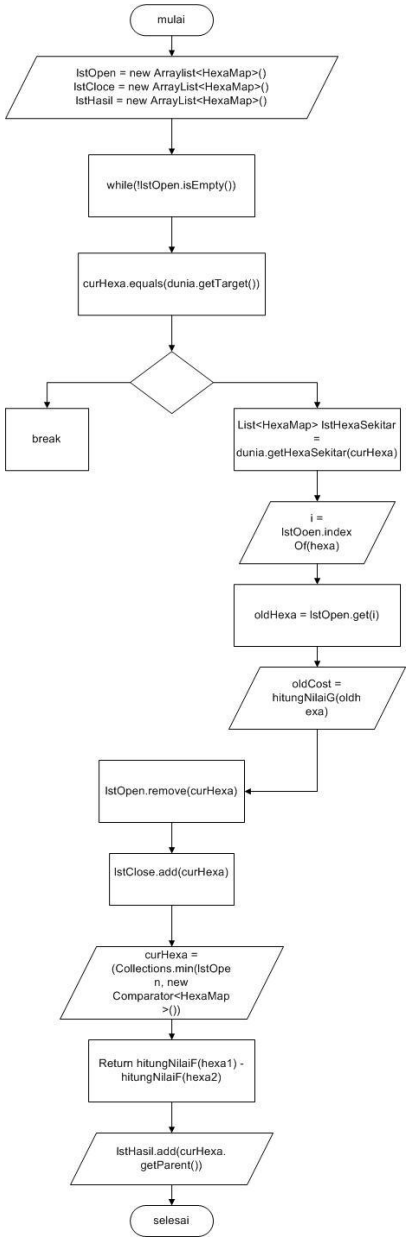
Gambar 3.54 *Flowchart Method Attack*

5. Method *Summon*



Gambar 3.55 *Flowchart Method Summon*

6. Method Pathfinding



Gambar 3.56 *Flowchart Method Pathfinding*