BAB 3
ANALISIS DAN PERANCANGAN SISTEM

3.1 Analisis Sistem

Analisis sistem merupakan proses penguraian konsep ke dalam bagian-bagian yang lebih sederhana, sehingga struktur logisnya menjadi lebih jelas. Pada analisis sistem ini akan mencakup analisis masalah, analisis kebutuhan non-fungsional dan analisis kebutuhan fungsional. Metode digunakan untuk menguji, membandingkan, menilai, dan memahami sistem pemikiran yang kompleks dan memecahkan permasalahan dan berinteraksi untuk mencapai tujuan yang dicapai.

3.1.1 Analisis Masalah

Berdasarkan hasil wawancara dengan siswa-siswi di SMPN 2 dan SMAN 15 Bandung bahwa paradigma siswa-siswi memahami informasi tentang penanganan luka tidak baik. Siswa-siswi merasa pengetahuan terhadap penanganan luka dinilai masih kurang paham benar. Penanganan luka yang baik seharusnya dilakukan dengan prosedur-prosedur sesuai tindakan medis yang seharusnya merupakan tindakan mudah bagi orang awam sekalipun, tetapi pada pelaksanaannya masih banyak penanganan yang tidak sesuai sehingga banyak kasus terjadi akibat kesalahan dari penanganan luka itu sendiri.

Berdasarkan uraian latar belakang masalah tersebut, maka dapat dirumuskan beberapa masalahnya yaitu :

1. Masih kurangnya pengetahuan tentang cara penanganan luka pada tubuh.
2. Sosialisasi yang sudah ada masih belum tersampaikan dengan baik, sehingga masih banyak salah penanganan terhadap luka pada tubuh.

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.

Game Ella's Clinic merupakan *game* yang menceritakan tentang penyembuhan penyakit yang diderita pasien, *game* ini mengambil lokasi di sebuah klinik dimana Ella sebagai suster yang bertugas di klinik tersebut. Berikut tampilan menu pada *game* Ella's Clinic terlihat pada gambar 3.1



Gambar 3.1 Tampilan menu *game Ella's Clinic* [2]

Dalam *game* ini player berperan sebagai suster Ella yang bekerja di sebuah klinik, suster Ella bertugas untuk menyembuhkan para pasien yang datang ke kliniknya. Pemain diharuskan menyelesaikan misi dari setiap levelnya dan di setiap levelnya pemain diberi tantangan berupa *goal* poin, jika pemain berhasil menyelesaikan waktu yang diberikan untuk mencapai *goal* poin maka pemain telah menyelesaikan misi di level tersebut.

Game ini memiliki beberapa keterbatasan yaitu, dari penyampaian informasi yang terkandung di dalam *game* ini kurang tersampaikan dengan baik kepada penggunanya. Pemain tidak mengetahui penyakit apa yang diderita oleh pasien yang datang ke klinik tersebut dan bagaimana cara penyembuhannya. Dapat dilihat dari hasil komentar para pengguna *game* Ella's clinnic pada gambar 3.2.



Gambar 3.2. Hasil Komentar Pengguna *Game* [1]

Sedangkan kelebihan dari *game* ini yaitu dari sisi grafik, dimana pemain disuguhkan dengan tampilan *game* yang menarik mulai dari pemilihan warna hingga pemilihan karakter, fakta tersebut dapat dilihat dari gambar 3.3.

Evaluasi yang dapat diambil dari *game* Ella’s clinic yaitu dengan menerapkan kelebihan *game* tersebut dari sisi grafiknya, dan menyempurnakan kekurangan dari *game* ini yaitu dengan menambahkan informasi tentang luka dan cara penanganannya.



Gambar 3.3 Tampilan *game* Ella’s Clinic [1]

3.1.3. Analisis *Game* yang dibangun

Game Jaram Perlaya yang akan dibangun yaitu *game* yang mempunyai grafis 2D yang ber-*genre Casual Game. Game* akan dibuat berdasarkan hasil evaluasi dari game sejenis yang telah ada. *Game* ini menceritakan tentang perjalanan seorang dokter untuk menyembuhkan para pasien yang terkena luka pada tubuhnya. Beberapa fitur yang ditawarkan di dalam *game* ini :

1. *Single player*
2. *Game* dibuat dengan grafis 2D

3.1.4. *Storyline*

Game yang akan dibangun dengan satu pemain yang dijalankan seorang *actor* yang bernama dr. Mimo. Dokter Mimo merupakan tokoh utama dalam permainan ini, awalnya dokter mimo bekerja di rumah sakit ternama hingga akhirnya dia dipindah tugaskan ke pedesaan karena kelalaiannya, tetapi dr.Mimo mempunyai tekad yang kuat agar bisa kembali lagi ke rumah sakit itu. Lalu dia bekerja dengan gigih di desa yang dia singgahi demi untuk bisa kembali lagi menjadi dokter di rumah sakit tersebut. *Actor* tersebut harus menyelesaikan 3 *level* yang tiap-tiap *level*nya memiliki rintangan yang berbeda-beda.

3.1.5. *Gameplay*

Konsep dari permainan game ini berbeda jika dibandingkan dengan game sejenis yang sudah ada. Pada game ini pemain akan dihadapkan dengan pasien yang berkunjung, tugas utama dari game ini adalah menyembuhkan pasien sebanyak-banyaknya sesuai dengan *goal* di tiap *level*-nya. Pemain akan diberikan waktu sesuai dengan tingkat *level*-nya, jika waktu telah habis maka permainan akan berhenti, lalu akan dihitung akumulasi poin yang didapat oleh pemain, apabila poin sesuai dengan *goal* maka pemain berhak melanjutkan ke *level* berikutnya dan apabila pemain tidak dapat memenuhi *goal* poin maka permainan akan berakhir dan mengulang lagi ke awal permainan. Setiap pasien yang datang memiliki gejala yang berbeda sehingga pemain harus bisa memilih apa yang harus dilakukan dengan kondisi pasien yang ada.

Level 1, Pemain akan diberi tantangan untuk menyelesaikan tugasnya. Pemain akan dihadapkan dengan pasien penderita luka gores, dan luka memar. Lokasi pada level 1 berada di puskesmas.

Level 2, Pada level kali ini pemain akan diberi tantangan untuk menyelesaikan tugasnya, dan pemain akan dihadapi dengan pasien penderita luka memar, luka sayat . Lokasi pada level 1 berada di klinik.

Level 3, Pada level terakhir ini pemain akan diberi tantangan untuk menyelesaikan tugasnya dan akan dihadapkan dengan pasien yang lebih kompleks seperti luka gores, luka sayat, luka memar, dan luka bakar. Lokasi pada level 1 berada di rumah sakit.

3.1.6. Analisis Tingkat Kesulitan

Tingkat kesulitan dari tiap *level* akan dibedakan dari jumlah *goal* poin dan delay penyembuhan pasien. Berdasarkan karakteristik, tamu dapat dibedakan menjadi tiga jenis, antara lain :

1. Tamu cepat(fst)
2. Tamu normal(nor)
3. Tamu lambat(slw)

Setiap jenis tamu memiliki tingkat kesabaran(sab), lama waktu yang diperlukan untuk menyembuhkan luka(sem), dan pemberian poin(pon) untuk pemain, yang berbeda-beda. Kesabaran yang masih dimiliki oleh tamu dilambangkan dengan icon hati

Tabel 3.1 Tabel properti dibagi per jenis tamu

	Sab(detik)	Sem(detik)	Pon
Slw	60	70	200
Nor	50	40	150
Fast	40	40	150

Perbandingan tingkat kesulitan dari tiap *level* dapat dilihat pada tabel 3.2.

Tabel 3.2. Tingkat Kesulitan Tiap *Level*

	Level 1	*Level* 2	*Level* 3
Goal Poin	1000	2000	3000
Pasien luka gores (waktu penyembuhan)	Cepat	Cepat	Cepat
Pasien luka sayat (waktu penyembuhan)	Normal	Normal	Cepat
Pasien luka memar (waktu penyembuhan)	-	Lambat	Normal
Pasien luka bakar (waktu penyembuhan)	-	-	Lambat

Pada tabel 3.2 dijelaskan bahwa tingkat kesulitan berdasarkan perbandingan gol poin dengan waktu untuk penyembuhan pasien. Pada tiap *level* terdapat empat jenis pasien yang akan dihadapi oleh pemain. Masing-masing pasien memiliki waktu untuk penyembuhan berdasarkan dari jenis luka dan kategori *level*-nya

3.1.7. Analisis *Scoring*

Sistem penilaian pada *game* ini apabila pemain berhasil menyembuhkan satu pasien akan mendapatkan 150 poin untuk pasien normal, cepat, dan 200 poin untuk pasien lambat dan jika pemain tidak dapat menyembuhkan satu pasien akan mengurangi 50 poin. Poin akan di akumulasikan di akhir game untuk mencapai goal di setiap levelnya.

3.1.8. Analisis Algoritma Branch and Bound

Dalam permainan ”Jaram Perlaya”, pemain berperan sebagai seorang dokter, yang bernama Mimo. Sebagai Mimo, pemain harus menyembuhkan berbagai jenis pasien yang berkunjung ke tempat bekerjanya, dengan jumlah ruang penyembuhan pasien yang terbatas. Tantangan yang dihadapi pemain adalah memperoleh poin sebanyak-banyaknya, dengan menentukan urutan tamu yang dilayani oleh dr.Mimo.

Berdasarkan tabel 3.1 dapat dilihat klasifikasi jenis tamu, dimana tamu cepat memiliki kesabaran 40 detik, waktu penyembuhan 40 detik, dan akan

mendapatkan poin sebesar 150. Tamu normal memiliki kesabaran 50 detik, waktu penyembuhan 40 detik, dan akan mendapatkan poin sebesar 150, sedangkan tamu lambat memiliki kesabaran 60 detik, waktu penyembuhan 70 detik, dan akan mendapatkan poin sebesar 200.

Poin yang diberikan oleh tamu (pon) adalah hasil penjumlahan poin yang diberikan ketika pasien mulai konsultasi, penyembuhan luka, pasien selesai berobat. Poin diakumulasikan ketika tamu sudah menyelesaikan proses berobat

Seperti telah disebutkan di atas, masalah perolehan keuntungan maksimum pada game "Jaram Perlaya", akan didekati dengan menggunakan algoritma B&B.

Properti :

1. Solusi permainan

   Permainan terselesaikan jika tidak ada lagi pasien yang dapat dilayani

2. Fungsi pembangkit

   Membangkitkan segala kemungkinan langkah pada setiap langkahnya. Contohnya, pada keadaan antrian dua tamu lambat, satu tamu cepat, dan dua buah ruangan penyembuhan, terdapat 2 kemungkinan langkah berikutnya, yaitu dua ruangan diisi dua tamu lambat dan dua ruangan diisi seorang tamu lambat dan seorang tamu cepat.

3. Fungsi pembatas

   Fungsi untuk membatasi sejauh mana algoritma B&B akan menelusuri setiap kemungkinan yang ada. Misalnya ada batasan bahwa banyak pasien yang dilayani tidak boleh melebihi ruangan penyembuhan yang tersedia.

Setiap simpul yang terdapat dalam pohon ruang solusi memiliki properti nomor simpul, satu buah tabel dan ongkos dari simpul tersebut. Nomor simpul adalah urutan pembangkitan dari simpul. Tabel berisi keadaan pasien yang mengantri dan pasien yang sedang diobati. Setiap isi dari sel tabel menyatakan tingkat kesabaran dari pasien yang sedang mengantri(sab), dan sisa waktu yang

dibutuhkan pasien yang sedang menyelesaikan pengobatannya(sem). Ruang penyembuhan yang kosong dinyatakan secara implisit di dalam tabel, berupa selisih banyak ruangan dan banyak pasien yang sedang diobati.

Tabel 3.3. Tabel yang Terdapat pada Simpul Hidup

	Cepat	Normal	Lambat
Penyembuhan (sem)	10		
Kesabaran (pat)		30	20,30

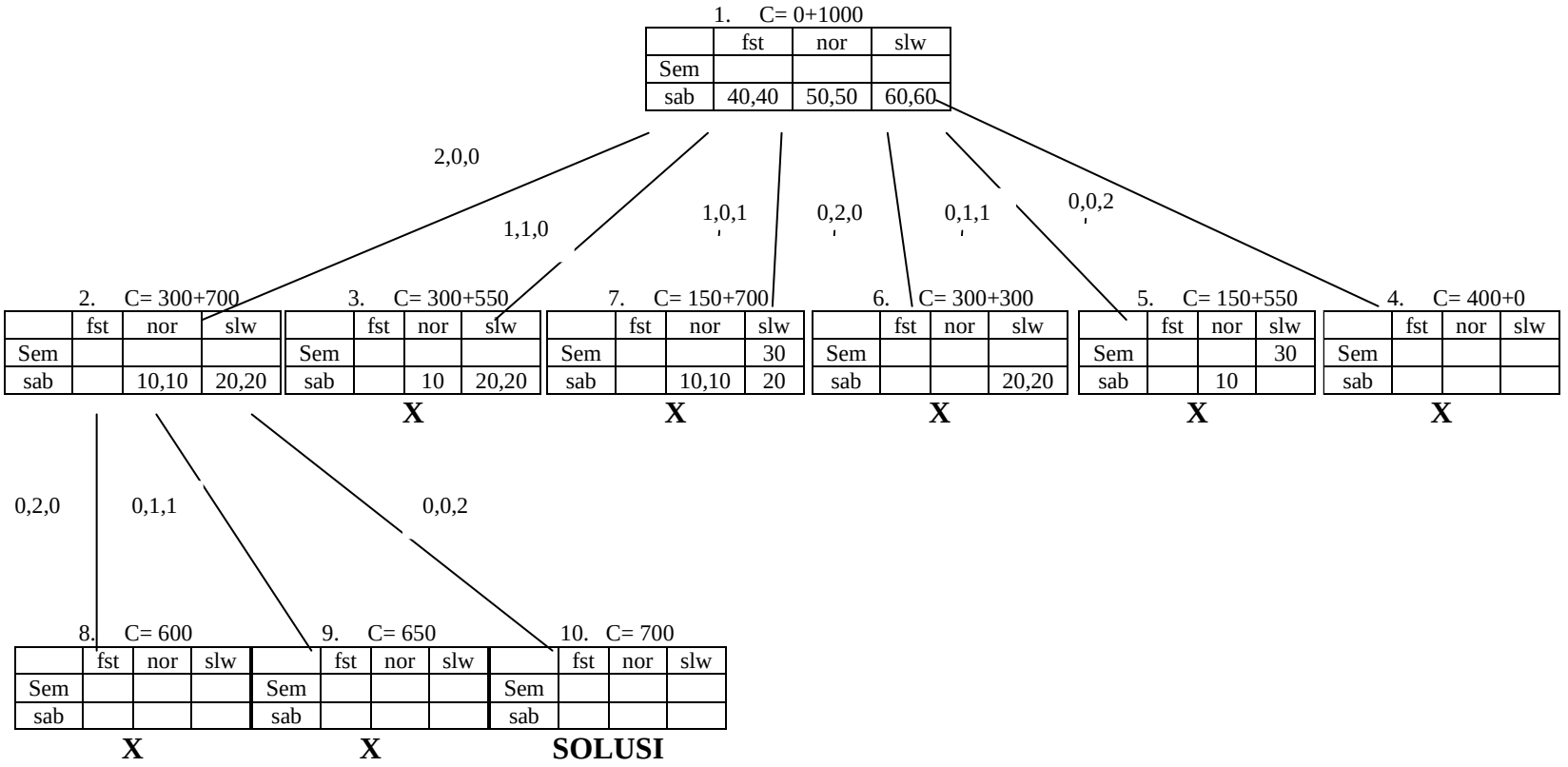
Contohnya, nilai 10 pada sel cepat, penyembuhan (sem) menyatakan terdapat satu pasien cepat yang sedang disembuhkan, dan akan menyelesaikan penyembuhannya dalam 10 detik. Nilai 20, 30 pada sel lambat, kesabaran (pat) menyatakan terdapat dua pasien lambat yang sedang mengantri dengan kesabaran tersisa masing-masing 20 dan 30. Jika kesabaran sudah habis, pasien akan meninggalkan rumah sakit. Jika banyak ruangan yang terdapat adalah 3, maka dengan melihat tabel 3.3, dapat diambil kesimpulan bahwa masih terdapat dua ruangan kosong yang dapat diisi oleh pasien yang sedang mengantri. Setiap simpul yang diperluas menyatakan keadaan di mana setidaknya satu pasien telah menyelesaikan pengobatannya, atau terdapat setidaknya satu ruangan yang kosong dan dapat ditempati oleh pasien yang sedang mengantri. Hal ini berarti kondisi waktu pada tiap simpul tidaklah selalu sama.

Nilai yang terdapat pada cabang menyatakan pilihan tamu yang dilayani, diurutkan sebagai pasien cepat, pasien normal, pasein lambat. Misal, nilai 2,0,0, berarti melayani 2 pasien cepat, 0 pasein normal, dan 0 pasien lambat. Ongkos dari setiap simpul dinyatakan pada rumus (2.1) bab 2, pada halaman 21

Berikut dikemukakan contoh kasus keadaan pada permainan Jaram Perlaya, dan penyelesaian pencarian keuntungan terbesar dengan menggunakan algoritma B&B, Properti :

1. Meja : 2 buah
2. Tamu : 6 tamu, 2 tamu cepat, 2 tamu normal, 2 tamu lambat

Simpul akar adalah simpul di mana belum satupun tamu memperoleh meja. Berdasarkan tahap-tahap yang telah dijelaskan, akan terbentuk suatu pohon ruang status yang dijelaskan pada gambar 3.4.



Gambar 3.4 Pohon ruang status untuk kasus

Berdasarkan pohon ruang status yang dibentuk dengan menggunakan algoritma *Branch and Bound*, keuntungan maksimal yang dapat diperoleh adalah 700, dengan urutan tamu yang dilayani :

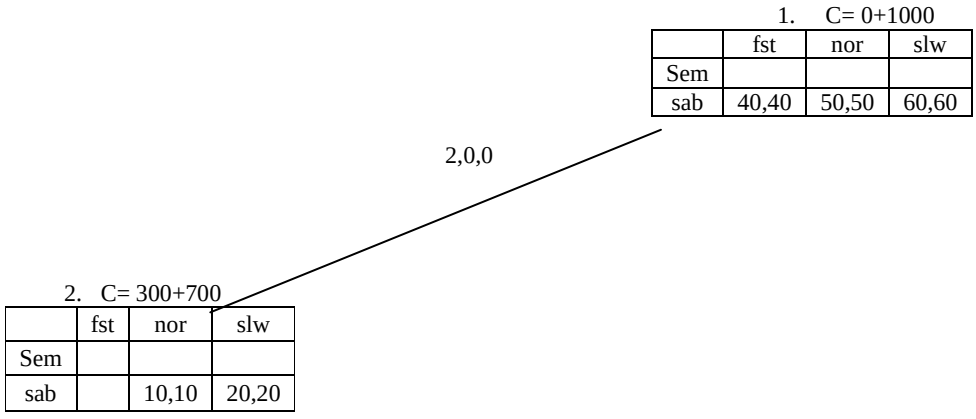
1. Dua tamu cepat
2. Dua tamu lambat

2. C= 300+700

	fst	nor	slw
Sem			
sab		10,10	20,20

Gambar 3.5 Simpul 2 Pohon Ruang Kasus

Pada simpul (2), dapat dilihat "c = 300 + 700". Nilai 300, merupakan f(2), menyatakan total poin yang telah diterima (dari dua tamu cepat yang telah menyelesaikan penyembuhan). Nilai 700, merupakan g(2), menyatakan total poin yang masih dapat diperoleh pemain, dengan menjumlahkan poin yang (akan) diberikan oleh tamu yang sedang mengantri (dua tamu sedang dan dua tamu lambat).



Gambar 3.6 Simpul 1 dan 2 Pohon Ruang Kasus

Antara simpul (1) dan (2), terdapat nilai 2,0,0. Seperti telah dijelaskan, nilai pertama menyatakan banyak tamu cepat yang dilayani, nilai kedua menyatakan banyak tamu normal yang dilayani, dan nilai ketiga menyatakan banyak tamu

lambat yang dilayani. Nilai 2,0,0 menyatakan program melayani 2 tamu cepat, 0 tamu normal, dan 2 tamu lambat. Simpul(2) merupakan kondisi di mana 2 tamu cepat telah selesai bersantap, dan pemain memperoleh poin sebesar 2x150.

Berdasarkan hasil perhitungan algoritma B&B maka solusi yang dihasilkan yaitu dengan mendahulukan 2 pasien cepat dan 2 pasien lambat sehingga permainan bisa mencapai *goal* poin dalam permainan tersebut.

3.1.9. Analisis Kebutuhan Non-Fungsional

Analisis 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

3.1.9.1. Analisis Kebutuhan Perangkat Lunak

Perangkat lunak digunakan dalam sebuah sistem merupakan perintah-perintah yang diberikan kepada perangkat keras agar bisa saling berinteraksi diantara keduanya. Perangkat lunak yang dibutuhkan untuk membangun *game* 2D Jaram Perlaya berbasis desktop ini adalah sebagai berikut :

1. Sistem operasi Windows 7.
2. NetBeans 7.3 digunakan untuk membuat game.
3. CorelDraw Graphics Suite 12 digunakan untuk pembuatan karakter.

3.1.9.2. Analisis Kebutuhan Perangkat Keras

Perangkat keras yang sesuai dengan kebutuhan aplikasi sangat diperlukan agar aplikasi dapat berjalan dengan baik. Spesifikasi minimum perangkat keras yang dibutuhkan untuk membangun *game* Jaram Perlaya ini adalah sebagai berikut:

1. Processor 1,80 Ghz
2. RAM 512 MB
3. VGA 128 MB
4. Harddisk 80 GB
5. Mouse, Keyboard dan Speaker

Perangkat keras minimum bagi pengguna untuk memainkan *game* ini adalah sebagai berikut :

1. Processor 1,60 Ghz
2. RAM 128 MB
3. VGA 64 MB
4. Harddisk 10 GB
5. Mouse, Keyboard dan Speaker

3.1.9.3. Analisis *User*

Pada analisis *user* (pemain) ini akan mencakup analisis beberapa parameter terhadap calon user dari applikasi.

3.1.9.3.1. *User Knowledge and Experience*

Game Jaram Perlaya ini bisa digunakan oleh kalangan apapun, tetapi pengetahuan dan pengalaman akan memudahkan *user* dalam penggunaanya. Terutama pengetahuan dan pengalaman dalam memainkan *game* tersebut. Tabel 3.4. berikut menjelaskan analisis klasifikasi *knowledge and experience* dari pengguna aplikasi.

Tabel 3.4. Analisis klasifikasi *knowledge and experience*

Educational *level*	**Reading *Level***	**Typing Skills**
Bisa digunakan oleh berbagai kalangan, seperti pelajar, mahasiswa hingga masyarakat awam	Bisa digunakan oleh berbagai *level* pendidikan dengan reading *level* yang sedang	*Average* 40 WPM (write per minute) Tidak memerlukan *typing* skils yang tinggi
Computer Literacy	**Task Experience**	**System Experience**
Moderate (menengah)	Bisa digunakan oleh dengan pengalaman penggunaan computer dan *game* yang sedang	Bisa digunakan oleh dengan pengalaman penggunaan komputer dan *game* yang sedang
Application Experience	**Native Language**	**Use Of Other System**
Bisa digunakan dalam semua sistem operasi Windows xp/vista/7	Menggunakan satu bahasa, yakni Indonesia	Bisa di jalankan tanpa perlu install applikasi lain

3.1.9.3.2. User Physical Characteristic

Keadaan fisik seseorang mungkin akan berpengaruh pada penggunaan aplikasi *game* ini. Ada hal-hal yang harus diperhatikan juga terhadap user dari karakteristik fisiknya untuk dapat menggunakan aplikasi ini yaitu, *Color Blind, Handednes*, dan *Gender* seperti terlihat pada tabel 3.5.

Tabel 3.5. Analisis *Users Physical Characteristic*

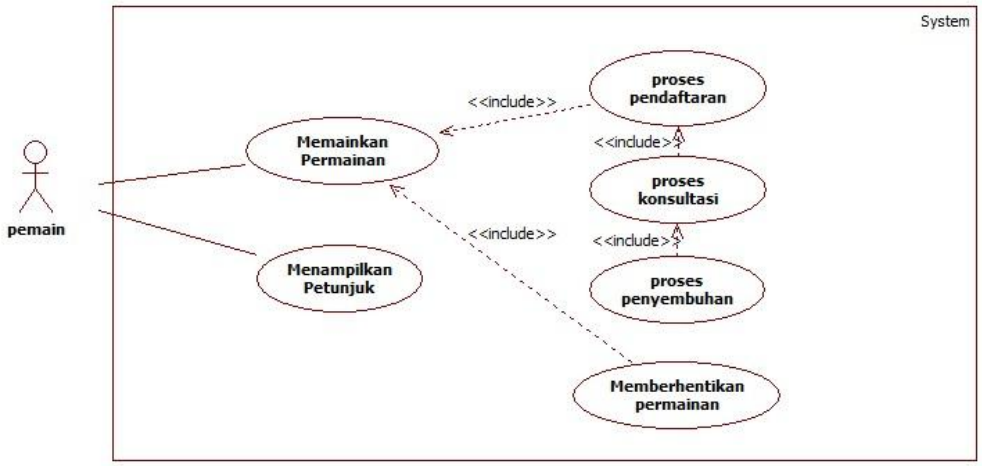
Age	13-17 tahun
Gender	Pria dan Wanita
Handedness	Kanan dan Kiri
Color Blind	User yang tidak bisa membedakan warna yang satu dengan yang lainnya (buta warna) tidak mampu menggunakan aplikasi ini, karena ada indicator warna-warna khusus yang membedakan antara fungsional yang satu dengan fungsional yang lainnya.

3.1.10. Analisis Kebutuhan Fungsional

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

3.1.10.1. *Use Case Diagaram*

Use case Diagram merupakan konstruksi untuk mendeskripsikan hubungan-hubungan yang terjadi antar aktor dengan aktivitas yang terdapat pada sistem. Sasaran pemodelan *use case* diantaranya adalah mendefinisikan kebutuhan fungsional dan operasional sistem dengan mendefinisikan skenario penggunaan sistem yang akan dibangun. Dari hasil analisis aplikasi yang ada maka *usecase diagram* untuk *game* 2D Jaram Perlaya dapat dilihat pada gambar 3.7



Gambar 3.7 *Use Case Diagram*

3.1.10.2. Definisi *Actor*

Definisi *Actor* berfungsi untuk menjelaskan *Actor* yang terdapat pada *Use Case Diagram*. Definisi *Actor* diterangkan pada Tabel 3.6

Tabel 3.6. Definisi *Actor*

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

3.1.10.3. Definisi *Use case*

Definisi *use case* berfungsi untuk menjelaskan proses yang terdapat pada setiap *use case*. Berikut definisi *use case* dapat dilihat pada tabel 3.7

Tabel 3.7 Tabel Definisi *Use Case*

No	***Use Case***	**Deskripsi**
1.	Memainkan Permainan	Proses ketika masuk permainan
2.	Menampilkan Petunjuk	Proses ketika pemain melihat petunjuk
3.	Proses Pendaftaramn	Proses untuk pemain menempatkan pasien ke tempat pendaftaran
4.	Proses Konsultasi	Proses untuk pemain menempatkan pasien ke tempat konsultasi
5.	Proses Penyembuhan	Proses untuk pemain menempatkan pasien ke tempat penyembuhan
6.	Memberhentikan Permainan	Proses untuk menghentikan sementara permainan

3.1.10.4. Skenario *Use Case*

Use Case Skenario berisi mengenai skenario (*flow of event*) untuk beberapa *use case* utama, yang menggambarkan urutan interaksi actor dengan *use case* tersebut dari awal sampai akhir. *Use case* skenario yang terdapat pada pembangunan *game* ini antara lain sebagai berikut:

Skenario *Use Case* Memainkan Permainan

Skenario *Use Case* Memainkan Permainan bertujuan untuk melihat *storyline* / cerita permainan dan masuk ke dalam permainan. Skenario *Use Case* diterangkan pada Tabel 3.8.

Tabel 3.8. Skenario *Use Case* Memainkan Permainan

Identifikasi		
Nomor	1	
Nama	Memainkan permainan	
Tujuan	Melihat *storyline* / cerita permainan dan masuk ke dalam permainan	
Deskripsi	Proses untuk melihat *storyline* / cerita permainan dan masuk ke dalam permainan	
Aktor	Pemain	
Skenario utama		
Kondisi Awal	Pemain berada di menu utama	
Aksi Aktor		**Aksi Sistem**
Memilih menu mainkan		
		Menampilkan *storyline* tentang awal mula perjalanan dr.Mimo
Memilih tombol *next*		
		Menampilkan permainan
Kondisi akhir	Sistem menampilkan halaman permainan	

Skenario *Use Case* Menampilkan Petunjuk

Skenario *Use Case* menampilkan petunjuk bertujuan untuk Menampilkan petunjuk dari permainan. Skenario *Use Case* diterangkan pada Tabel 3.9.

Tabel 3.9. Skenario *Use Case* Menampilkan Petunjuk

Identifikasi	
Nomor	2
Nama	Menampilkan Petunjuk
Tujuan	Menampilkan petunjuk dari permainan
Deskripsi	Proses ketika pemain memilih menu petunjuk
Aktor	Pemain
Skenario utama	
Kondisi Awal	Sistem menampilkan menu utama

Aksi Aktor	Aksi Sistem
Pemain memilih tombol petunjuk	
	Sistem menampilkan petunjuk
Kondisi akhir	Sistem menampilkan informasi petunjuk

Skenario *Use Case* Proses Pendaftaran

Skenario *Use Case* proses pendaftaran bertujuan untuk menempatkan pasien pada posisi yang ada ke tempat pasien untuk daftar. Skenario *Use Case* diterangkan pada Tabel 3.10.

Tabel 3.10. Skenario *Use Case* Proses Pendaftaran

Identifikasi	
Nomor	3
Nama	Proses Pendaftaran
Tujuan	Menempatkan pasien pada posisi yang ada ke tempat pasien untuk daftar.
Deskripsi	Proses untuk memindahkan pasien yang ada untuk proses penyembuhan dengan menggunakan *mouse*
Aktor	Pemain
Skenario utama	
Kondisi awal	Pemain menekan tombol kiri pada *mouse*
Aksi Aktor	**Aksi Sistem**
Pemain menekan tombol kiri pada *mouse* dan diarahkan pada pasien	
	Sistem menangkap posisi pasien
Pemain menempatkan pasien ke tempat pendaftaran	
	Sistem memindahkan posisi pasien ke tempat pendaftaran
Kondisi akhir	Sistem akan memindahkan pasien sesuai dengan penempatan yang dipilih oleh pemain

Skenario *Use Case* Proses Konsultasi

Skenario *Use Case* proses konsultasi bertujuan untuk menempatkan pasien dari posisi pendaftaran ke posisi konsultasi proses penyembuhan. Skenario *Use Case* diterangkan pada Tabel 3.11.

Tabel 3.11. Skenario *Use Case* Proses Konsultasi

Identifikasi		
Nomor	4	
Nama	Proses Konsultasi	
Tujuan	Menempatkan pasien dari posisi pendaftaran ke posisi konsultasi	
Deskripsi	Proses untuk memindahkan pasien yang ada untuk proses penyembuhan dengan menggunakan *mouse*	
Aktor	Pemain	
Skenario utama		
Kondisi awal	Pemain menekan tombol kiri pada *mouse*	
Aksi Aktor		**Aksi Sistem**
Pemain menekan tombol kiri pada *mouse* dan diarahkan pada pasien		
		Sistem menangkap posisi pasien
Pemain menempatkan pasien ke posisi konsultasi		
		Sistem memindahkan posisi pasien ke posisi konsultasi
Kondisi akhir	Sistem akan memindahkan pasien sesuai dengan penempatan yang dipilih oleh pemain	

Skenario *Use Case* Proses Penyembuhan

Skenario *Use Case* proses penyembuhan bertujuan untuk menempatkan pasien dari posisi konsultasi ke posisi penyembuhan. Skenario *Use Case* diterangkan pada Tabel 3.12.

Tabel 3.12. Skenario *Use Case* Proses Penyembuhan

Identifikasi	
Nomor	5
Nama	Proses Penyembuhan
Tujuan	Menempatkan pasien dari posisi konsultasi ke posisi

	penyembuhan	
Deskripsi	Proses untuk memindahkan pasien yang ada untuk proses penyembuhan dengan menggunakan *mouse*	
Aktor	Pemain	
Skenario utama		
Kondisi awal	Pemain menekan tombol kiri pada *mouse*	
Aksi Aktor		**Aksi Sistem**
Pemain menekan tombol kiri pada *mouse* dan diarahkan pada pasien		
		Sistem menangkap posisi pasien
Pemain menempatkan pasien ke posisi penyembuhan		
		Sistem memindahkan posisi pasien ke posisi penyembuhan
Kondisi akhir	Sistem akan memindahkan pasien sesuai dengan penempatan yang dipilih oleh pemain	

Skenario *Use Case* Memberhentikan Permainan

Skenario *Use Case* Memberhentikan Permainan bertujuan untuk melihat petunjuk permainan, melanjutkan permainan, dan keluar dari permainan. Skenario *Use Case* diterangkan pada Tabel 3.13.

Tabel 3.13 Skenario *Use Case* Memberhentikan Permainan

Identifikasi		
Nomor	6	
Nama	Memberhentikan Permainan	
Tujuan	Melihat petunjuk permainan, melanjutkan permainan dan keluar dari permainan	
Deskripsi	Proses ketika pemain ingin melihat petunjuk permainan, melanjutkan permainan dan keluar dari permainan	
Aktor	Pemain	
Skenario Utama		
Kondisi awal	Sistem menampilkan permainan	
Aksi Aktor		Reaksi Sistem
Pemain menekan tombol escape		

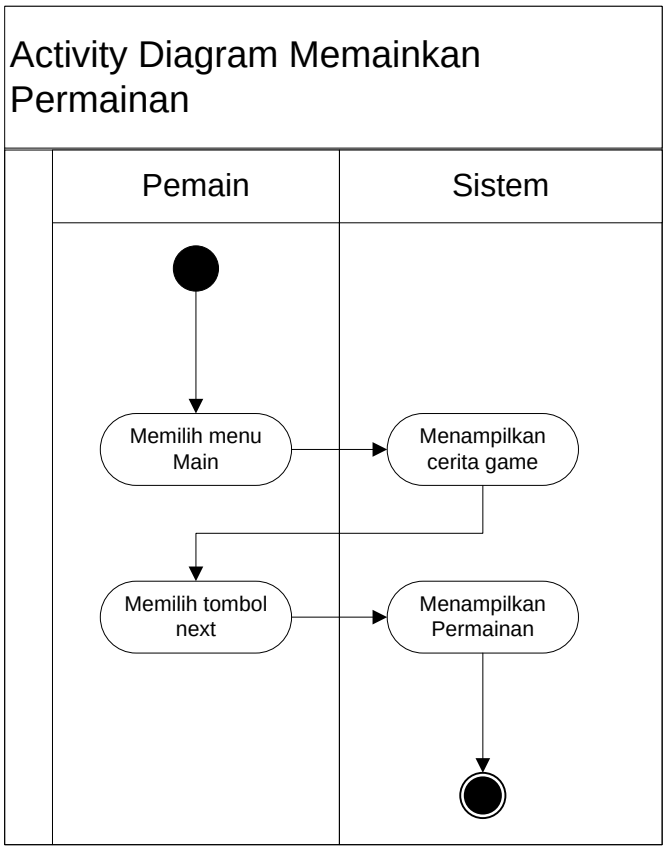
	Sistem menampilkan menu jeda
Pemain memilih tombol kembali	
	Sistem menampilkan kembali ke permainan
Pemain memilih tombol petunjuk	
	Sistem menampilkan petunjuk
Pemain memilih tombol keluar	
	Sistem keluar dari permainan
Kondisi Akhir	Sistem menampilkan menu jeda

3.1.10.5. Activity Diagram

Activity diagram menggambarkan berbagai alir aktivitas dalam sistem yang sedang dirancang, bagaimana masing-masing alir berawal, *decision* yang mungkin terjadi, dan bagaimana mereka berakhir. *Activity diagram* juga dapat menggambarkan proses paralel yang mungkin terjadi pada beberapa eksekusi. *Activity diagram* merupakan state diagram khusus, dimana sebagian besar *state* adalah *action* dan sebagian besar transisi di-*trigger* oleh selesainya *state* sebelumnya (internal processing).

ActivityDiagram Memainkan Permainan

Aktivity diagram memainkan permainan adalah aktivitas pemain memainkan *game* yang dapat dilihat pada Gambar 3.8



Gambar 3.8 *Activity Diagram* Memainkan Permainan

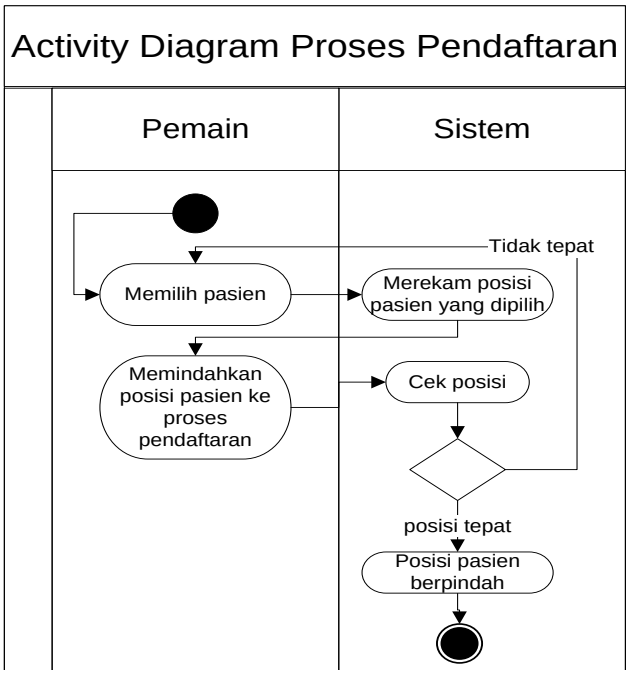
Penjelasan proses yang terjadi pada *activity diagram* memainkan permainan dijelaskan pada Tabel 3.14.

Tabel 3.14 Penjelasan Proses Memainkan Permainan

Deskripsi	Permainan merupakan kondisi pemain ketika telah berada di dalam permainan.
Normal Flow	1. Pemain memulai permainan 2. Sistem menampilkan *storyline* 3. Pemain menekan tombol next 4. Sistem menampilkan permainan

***Activity Diagram* Proses Pendaftaran**

Activity diagram proses pendaftaran adalah aktivitas dimana pemain memilih pasien untuk memindahkan posisi pasien. *Aktivity diagram* proses pendaftaran dapat dilihat pada Gambar 3.9.



Gambar 3.9 *Activity Diagram* Proses Pendaftaran

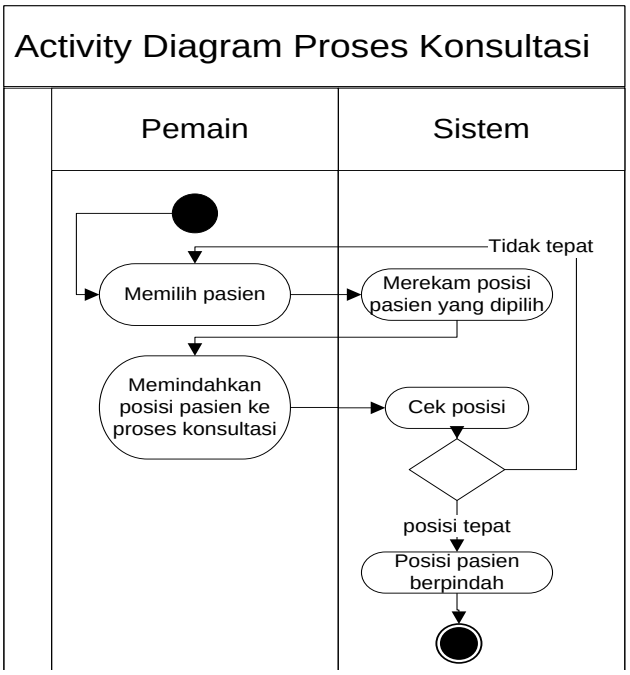
Penjelasan proses yang terjadi pada *activity diagram* proses pendaftaran dijelaskan pada Tabel 3.15

Tabel 3.15 Penjelasan Proses pendaftaran

Deskripsi	Menempatkan pasien merupakan kondisi ketika pemain memindahkan posisi pemain
Normal Flow	1. Pemain memilih pasien 2. Sistem merekam posisi pasien yang dipilih 3. Pemain memindahkan posisi pasien ke langkah 1 yaitu tempat pendaftaran 4. Jika posisi sesuai maka pasien akan berpindah 5. Jika posisi tidak sesuai maka pasien kembali lagi ke posisi sebelum dipindahkan 6. Sistem akan memindahkan pasien sesuai dengan posisi yang dipilih pemain

Activity Diagram Proses Konsultasi

Activity diagram proses konsultasi adalah aktivitas dimana pemain memilih pasien untuk memindahkan posisi pasien. *Activity diagram* proses konsultasi dapat dilihat pada Gambar 3.10.



Gambar 3.10 *Activity Diagram* Proses Konsultasi

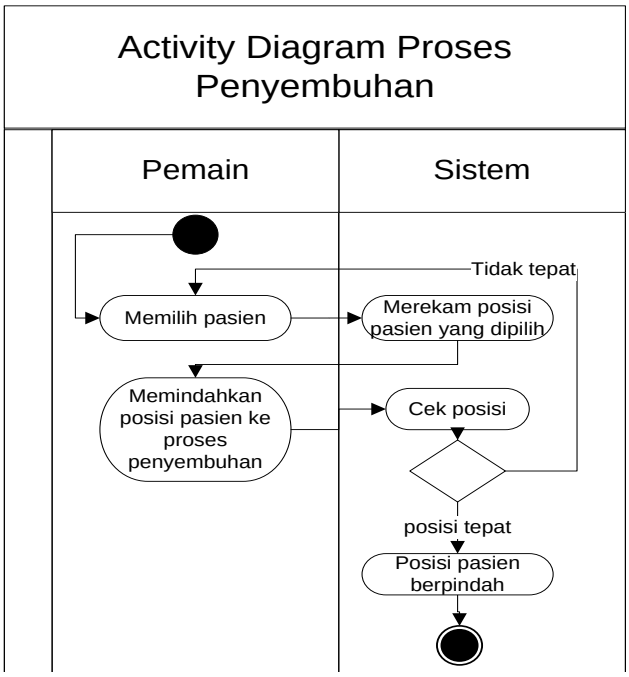
Penjelasan proses yang terjadi pada *activity diagram* proses konsultasi dijelaskan pada Tabel 3.16

Tabel 3.16 Penjelasan Proses Konsultasi

Deskripsi	Menempatkan pasien merupakan kondisi ketika pemain memindahkan posisi pemain
Normal Flow	1. Pemain memilih pasien 2. Sistem merekam posisi pasien yang dipilih 3. Pemain memindahkan posisi pasien dari langkah 1 ke langkah 2 yaitu tempat konsultasi 4. Jika posisi sesuai maka pasien akan berpindah 5. Jika posisi tidak sesuai maka pasien kembali lagi ke posisi sebelum dipindahkan 6. Sistem akan memindahkan pasien sesuai dengan posisi yang dipilih pemain

Activity Diagram Proses Penyembuhan

Activity diagram proses penyembuhan adalah aktivitas dimana pemain memilih pasien untuk memindahkan posisi pasien. *Activity diagram* proses penyembuhan dapat dilihat pada Gambar 3.11.



Gambar 3.11 *Activity Diagram* Proses Penyembuhan

Penjelasan proses yang terjadi pada *activity diagram* proses penyembuhan dijelaskan pada Tabel 3.17

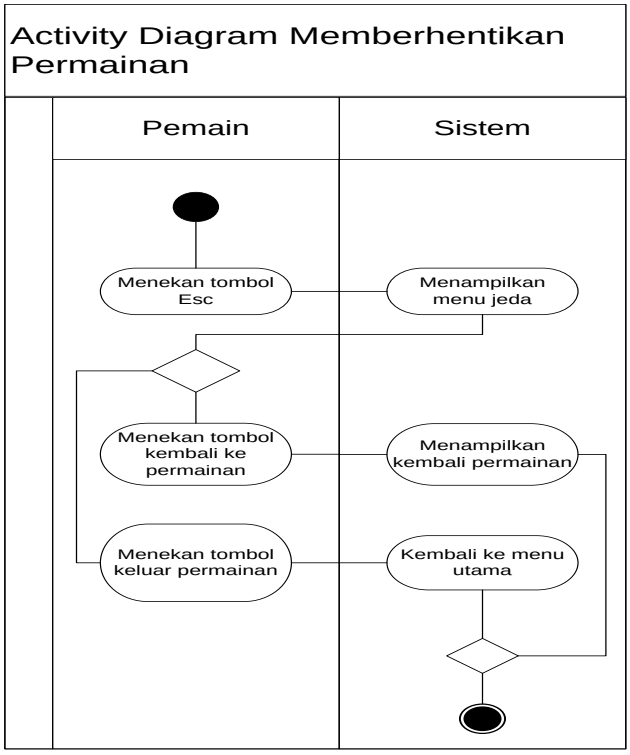
Tabel 3.17 Penjelasan Proses Penyembuhan

Deskripsi	Menempatkan pasien merupakan kondisi ketika pemain memindahkan posisi pemain
Normal Flow	1. Pemain memilih pasien 2. Sistem merekam posisi pasien yang dipilih 3. Pemain memindahkan posisi pasien dari langkah 2 ke langkah 3 yaitu tempat penyembuhan 4. Jika posisi sesuai maka pasien akan berpindah 5. Jika posisi tidak sesuai maka pasien kembali lagi ke posisi sebelum dipindahkan 6. Sistem akan memindahkan pasien sesuai dengan posisi yang dipilih pemain

Activity Diagram Memberhentikan Permainan

Aktivity diagram memberhentikan permainan adalah aktivitas untuk melihat petunjuk permainan, melanjutkan permainan, dan keluar

dari permainan. *Aktivitas diagram* memberhentikan permainan dapat dilihat pada Gambar 3.12.



Gambar 3.11 *Activity Diagram* Memberhentikan Permainan

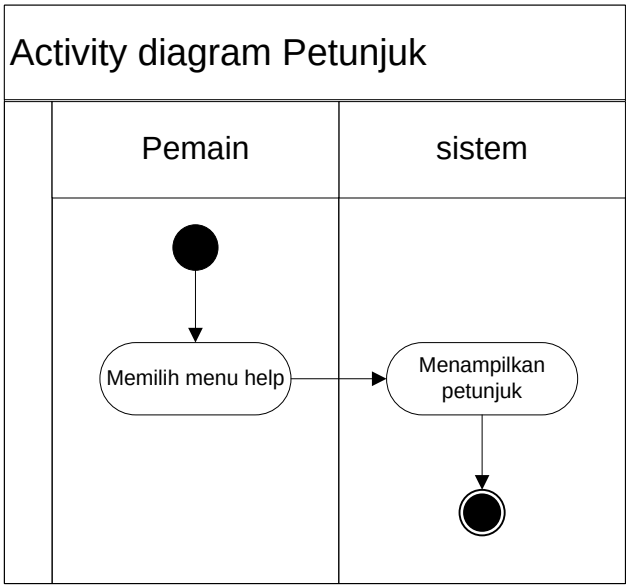
Penjelasan proses yang terjadi pada *activity diagram* memberhentikan permainan dijelaskan pada Tabel 3.18

Tabel 3.18 Penjelasan Proses Memberhentikan Permainan

Deskripsi	Memberhentikan permainan merupakan kondisi ketika pemain sedang memainkan permainan dan pemain akan memberhentikan permainan
Normal Flow	1. Pemain menekan tombol Esc pada keyboard 2. Sistem menampilkan menu jeda 3. Pemain menekan tombol kembali ke permainan 4. Sistem akan melanjutkan permainan yang terhenti. 5. Pemain menekan tombol keluar permainan 6. Sistem akan kembali ke menu utama

***Activity Diagram* Petunjuk**

Activity diagram petunjuk adalah aktivitas untuk melihat petunjuk permainan. *Aktivitas diagram* memberhentikan permainan dapat dilihat pada Gambar 3.13.



Gambar 3.12 Activity Diagram Petunjuk

Penjelasan proses yang terjadi pada *activity diagram* petunjuk dijelaskan pada Tabel 3.19

Tabel 3.19 Penjelasan Proses Petunjuk

Deskripsi	Petunjuk merupakan kondisi ketika pemain sedang memainkan permainan dan pemain akan memberhentikan permainan
Normal Flow	1. Pemain memilih menu *help* 2. Sistem menampilkan menu petunjuk

3.1.10.6.*Sequence* Diagram

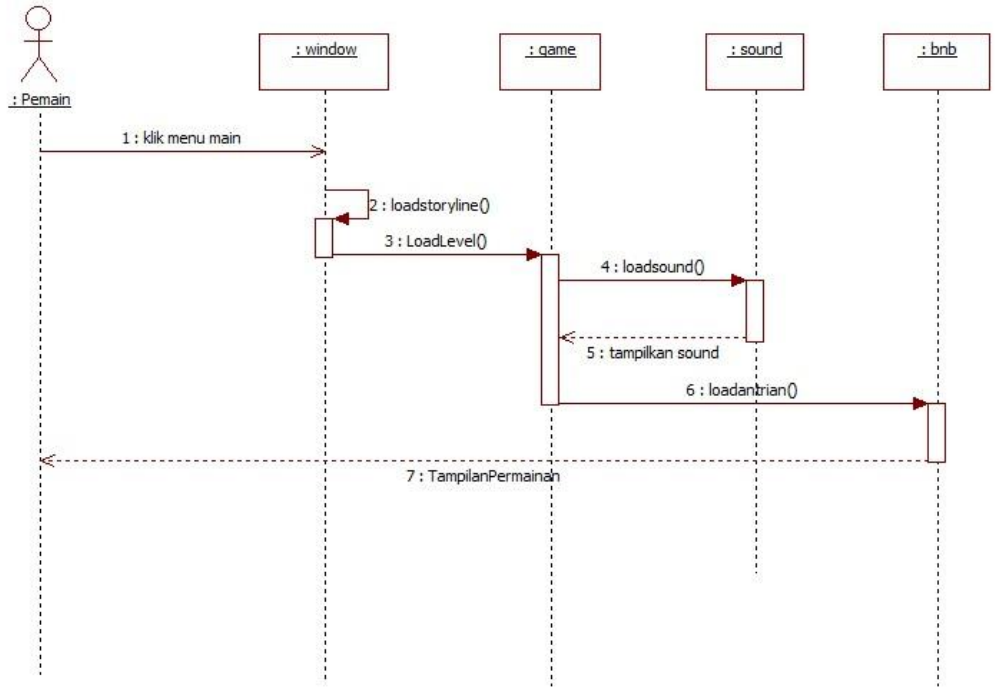
Sequence diagram menggambarkan interaksi antar objek di dalam dan di sekitar sistem (termasuk pengguna, display, dan sebagainya) berupa *message* yang digambarkan terhadap waktu. *Sequence diagram* terdiri atas dimensi vertikal (waktu) dan dimensi horizontal (objek-objek yang terkait).

Sequence diagram biasa digunakan untuk menggambarakan skenario atau rangkaian langkah-langkah yang dilakukan sebagai respon dari sebuah event untuk menghasilkan *output* tertentu. Diawali dari apa yang men-*trigger* aktivitas tersebut, proses dan perubahan apa saja yang terjadi secara internal dan *output* apa yang dihasilkan.

Berikut ini beberapa *sequencial diagram* yang terdapat pada *game* Jaram Perlaya yaitu :

***Sequence Diagram* Memainkan Permainan**

Sequence diagram untuk *Use Case* memulai permainan menggambarkan interaksi antara objek dari *class* window dan objek yang berkaitan dengan *class* lainya. Keseluruhan *sequence* memainkan permainan dapat dilihat pada Gambar 3.14

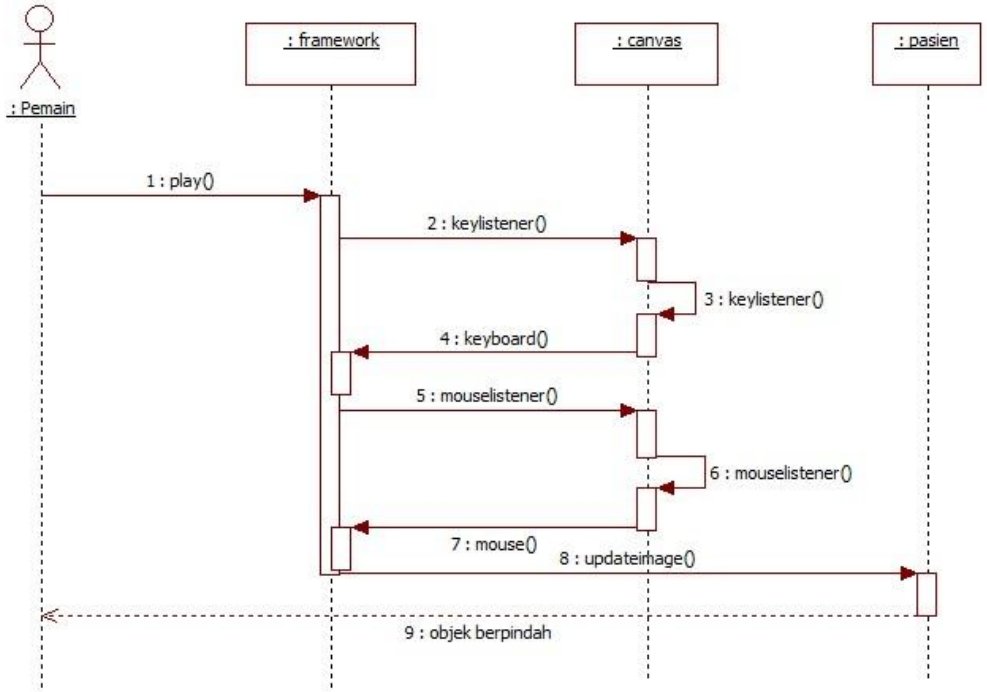


Gambar 3.14 *Sequence Diagram* Memainkan Permainan

***Sequence Diagram* Proses Pendaftaran**

Sequence diagram untuk *Use Case* proses pendaftaran menggambarkan interaksi antara objek dari *class game* dan objek yang

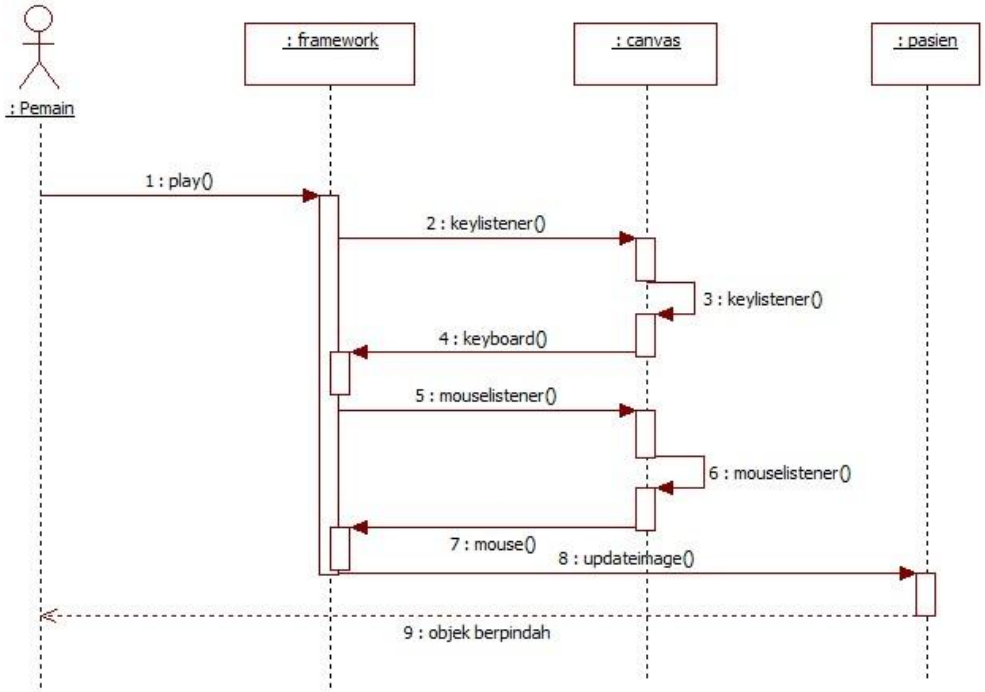
berkaitan dengan *class* lainya. Keseluruhan *sequence* proses pendaftaran dapat dilihat pada Gambar 3.15



Gambar 3.15 *Sequence Diagram* Proses Pendaftaran

***Sequence Diagram* Proses Konsultasi**

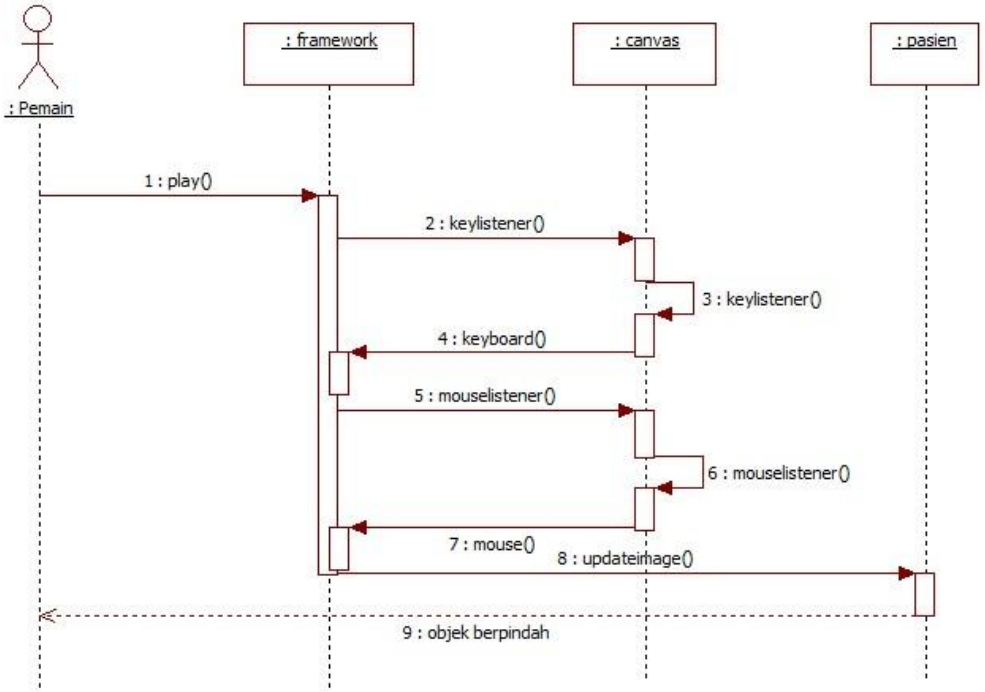
Sequence diagram untuk *Use Case* proses konsultasi menggambarkan interaksi antara objek dari *class game* dan objek yang berkaitan dengan *class* lainya. Keseluruhan *sequence* proses konsultasi dapat dilihat pada Gambar 3.16



Gambar 3.16 *Sequence Diagram* Proses Konsultasi

***Sequence Diagram* Proses Penyembuhan**

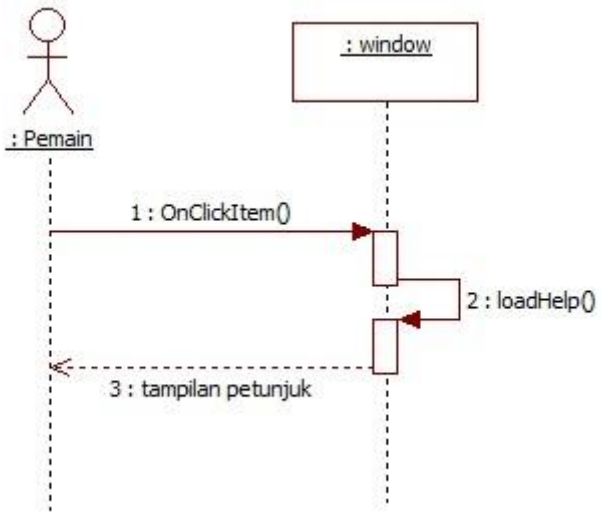
Sequence diagram untuk *Use Case* proses penyembuhan menggambarkan interaksi antara objek dari *class game* dan objek yang berkaitan dengan *class* lainya. Keseluruhan *sequence* proses konsultasi dapat dilihat pada Gambar 3.17.



Gambar 3.17. *Sequence Diagram* Proses Penyembuhan

***Sequence Diagram* Petunjuk**

Sequence diagram untuk *Use Case* petunjuk menggambarkan interaksi antara objek dari *class window* dan objek yang berkaitan dengan *class* lainya. Keseluruhan *sequence* petunjuk dapat dilihat pada Gambar 3.18



Gambar 3.18 *Sequence Diagram* Petunjuk

3.1.10.7.Class Diagram

Class diagram menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang dibuat untuk membangun sistem. *Class diagram* pada *game* ini dapat dilihat pada gambar 3.19



Gambar 3.19 *Class Diagram*

3.2. Perancangan Sistem

Perancangan adalah suatu bagian dari metodologi pengembangan suatu perangkat lunak yang dilakukan setelah tahapan untuk memberikan gambaran secara terperinci. Dalam tahap ini digambarkan rancangan sistem yang akan dibangun sebelum dilakukan pengkodean ke dalam suatu bahasa pemrograman.

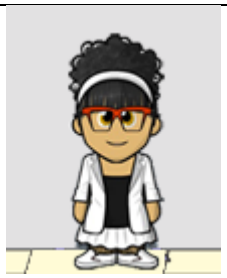
3.2.1. Perancangan Komponen Permainan

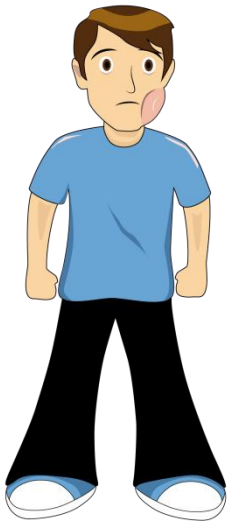
Perancangan komponen permainan meliputi karakter dan *storyboard* yang didalamnya menjelaskan karakter dan storyboard sebagai gambar perancangan pembuatan *game* Jaram Perlaya.

3.2.1.1. Karakter

Karakter utama dalam *game* ini bernama dr.Mimo seorang dokter yang bertugas menyembuhkan para pasien yang terkena penyakit. Sedangkan karakter pasien yang ada pada game ini terdapat empat karakter sesuai dengan jenis penyakit dari setiap karakter, berikut karakter-karakter dari pasien yang ada pada *game* Jaram Perlaya dijelaskan pada tabel 3.20.

Tabel 3.20 Pengenalan Karakter permainan

No.	Gambar	Keterangan
1.		Karakter utama dr.Mimo
2		Karakter perawat

3		Karakter pasien luka gores
4		Karakter pasien luka sayat
5		Karakter pasien luka memar

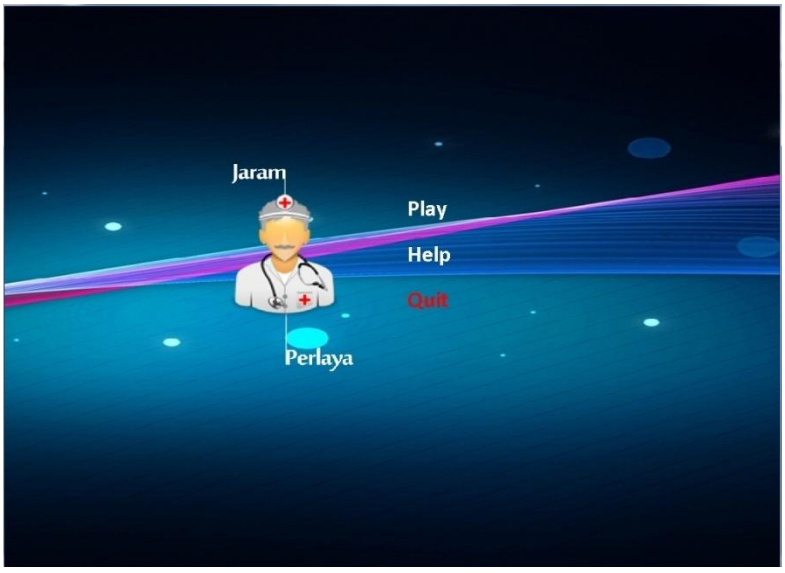
6		Karakter pasien luka bakar

3.2.1.2. *Story Board*

Storyboard adalah sketsa gambar yang disusun berurutan sesuai dengan naskah, dengan menggunakan *storyboard* dapat menyampaikan ide cerita kepada orang lain dengan lebih mudah karena dapat dengan *storyboard* dapat mengimajinasikan khayalan yang sesuai dengan gambar-gambar yang ada sehingga dapat menghasilkan persepsi yang sama pada ide cerita yang dibuat. Berikut *storyboard* dari *game* Jaram Perlaya :

1. Storyboard Menu Utama

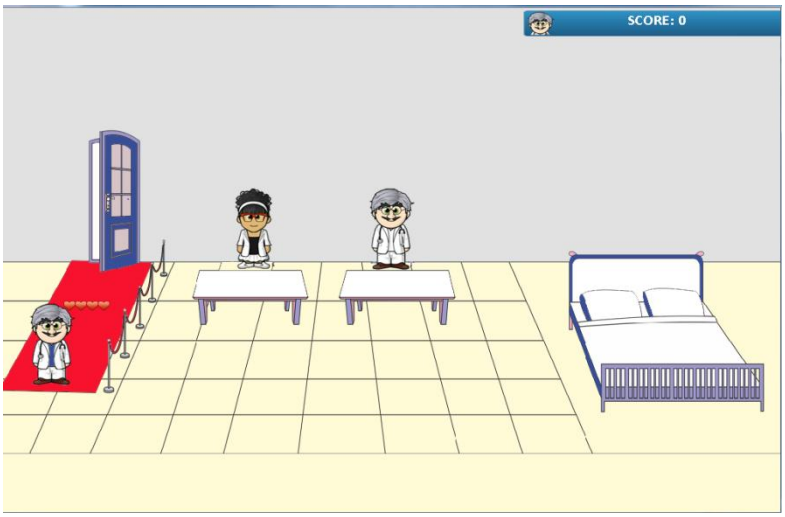
   Berikut adalah *storyboard* menu utama dari *game* Jaram Perlaya :



Gambar 3.20 *Storyboard* Menu Utama

2. Storyboard *Game*

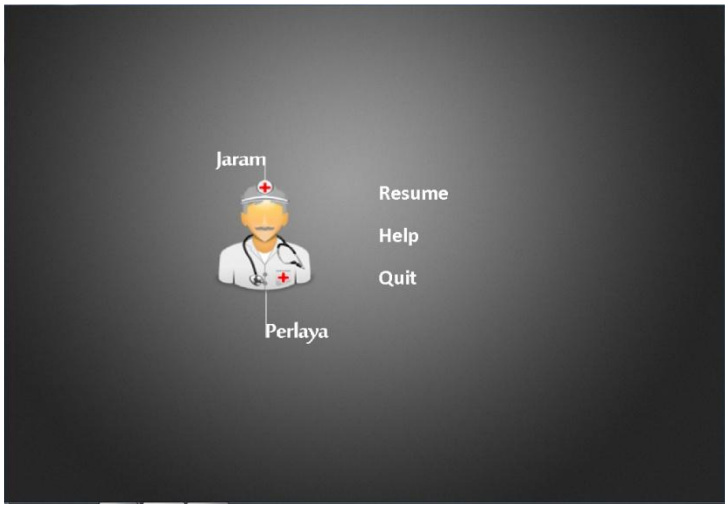
   Berikut adalah *storyboard game* pada permainan Jaram Perlaya:



Gambar 3.21 *Storyboard game*

3. Storyboar Memberhentikan Permainan

   Berikut adalah *storyboard* memberhentikan permainan dari *game* Jaram Perlaya



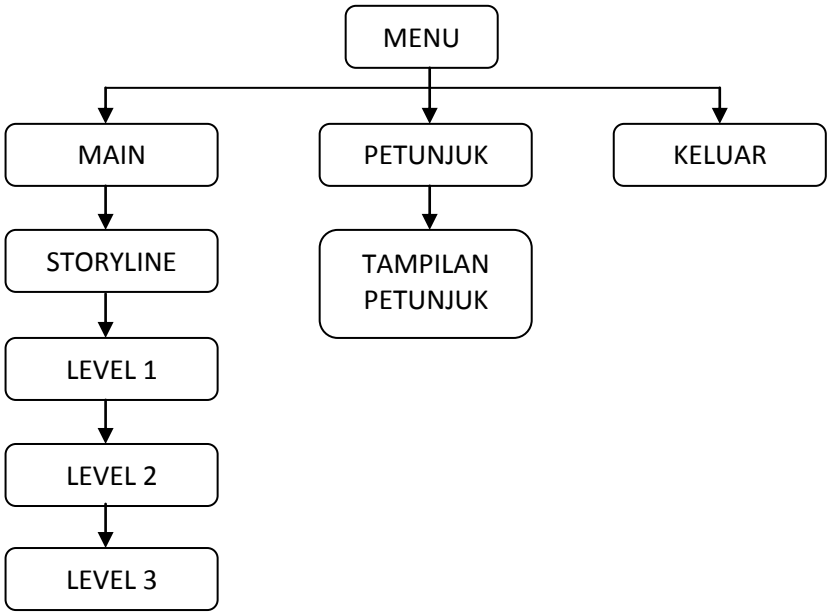
Gambar 3.22 *Storyboard* Memberhentikan Permainan

3.2.2. Perancangan Arsitektur

Setelah melakukan perancangan data pada sistem yang dibangun, maka dilakukanlah perancangan arsitektur. Perancangan arsitektur yang telah dibuat meliputi beberapa perancangan diantaranya struktur menu, perancangan antarmuka.

3.2.2.1. Perancangan Struktur Menu

Perancangan menu dibuat dengan harapan agar pengguna dapat mengunakan *Game* dengan mudah tanpa kesulitan, sehingga memudahkan pemakai dalam memilih menu aplikasi yang sedang berjalan. Struktur menu dapat dilihat pada gambar 3.23.



Gambar 3.23 Perancangan Struktur Menu Utama

3.2.3. Perancangan Antarmuka

Perancangan antarmuka atau *interface* merupakan tampilan dari suatu perangkat lunak yang berperan sebagai media komunikasi yang digunakan sebagai sarana berdialog antara perangkat lunak dengan pengguna. Sistem yang akan dibangun diharapkan menyediakan antarmuka yang mudah dipahami dan digunakan oleh pengguna.

Perancangan antarmuka *game* Jaram Perlaya diantaranya adalah sebagai berikut :

A. Antarmuka Menu Utama

Perancangan antarmuka menu utama dari *game* Jaram Perlaya dapat dilihat pada gambar 3.24.

T01	Keterangan
Background PLAY HELP QUIT	• Klik play untuk masuk ke T02 • Klik Help untuk masuk ke T03 • Klik Quit untuk keluar dari T01

Gambar 3.24 Perancangan Antar muka Menu Utama

B. Perancangan Antar muka Permainan

Perancangan antarmuka permainan dari *game* Jaram Perlaya dapat dilihat pada gambar 3.25.

T02	Keterangan
TIMER *SCORE*	• Tekan tombol Esc pada keyboard untuk masuk ke T04

Gambar 3.25 Perancangan Antar muka Permainan

C. Perancangan Antar muka Petunjuk

Perancangan antarmuka petunjuk dari *game* Jaram Perlaya dapat dilihat pada gambar 3.26.

T03	Keterangan
Tampilan Petunjuk KELUAR	• Tekan tombol Keluar untuk kembali ke T01

Gambar 3.26 Perancangan Antar muka Petunjuk

D. Perancangan Antar muka Jeda

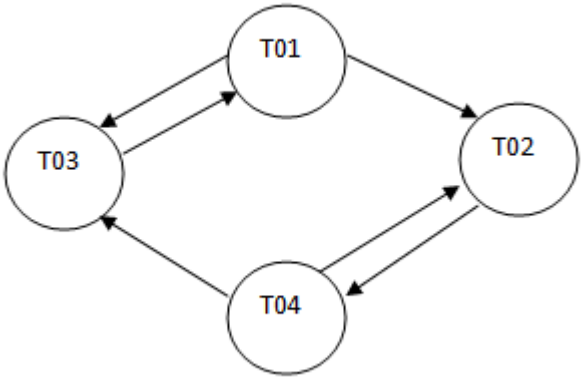
Perancangan antarmuka jeda dari *game* Jaram Perlaya dapat dilihat pada gambar 3.27.

T04	Keterangan
Tampilan Jeda LANJUTKAN PETUNJUK KELUAR	• Tekan tombol lanjutkan untuk kembali ke T02 • Tekan tombol petunjuk untuk ke T03 • Tekan tombol keluar untuk kembali ke T01

Gambar 3.27 Perancangan Antar muka Jeda

3.2.4. Jaringan Semantik

Setelah melakukan perancangan antarmuka, maka dilakukanlah perancangan terhadap aliran dari menu-menu yang ada pada program. Untuk menggambarkan hubungannya maka digambarkan dalam sebuah jaringan semantik pada gambar 3.28.



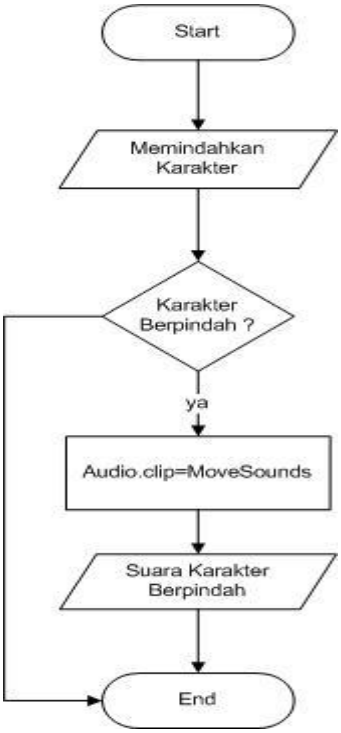
Gambar 3.28 Perancangan Jaringan Semantik

3.2.5. Perancangan Method

Perancangan *method* merupakan perancangan 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* Jaram Perlaya yang akan dibangun dijelaskan melalui *flowchart* sebagai berikut.

3.2.5.1. Perancangan *Method Play Move Sounds*

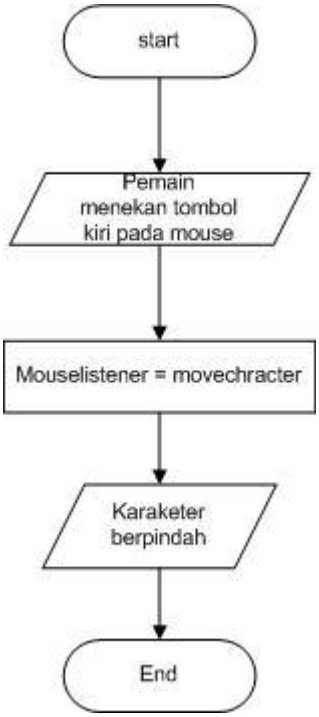
Method ini akan memberikan efek suara berpindah tempat terhadap karakter ketika pemain menggerakan karakter yang dapat dilihat pada gambar 3.29



Gambar 3.29 Flowchart *Method Play Move Sounds*

3.2.5.2. Perancangan Method Perpindahan Karakter

Method ini akan melakukan update berupa gerakan berpindah terhadap karakter yang diawali dari inputan yang dilakukan pemain yang dapat dilihat pada gambar 3.30.



Gambar 3.30. Perancangan Method Perpindahan Karakter