BAB 3

ANALISIS DAN PERANCANGAN SISTEM

3.1 Analisis Sistem

Analisis sistem merupakan kegiatan penguraian suatu sistem informasi yang utuh dan nyata ke dalam bagian-bagian atau komponen-komponen komputer yang bertujuan untuk mengidentifikasi serta mengevaluasi masalah-masalah yang muncul, hambatan-hambatan yang mungkin terjadi dan kebutuhan-kebutuhan yang diharapkan sehingga mengarah kepada solusi dengan kebutuhan serta perkembangan teknologi.

3.2 Analisis Masalah

Game dengan *genre construction and management simulation Game* sudah tidak asing lagi dalam permainan *Game*. Dalam permainannya, *Game* ini biasanya mempunyai ciri *Gameplay* seperti mendirikan, mengelola, dan memelihara. Selain itu, beberapa *Game* dari *genre* ini dapat mengambil dari kejadian nyata yang diantaranya seperti *Game sky park* dan *Contruction Tycoon*.

Dalam pembahasannya terdapat beberapa faktor pada *Game* simulasi ini menimbulkan beberapa masalah, diantaranya setelah membangun rumah atau bangunan secara otomatis mendapatkan poin dalam 10 detik secara otomatis, dalam *Game*-nya hanya bertugas untuk membangun dan mengelola lahan kosong saja, dalam tiap akhir permainannya hanya dipacu oleh waktu saja. Berdasarkan masalah yang menjadi acuan untuk membangun *Game Building of House* yaitu:

1. bagaimana membangun *Game* pembangunan rumah yang dapat memberikan keuntungan atau poin tidak secara otomatis .
2. bagaimana membangun *Game* yang mempunyai skenario *Game*-nya tidak hanya membangun rumah.
3. bagaimana membangun *Game* yang mempunyai *Gameplay* tidak hanya waktu yg dijadikan acuan akhir permainannya.

3.3 Analisis *Game* Sejenis

Analisis yang dilakukan merupakan observasi dari *Game* sejenis untuk melihat cara bermain, tujuan dan ada tidaknya kekurangan atau kelebihan *Game* itu sendiri. Hasil observasi tersebut untuk membantu dalam penelitian. *Game* sejenis yang dianalisis yaitu *sky park* dan *construction tycoon.*

3.3.1 *Sky Park*

Sky Park adalah sebuah *Game* konstruksi simulasi online. Sistem menyediakan tempat untuk membangun rumah pada awal permainan. *Game* ini tidak memiliki karakter yang dapat dijalankan pemain, karena pada dasarnya *Game* ini hanya menampilkan sebuah tempat untuk membangun sebuah bangunan yang dapat digunakan pemain untuk mendapatkan poin agar dapat membangun bangunan yang lain.

3.3.1.1 Storyline Sky Park

Storyline merupakan sebuah alur dari cerita *Game*, dimana alur tersebut dapat menggambarkan aktivitas yang berada dalam *Game*. Penjelasan *storyline* dari *Game sky park* akan dijelaskan dalam table berikut :

Tabel 3. 1 *Storyline Sky Park*

Storyline
Dalam permainan *skypark* menjelaskan tentang pembangunan taman diatas langit. Disini pemain harus melakukan pembersihan lahan dahulu karena di beberapa lahan tersebut masih ditempati oleh batuan atau pohon. Setelah pemain membersihkan lahan tersebut, pemain dapat membangun bangunan-bangunan taman. Bangunan tersebut memiliki variasi harga sesuai dengan bentuk bangunannya. Ketika pemain sudah membangun bangunan maka pemain akan mendapatkan point yang akan berjatuhan dari bangunan tersebut. Selain itu pemain diberi kesempatan waktu untuk membangun, jika kesempatan waktu habis maka permainan berakhir. Dalam *Game* ini tidak memiliki level selanjutnya atau hanya memiliki satu level.

Gameplay merupakan sebuah aturan bermain dari *Game*, dimana aturan tersebut dapat menjelaskan cara bermain dalam *Game*. Penjelasan dari *Gameplay* akan dijelaskan dalam table berikut :

Tabel 3. 2 *Gameplay Sky Park*

Gameplay
1. Pada awal permainan, pemain akan mempunyai saldo untuk membangun bangunan. 2. Saldo digunakan untuk membeli jasa pembuatan bangunan. 3. Permainan memiliki batas waktu untuk menyelesaikan permainan. 4. Pemain harus menghancurkan batuan atau pohon untuk membangun bangunan. 5. Saat bangunan telah dibangun, pemain dapat mengambil point berbentuk bintang yang keluar dari bangun tersebut..

Dalam penjelasan dari alur *Game sky park* diatas, ada beberapa hal yang dapat dihubungkan dengan *Game* yang akan dibangun.

Beberapa hal yang berhubungan dengan *Game* yang akan dibangun :

Tabel 3. 3 Hubungan dengan *Game Sky Park*

***Game* Sejenis**	***Game* yang akan dibangun**
Tidak menggambarkan keadaan nyata	Menggambarkan keadaan nyata
Membangun memakai biaya	Membangun memakai biaya
Membangun bangunan taman	Membangun bangunan rumah
Mendapatkan poin setiap detik	Mendapatkan poin dari pelanggan
Simulasi pembangunan taman	Simulasi pembangunan rumah



Gambar 3. 1 Awal Permainan *Game Sky Park*



Gambar 3. 2 Arena *Game Sky Park*

3.3.2 *Construction Tycoon*

Construction Tycoon adalah *Game* simulasi konstruksi *online*. Sistem menyediakan arena untuk membangun rumah pada awal permainan. *Game* ini tidak memiliki karakter utama, tetapi setiap pemain diberi poin untuk membangun sebuah rumah, dari rumah yang telah dibangun pemain akan mendapatkan poin, besarnya poin tergantung dari rumah yang telah dibangun. *Game* ini tidak memiliki level tertentu. Tujuan *Game* ini adalah mendapatkan target poin dari bangunan yang sudah dibangun.

3.3.2.1 Storyline Construction Tycoon

Storyline merupakan sebuah alur dari cerita *Game*, dimana alur tersebut dapat menggambarkan aktivitas yang berada dalam *Game*. Penjelasan *storyline* dari *Game Construction Tycoon* akan dijelaskan dalam table berikut :

Tabel 3. 4 *Storyline Construction Tycoon*

Storyline
Dalam permainan *Construction Tycoon* menjelaskan tentang seorang kontraktor yang gemar membangun rumah dan sejenisnya. Disini akan berperan sebagai kontraktor, pemain diberikan poin atau uang untuk membangun rumah dilahan yang sudah di sediakan. selain itu, disediakan beberapa bangunan rumah dengan bentuk yang berbeda dan harga yang berbeda. Setelah pemain membangun rumah di lahan yang kosong tersebut pemain dapat menaikan profit bangunan tersebut hingga 5 bintang. Ketika pemain membangun secara bersamaan musuh dengan acak ikut membangun sebagai musuh. Dalam permainan ini pemain harus mengalahkan profit yang di dapatkan oleh musuh. Pemain tidak dibatasi oleh waktu akan tetapi pemain mempunyai batasan profit yang harus dicapai dengan cara berlomba dengan musuh. Dalam *Game* ini tidak memiliki level selanjutnya atau hanya memiliki satu level.

Gameplay merupakan sebuah aturan bermain dari *Game*, dimana aturan tersebut dapat menjelaskan cara bermain dalam *Game*. Penjelasan dari *Gameplay* akan dijelaskan dalam table berikut :

Tabel 3. 5 *Gameplay Construction Tycoon*

Gameplay
1. Pada awal permainan, pemain akan mempunyai saldo untuk membangun bangunan. 2. Saldo digunakan untuk membeli jasa pembuatan bangunan. 3. Permainan memiliki target saldo untuk menyelesaikan permainan. 4. Pemain harus bersaing membangun bangunan dengan cepat dilahan yang kosong dengan musuh secara bersamaan. 5. Saat bangunan telah dibangun, pemain dapat mengambil point berbentuk bintang yang keluar dari bangun tersebut. 6. Pemain tidak dapat membangun dilahan yang telah di bangun oleh musuh 7. Jika musuh dapat menempuh target saldo sebelum pemain, pemain akan dinyatakan kalah.

Dalam penjelasan dari alur *Game sky park* diatas, ada beberapa hal yang dapat dihubungkan dengan *Game* yang akan dibangun

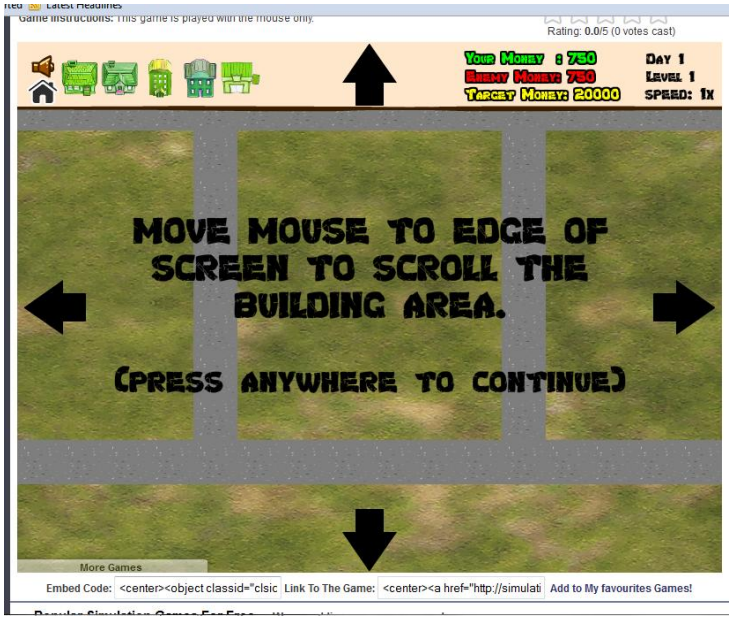
Beberapa hal yang berhubungan dengan *Game* yang akan dibangun :

Tabel 3. 6 Hubungan dengan *Game construction tycoon*

***Game* Sejenis**	***Game* yang akan dibangun**
Menggambarkan keadaan nyata	Menggambarkan keadaan nyata
Pembangunan memakai biaya	Pembangunan memakai biaya
Memiliki waktu akhir permainan	Memiliki waktu akhir permainan
Mendapatkan poin setiap detik	Mendapatkan poin dari pelanggan
Tidak ada akhir permainan	Memiliki akhir permainan



Gambar 3. 3 Menu *Game Construction Tycoon*



Gambar 3. 4 Arena *Game Construction Tycoon*



Gambar 3. 5 Permainan *Game Construction Tycoon*

3.4 Analisis *Game* yang akan dibangun

Analisis *Game* yang akan dibangun merupakan bagian yang mendeskripsikan *Game* yang akan dibangun. Pada bagian ini analisis terdiri dari arsitektur, *storyline*, tingkat kesulitan, *Gameplay*.

3.4.1 Arsitektur

Aplikasi ini dibangun dalam bentuk dua dimensi (2D), serta memiliki *fitur* seperti berikut.

Tabel 3. 7 Fitur *Game*

Fitur *Game*
1. *Single player* 2. Karakter lain hanya dapat digerakan sistem sebagai pembeli. 3. Tersedia beberapa lahan kosong untuk pembangunan rumah. 4. Tersedia system tawar menawar dalam penyewaan rumah. 5. Saldo tersedia untuk melakukan pembangunan rumah. 6. Memiliki tiga level permainan. 7. Setiap level mempunyai misi untuk mengumpulkan pendapatan yang berbeda. 8. Terdapat 3 jenis rumah yang dapat dibangun.

3.4.2 *Storyline*

Storyline merupakan sebuah alur cerita *Game*. Seorang pemuda diperintahakan oleh seorang pengusaha untuk mengolah lahan kosong yang dimilikinya agar menjadi asset berharga. Pemuda itu diberi modal untuk mengurusi tanah itu. Sehingga pemuda itu berpikir bagaimana caranya mengolah tanah ini menjadi asset berharaga. Lalu dia mendapatkan ide dengan merubah lahan kosong ini menjadi perumahan yang bisa disewakan. Penjelasan *storyline* dari tiap level akan dijelaskan di dalam tabel berikut :

Tabel 3. 8 *Storyline level* pertama

Level 1
Pada *storyline* level 1 akan tampil lahan kosong yang belum terdapat bangunan. Disini pemain akan membangun beberapa bangunan setelah memilih lahan kosong mana yang akan dibangun rumah. Rumah yang akan dibangun memiliki 3 macam type dan harga bangun yang berbeda-beda. Pemain dapat mengatur harga sewa rumah yang telah dibangun. Di level 1 ini akan tampil 1 jenis NPC yang diterapkan algoritma *fuzzy*, algoritma tersebut dipakai untuk memilih rumah yang akan disewa. Setelah NPC memilih rumah tersebut, pemain akan mendapatkan status tawaran menyewa dari NPC dan pemain pun harus menerima tawaran atau menolak tawaran NPC tersebut. Permainan di level 1 ini akan berakhir ketika pemain mendapatkan saldo yang ditargetkan dalam waktu yang ditentukan.

Tabel 3. 9 *Storyline level* kedua

Level 2
Pada *storyline* level 2 akan tampil lahan kosong yang belum terdapat bangunan. Disini pemain akan membangun beberapa bangunan setelah memilih lahan kosong mana yang akan dibangun rumah. Rumah yang akan dibangun memiliki 3 macam type dan harga bangun yang berbeda-beda. Pemain dapat mengatur harga sewa rumah yang telah dibangun. Di level 2 ini akan tampil 2 jenis NPC yang diterapkan algoritma *fuzzy*, algoritma tersebut dipakai untuk memilih rumah yang akan disewa. Setelah NPC memilih rumah tersebut, pemain akan mendapatkan status tawaran menyewa dari NPC dan pemain pun harus menerima tawaran atau menolak tawaran NPC tersebut. Permainan di level 2 ini akan berakhir ketika pemain mendapatkan saldo yang ditargetkan dalam waktu yang ditentukan. Saldo yang ditargetkan di level ini akan bertambah dan waktu berakhir permainannya pun akan berkurang.

Tabel 3. 10 *Storyline level* ketiga

Level 3
Pada *storyline* level 3 akan tampil lahan kosong yang belum terdapat bangunan. Disini pemain akan membangun beberapa bangunan setelah memilih lahan kosong mana yang akan dibangun rumah. Rumah yang akan dibangun memiliki 3 macam type dan harga bangun yang berbeda-beda. Pemain dapat mengatur harga sewa rumah yang telah dibangun. Di level 3 ini akan tampil 3 jenis NPC yang diterapkan algoritma *fuzzy*, algoritma tersebut dipakai untuk memilih rumah yang akan disewa. Setelah NPC memilih rumah tersebut, pemain akan mendapatkan status tawaran menyewa dari NPC dan pemain pun harus menerima tawaran atau menolak tawaran NPC tersebut. Permainan di level 3 ini akan berakhir ketika pemain mendapatkan saldo yang ditargetkan dalam waktu yang ditentukan. Saldo yang ditargetkan di level ini akan bertambah dan waktu berakhir permainannya pun akan berkurang.

3.4.3 *Gameplay*

Gameplay dilakukan untuk menggambarkan aturan-aturan dalam *Game*.

Terdapat *Gameplay Game* yang akan dibangun seperti berikut.

Tabel 3. 11 *Gameplay*

Gameplay
1. Pada awal permainan setiap level, pemain akan mendapatkan saldo untuk membangun rumah. Besar saldo yang didapatkan tergantung dari level yang sedang dimainkan. 2. Saldo digunakan sebagai alat tukar menukar pada *Game*. 3. Permainan memiliki batas waktu untuk menyelesaikan misi. 4. Saat prosses pelayanan, pemain mendapatkan waktu untuk bernegosiasi dengan pembeli. 5. Saldo akan berkurang jika proses pembangunan rumah telah dilakukan. 6. Karakter pembeli (sistem) akan muncul ketika pemain sudah membangun rumah. 7. Karakter pemain dapat menentukan sewa terhadap rumah yang telah dibangun. 8. Proses negosiasi hanya diberikan kesempatan tiga kali. 9. Setelah kesempatan negosiasi ketiga tapi belum ada kesepakatan, maka proses tawar menawar akan berakhir. 10. Pertama kali pembeli menawarkan harga, dan pemain dapat membuat kesepakatan maupun penawaran dari pemain.. 11. Pemain hanya dapat melayani satu pembeli 12. Pembeli yang lain akan menunggu jika pemain sedang melakukan proses

tawar menawar.
13. Pembeli akan meninggalkan arena permainan jika terlalu lama menunggu .
14. Rumah yang telah dibangun dapat diratakan kembali jika rumah tersebut dalam kondisi kosong.
15. Hasil tawar menawar akan mejadi saldo untuk pemain.

3.4.4 Misi

Misi yang harus diselesaikan yaitu mengumpulkan poin dari biaya pembangunan , sewa rumah rumah sebanyak mungkin. Besar rumah dan harga rumah menentukan poin penawaran. Terdapat 2 kondisi pada akhir permainan, yaitu permainan berhenti jika batas waktu habis, permainan dilanjutkan dengan adanya batasan waktu atau permainan dihentikan jika pemain telah memenuhi target poin.

3.4.5 Sasaran Pemain dan Kegunaan *Game*

Sasaran pemain menentukan layak tidaknya pemain memainkan permainan ini. Berikut adalah sasaran pemain yang layak memainkan permainan ini :

1. Berdasarkan umur pengguna.
2. Dapat mempergunakan komputer, karena *Game* berbentuk aplikasi dekstop.
3. Tidak buta total.
4. Dapat melakukan perhitungan dengan baik, karena proses pembangunan dan tawar menawar membutuhkan perhitungan agar mendapatkan keuntungan poin (saldo).

Kegunaan *Game* menunjukan kegunaan *Game* untuk pemain. Kegunaan dalam *Game* ini untuk meningkatkan daya pikir pemain dalam segi memanajemen biaya dan sebagai sarana hiburan karena pemain bebas bermain sesuai keinginan mereka untuk dapat menyelesaikan misi permainan.

3.5 Analisis Algoritma

Analisis algoritma yang dilakukan dalam penelitian ini adalah meneliti bagaimana cara kerja algoritma *fuzzy* pada NPC di dalam *Game*. Kondisi dimana NPC muncul dan berjalan menuju kondisi rumah yang telah dibangun.

3.5.1 Sistem Inferensi Fuzzy (Fuzzy Inference system)

Sistem inferensi *fuzzy* yang digunakan dalam perancangan sistem untuk mengambil keputusan dalam menentukan jalur teraman. Sistem inferensi *fuzzy* mempunyai salah satu metode yaitu mamdani, dimana mekanismenya sistem inferensi *fuzzy* bekerja dalam 4 tahap yaitu, pembentukan himpunan *fuzzy* (*fuzzyfikasi*), fungsi implikasi (aturan), komponen aturan, dan penegasan (*defuzzyfikasi*).

1. Pembentukan Himpunan *Fuzzy*

   Pada metode mamdani, baik variabel input maupun variabel output dibagi menjadi satu atau lebih himpunan *fuzzy*. Pada proses tersebut disebut juga proses *fuzzyfikasi*. *Fuzzyfikasi* adalah proses mengubah masukan eksak menjadi masukan berupa derajat keanggotaan.

A. Variabel *Fuzzy*

Variabel yang digunakan dalam tawar-menawar harga rumah di *Game building of house* adalah besar rumah dan harga rumah.

B. Himpunan *Fuzzy*

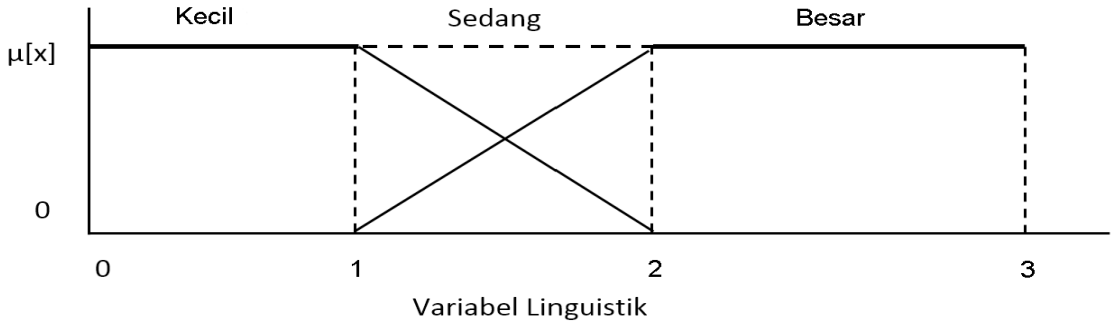
Dalam *Game building of house* ini, himpunan *fuzzy* untuk variabel ukuran rumah dibagi menjadi 3 kategori, yaitu sebagai berikut :

Kecil	ukuran rumah = 1
Sedang	1 > Ukuran rumah <= 2
Besar	2 > Ukuran rumah <= 3

Tabel 3. 12 Variabel Himpunan *fuzzy* untuk ukuran rumah

Variabel	Kecil	Sedang	Besar
Ukuran rumah harga murah	1	1 - 2	2 - 3

Ukuran rumah harga sedang	1	1 - 2	2 - 3
Ukuran rumah harga mahal	1	1 - 2	2 - 3



Gambar 3. 6 Himmpunan *fuzzy* pada Variabel ukuran rumah

Untuk variabel harga rumah dari yang murah, sedang dan mahal, perbandingan variabelnya adalah sebagai berikut :

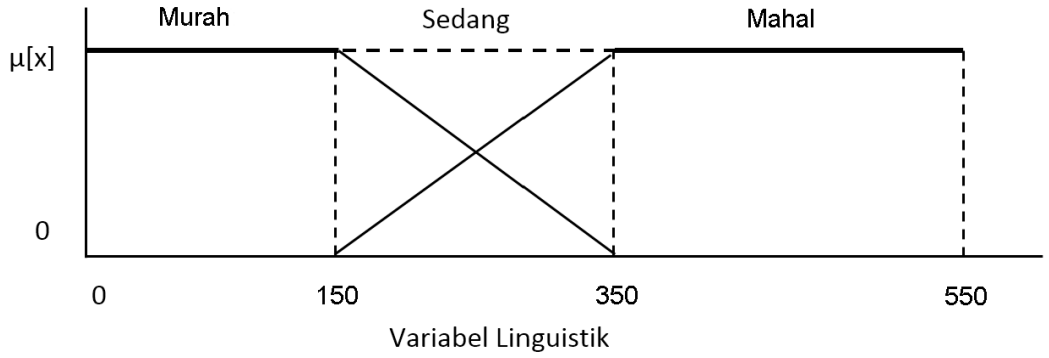
Murah Harga rumah <= 150

Sedang 150 > harga rumah <= 350

Mahal 350 > harga rumah <= 550

Tabel 3. 13 Variabel Himpunan *Fuzzy* untuk harga rumah

Variabel	Murah	Sedang	Mahal
Harga rumah dari ukuran rumah kecil	0 – 150	150 – 350	350 – 550
Harga rumah dari ukuran rumah sedang	0 – 150	150 – 350	350 – 550
Harga rumah dari ukuran rumah besar	0 – 150	150 – 350	350 – 550



Gambar 3. 7 Himmpunan *fuzzy* pada Variabel harga rumah

2. Fungsi Implikasi

Tiap-tiap aturan aturan pada basis *fuzzy* akan berhubungan dengan suatu relasi *fuzzy*. Variabel yang digunakan dalam sistem ini terdiri dari 6 *input* (ukuran rumah kecil harga murah, ukuran rumah kecil harga sedang, ukuran rumah kecil harga mahal, ukuran rumah sedang harga murah, ukuran rumah sedang harga sedang, ukuran rumah sedang harga mahal, ukuran rumah besar harga murah, ukuran rumah besar harga sedang, ukuran rumah besar harga mahal, harga rumah murah dari ukuran rumah kecil, harga rumah murah dari ukuran rumah sedang, harga rumah murah dari ukuran rumah besar, harga rumah sedang dari ukuran rumah kecil, harga rumah sedang dari ukuran rumah sedang, harga rumah sedang dari ukuran rumah besar, harga rumah mahal dari ukuran rumah kecil, harga rumah sedang dari ukuran rumah sedang, harga rumah mahal dari ukuran rumah besar) dan *output* (tentukan jalur teraman). Variabel harga rumah terdiri dari 3 himpunan *fuzzy* yaitu, murah, sedang, mahal. Variabel ukuran rumah terdiri dari 3 himpunan *fuzzy* yaitu, kecil, sedang, besar.

Tiap-tiap aturan pada basis pengetahuan *fuzzy* akan berhubungan dengan suatu relasi *fuzzy*, maka inferensi diperoleh dari kumpulan dan korelasi antar aturan. Metode yang digunakan dalam melakukan inferensi sistem *fuzzy* yaitu Metode Max (*Maximum*). Pada metode ini, solusi himpunan *fuzzy* diperoleh dengan cara mengambil nilai maksimum aturan,

kemudian menggunakannya untuk memodifikasi daerah *fuzzy*, dan mengaplikasikannya ke *output*.

Aturan yang diberikan di dalam implementasi *fuzzy* pada *Game* ini adalah sebagai berikut :

1) **IF** ukuran rumah kecil **AND** harga rumah murah **THEN** bagus.
2) **IF** ukuran rumah kecil **AND** harga rumah sedang **THEN** Sedang.
3) **IF** ukuran rumah kecil **AND** harga rumah mahal **THEN** kurang.
4) **IF** ukuran rumah sedang **AND** harga rumah murah **THEN** bagus.
5) **IF** ukuran rumah sedang **AND** harga rumah sedang **THEN** sedang.
6) **IF** ukuran rumah sedang **AND** harga rumah mahal **THEN** kurang.
7) **IF** ukuran rumah besar **AND** harga rumah murah **THEN** bagus.
8) **IF** ukuran rumah besar **AND** harga rumah sedang **THEN** sedang.
9) **IF** ukuran rumah besar **AND** harga rumah mahal **THEN** kurang.

3. Keluaran (*Defuzzyfikasi*)

   Defuzzyfikasi merupakan proses pengubahan besaran *fuzzy* yang disajikan dalam bentuk himpunan-himpunan *fuzzy* keluaran dengan fungsi keanggotaan. Keluaran yang dihasilkan merupakan suatu bilangan pada himpunan *fuzzy*.

   Input dari proses *defuzzyfikasi* adalah suatu himpunan *fuzzy* yang diperoleh dari komposisi aturan-aturan *fuzzy*, sedangkan *output* yang dihasilkan merupakan suatu bilangan pada himpunan *fuzzy* tersebut.

3.5.2 Analisis Metode pada NPC

Metode yang diterapkan pada NPC dalam *Game Building of House* digambarkan melalui *Pseudocode* dan *Diagram Activity* berikut:

Tabel 3. 14 *Psedocode Fuzzy* harga

```
Procedure Harga (Input : harga, Output : keputusan)
{ I.S : menghitung harga dan kondisi rumah }
{ F.S : menghasilkan status harga dari rumah }

Kamus
harga, posisiHome, posisiTree : real;

Algoritma

Begin
read(posisiHome, posisiTree);

i := 0;

for i to 600
        if posisiHome != null AND posisiHome != posisiTree then
                harga += 1;
        endif
        i += 150;
endfor

if harga == 0 then
                write ("murah");
        else if harga > 0 AND harga <= 150 then
                write ("murah");
                else if harga >150 AND harga <= 350 then
                        write ("sedang");
                        else if harga > 350 AND harga <= 550 then
                                write ("mahal");
                        endif
                endif
        endif
endif
return ( harga );
End
```

Tabel 3. 15 *Psedocode Fuzzy* ukuran

```
Procedure Ukuran (Input : ukuran, Output : keputusan)
{ I.S : menghitung ukuran dan kondisi rumah }
{ F.S : menghasilkan status ukuran dari rumah }

Kamus
ukuran, posisiHome, posisiTree : real;
```

```
Algoritma

Begin
read(posisiHome, posisiTree);

i := 0;

for i to 600
        if posisiHome != null AND posisiHome != posisiTree then
                ukuran += 1;
        endif
        i += 150;
endfor

if ukuran == 1 then
                write ("kecil");
        else if ukuran > 1 AND ukuran <= 2 then
                write ("sedang");
                else if ukuran > 2 AND ukuran <= 3 then
                        write ("besar");
                endif
        endif
endif
return ( ukuran );
End
```

Tabel 3. 16 ***Psedocode Defuzzyfikasi***

```
Procedure Defuzzyfikasi (Input : ukuran, harga, Output : keputusan)
{ I.S : menghitung defuzzyfikasi dari harga dan ukuran rumah }
{ F.S : menghasilkan status defuzzyfikasi dari rumah }

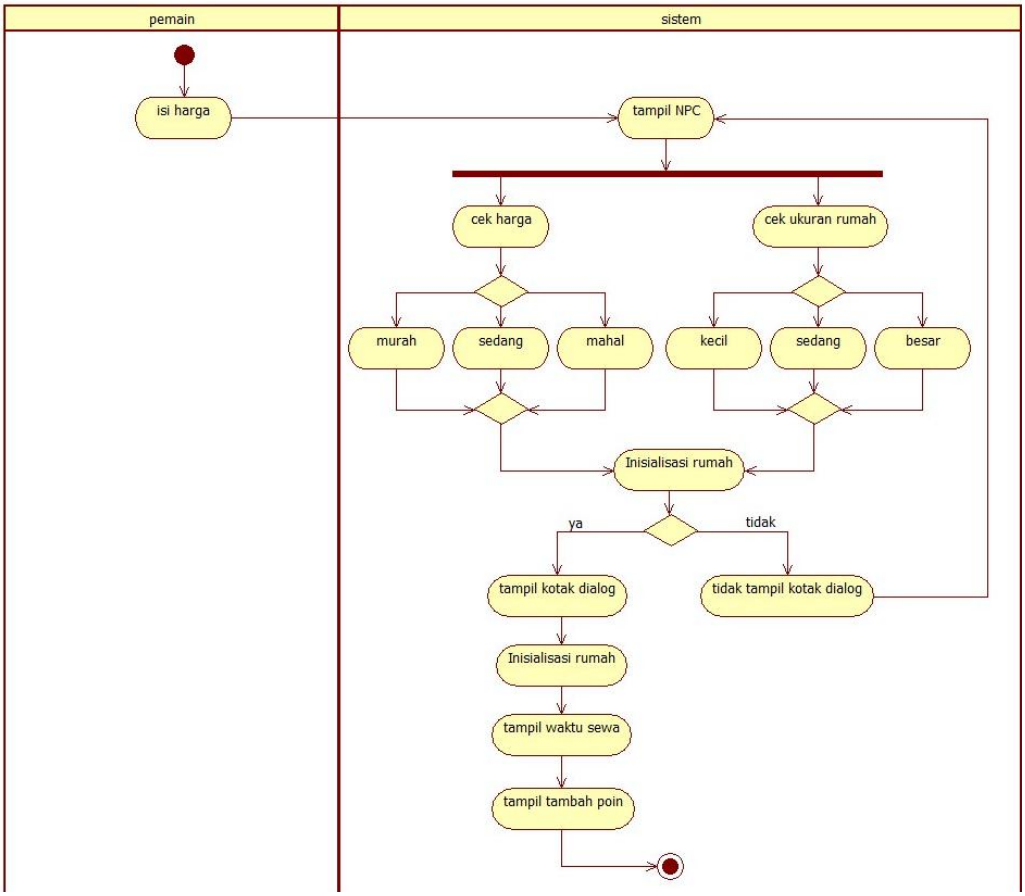
Kamus
ukuran, harga, defuzifikasi: real;

Algoritma

Begin
read(harga);
read(ukuran);
read(defuzifikasi) ← (ukuran + harga);

if defuzifikasi = 1 AND <= 150 then
                write ("bagus");
        else if defuzifikasi = 1 AND <= 350 then
```

```
                write ("sedang");
                else if defuzifikasi = 1 AND <= 550 then
                        write ("kurang");
                        else if defuzifikasi = 2 AND <= 150 then
                                write ("bagus");
                                else if defuzifikasi = 2 AND <= 350 then
                                        write ("sedang");
                                        else if defuzifikasi = 2 AND 550 then
                                                write ("kurang");
                                                else if defuzifikasi = 3 AND <= 150
then
                                                        write ("bagus");
                                                        else if defuzifikasi = 3 AND
<= 350 then
                                                                write ("sedang");
                                                                else if defuzifikasi =
3 AND 550 then
                                                                        write
("kurang");
                                                                endif
                                                        endif
                                                endif
                                        endif
                                endif
                        endif
                endif
        endif
endif
End
```



Gambar 3. 8 Penerapan Metode *Fuzzy* pada NPC

3.6 Analisis Kebutuhan Non-Fungsional

Analisis kebutuhan non-fungsional dilakukan untuk mengetahui spesifikasi kebutuhan untuk sistem.Spesifikasi kebutuhan melibatkan analisis perangkat keras/*hardware*, analisis perangkat lunak/*software*, analisis perangkat pikir/*user*.

3.6.1 Analisis Kebutuhan Perangkat Keras

Analisis kebutuhan perangkat keras digunakan untuk mengetahui spesifikasi minimun perangkat keras dalam membangun *Game Building of House* pada tabel 3.17 dan menjalankan *Game Building of House* pada tabel 3.18.

Tabel 3. 17 Spesifikasi Perangkat Keras Pembangun

Nama Perangkat	Spesifikasi
Processor	1.80Ghz
RAM	4GB
Harddisk	50 GB
Display	Resolusi 1366 x 768
Keyboard	Standar
Mouse	Standar

Tabel 3. 18 Spesifikasi Perangkat Keras Pengguna

Nama Perangkat	Spesifikasi
Processor	1.80Ghz
RAM	512MB
Harddisk	200MB
Display	Resolusi 800 x 600
Keyboard	Standar
Mouse	Standar

3.6.2 Analisis Kebutuhan Perangkat Lunak

Perangkat lunak digunakan dalam sebuah sistem merupakan perintah yang diberikan kepada perangkat keras agar dapat saling berinteraksi diantaranya. Perangkat lunak yang dibutuhkan untuk pembangunan *Game Building of House* ini adalah seperti pada tabel 3.19 dan 3.20.

Tabel 3. 19 Spesifikasi Kebutuhan Perangkat Lunak Pembangun

Sistem operasi	Windows 7 *Ultimate* 32-bit
Compiler	Construct 2
Development toolkit	HTML 5
Design tools	Adobe Photoshop CS 3
Design analysis	StarUML

Tabel 3. 20 Spesifikasi Kebutuhan Perangkat Lunak Pengguna

Operating System	Windows 7 *Ultimate* 32-bit
Tools	Mozilla Firefox

3.6.3 Analisis Kebutuhan Perangkat Pikir

Analisis kebutuhan perangkat pikir digunakan untuk mengetahui spesifikasi pengguna untuk dapat memainkan *Game building of house*, sebagai berikut.

Tabel 3. 21 Spesifikasi Pengguna

Peran	**Pemain**
Usia	13 tahun ke atas
Tingkat Keterampilan	Dapat menggunakan komputer
Kondisi Fisik	Tidak cacat

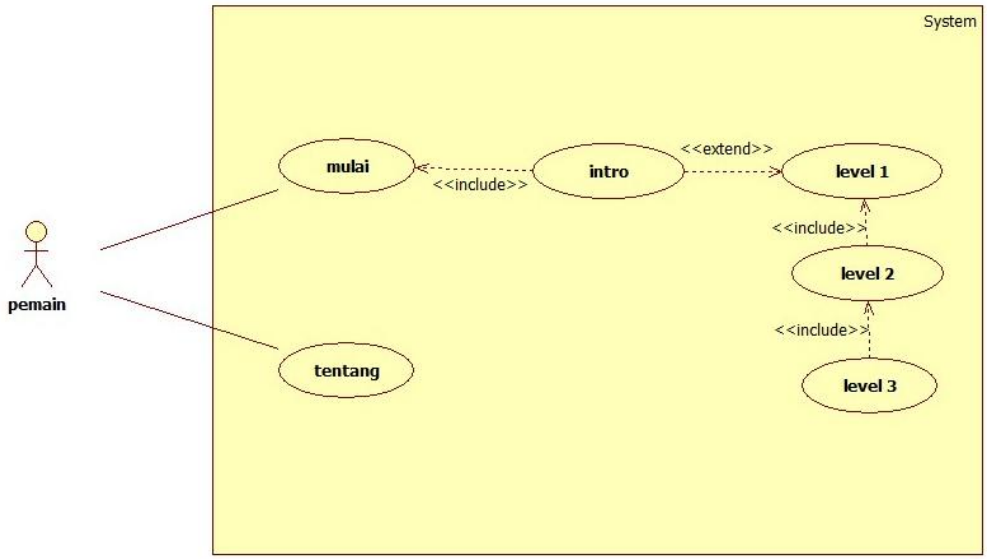
3.7 Analisis Kebutuhan Fungsional

Analisis kebutuhan fungsional dilakukan untuk mengetahui gambaran sistem yang akan dibangun. Analisis kebutuhan fungsional terdiri dari pendefinisan fungsi *Game*, penggambaran alir aktivitas pada *Game* dan penggambaran interaksi antar objek pada *Game*.

Pendefinisian fungsi *Game* digunakan untuk menggambarkan fungsi-fungsi pada *Game*. Pendefinisian fungsi pada *Game* ini digambarkan dengan menggunakan *use case diagram*, definisi aktor,definisi*use case* dan skenario *use case*.

3.7.1 *Use Case* Diagram

Penggambaran fungsi *Game* berdasarkan interaksi antar aktor dan objek pada sistem digambarkan dengan menggunakan diagram use case seperti pada gambar.



Gambar 3. 9 *Use Case* Diagram

1.1. Definisi Aktor

Definisi aktor berfungsi menjelaskan peran setiap aktor yang terdapat pada use case. Berikut ini adalah definisi aktor.

Tabel 3. 22 Definisi Aktor

No.	Aktor	Deskripsi
1.	Pemain	Pengguna/orang yang memainkan *Game* ini.

1.2. Definisi *Use Case*

Definisi *use case* berfungsi menjelaskan tiap-tiap proses dan objek yang terdapat pada *use case*. Berikut ini adalah definisi *use* case.

Tabel 3. 23 Definisi *Use Case*

No.	Use Case	Deskripsi
1.	Mulai	Proses memulai permainan
2.	Intro	Proses menampilkan petunjuk permainan
3.	Level 1	Proses memulai permainan level 1
4.	Level 2	Proses memulai permainan level 2
5.	Level 3	Proses memulai permainan level 3
6.	Tentang	Proses menampilkan informasi permainan

1.3. Skenario *Use Case*

Skenario *use case* menunjukan proses apa yang terjadi pada setiap bagian di dalam *use case* tersebut, dimana *user* memberikan perintah pada bagian-bagian *use case* dan respon apa yang diberikan oleh sistem kepada *user* setelah *user* memberikan perintahnya. Berikut ini adalah skenario *use case.*

Tabel 3. 24 Skenario *Use Case* Mulai

Identifikasi	
Nomor	1
Nama	Mulai
Tujuan	Memulai permainan
Deskripsi	Proses untuk memulai permainan
Aktor	Pemain
Skenario Utama	
Kondisi Awal	Pemain berada di menu utama *Game*
Aksi Aktor	Reaksi Sistem
1. Tekan mulai	
	2. Menampilkan petunjuk permainan
3. Tekan selesai	
	4. Menampilkan permainan

Tabel 3. 25 Skenario *Use Case* Tentang

Identifikasi	
Nomor	2
Nama	Tentang
Tujuan	Menampilkan informasi permainan
Deskripsi	Proses untuk menampilkan informasi permainan
Aktor	Pemain
Skenario Utama	
Kondisi Awal	Pemain berada di menu utama *Game*
Aksi Aktor	Reaksi Sistem
1. Tekan tentang	
	2. Menampilkan informasi permainan
3. Tekan selesai	
	4. Menampilkan halaman utama

Tabel 3. 26 Skenario *Use Case Intro*

Identifikasi	
Nomor	3
Nama	Intro
Tujuan	Menampilkan petunjuk permainan
Deskripsi	Proses untuk menampilkan petunjuk permainan
Aktor	Pemain
Skenario Utama	
Kondisi Awal	Pemain berada di halaman intro
Aksi Aktor	Reaksi Sistem
1. Tekan mulai	
	2. Menampilkan intro
3. Menekan tombol lanjut	
	4. Menampilkan petunjuk
5. Menekan tombol selesai	
	6. Menampilkan permainan

Tabel 3. 27 Skenario *Use Case Level* 1

Identifikasi	
Nomor	4
Nama	Level 1
Tujuan	Menampilkan permainan level 1
Deskripsi	Proses untuk memainkan permainan level 1
Aktor	Pemain
Skenario Utama	
Kondisi Awal	Pemain berada di tampilan permainan level 1
Aksi Aktor	Reaksi Sistem
1. Memulai permainan	
	2. Inisialisasi Permainan
	3. Menampilkan Arena Permainan
4. Tekan membangun dan	
5. Tekan arena permainan	
	6. Tampil kotak harga
7. Ketik harga	
8. Tekan selesai	
	9. Tampil rumah
	10. Tampil npc pembeli
	11. Inisialisasi fuzzy
	12. Ukuran rumah
	13. Harga rumah
	14. Tampil kotak dialog

15. Tekan lanjut	
16. Setuju	
17. Tekan selesai	
	18. Tampil waktu sewa
	19.
	20. Tampil info permainan berakhir
Skenario Tambahan	
21. Tidak setuju	
	22. Tampilan npc pembeli

Tabel 3. 28 Skenario *Use Case Level* 2

Identifikasi	
Nomor	5
Nama	Level 2
Tujuan	Menampilkan permainan level 2
Deskripsi	Proses untuk memainkan permainan level 2
Aktor	Pemain
Skenario Utama	
Kondisi Awal	Pemain berada di tampilan permainan level 2
Aksi Aktor	Reaksi Sistem
1. Memulai permainan	
	2. Inisialisasi Permainan
	3. Menampilkan Arena Permainan
4. Tekan membangun dan	
5. Tekan arena permainan	
	6. Tampil kotak harga
7. Ketik harga	
8. Tekan selesai	
	9. Tampil rumah
	10. Tampil npc pembeli
	11. Inisialisasi fuzzy
	12. Ukuran rumah
	13. Harga rumah
	14. Tampil kotak dialog
15. Tekan lanjut	
16. Setuju	
17. Tekan selesai	
	18. Tampil waktu sewa
	19. Tampil tambahan poin
	20. Tampil info permainan berakhir
Skenario Tambahan	
21. Tidak setuju	

	22. Tampilan npc pembeli

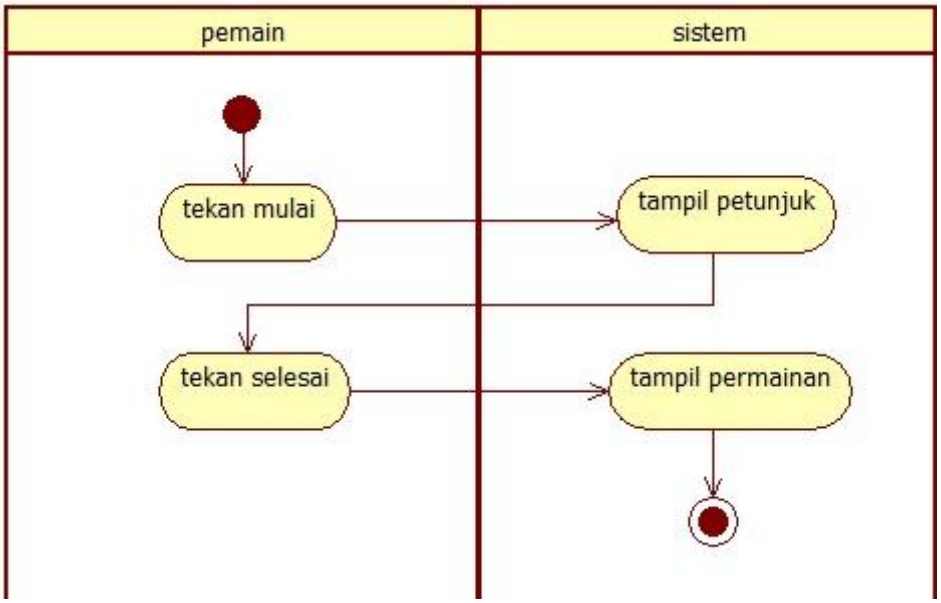
Tabel 3. 29 Skenario *Use Case Level* 3

Identifikasi	
Nomor	6
Nama	Level 3
Tujuan	Menampilkan permainan level 3
Deskripsi	Proses untuk memainkan permainan level 3
Aktor	Pemain
Skenario Utama	
Kondisi Awal	Pemain berada di tampilan permainan level 3
Aksi Aktor	Reaksi Sistem
1. Memulai permainan	
	2. Inisialisasi Permainan
	3. Menampilkan Arena Permainan
4. Tekan membangun dan	
5. Tekan arena permainan	
	6. Tampil kotak harga
7. Ketik harga	
8. Tekan selesai	
	9. Tampil rumah
	10. Tampil npc pembeli
	11. Inisialisasi fuzzy
	12. Ukuran rumah
	13. Harga rumah
	14. Tampil kotak dialog
15. Tekan lanjut	
16. Setuju	
17. Tekan selesai	
	18. Tampil waktu sewa
	19. Tampil tambahan poin
	20. Tampil info permainan berakhir
Skenario Tambahan	
21. Tidak setuju	
	22. Tampilan npc pembeli

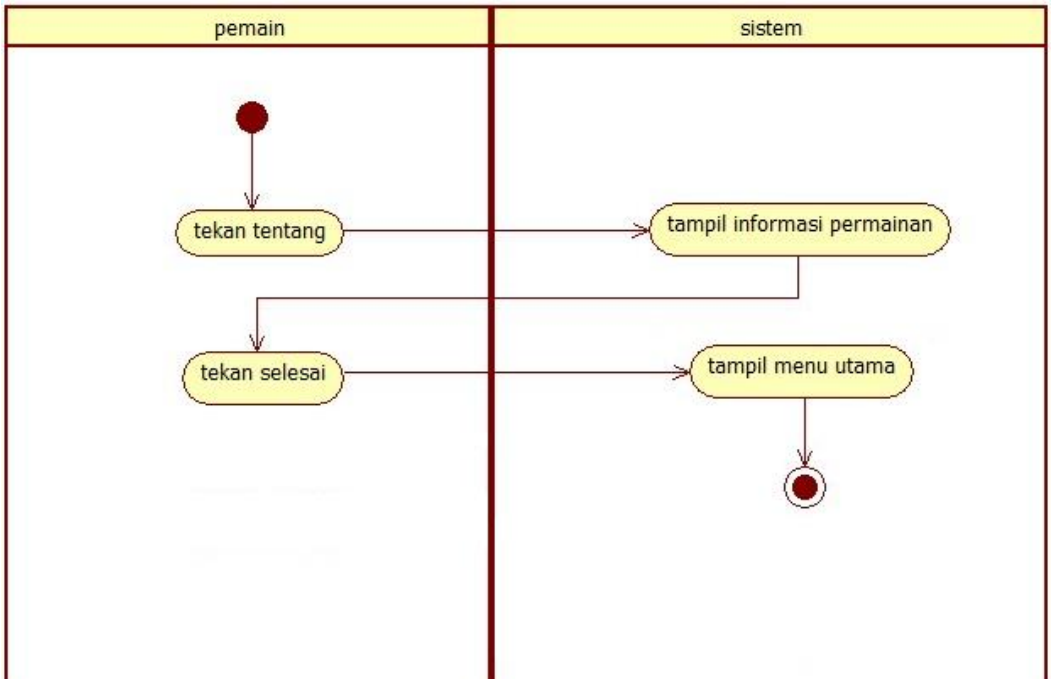
3.7.2 Penggambaran *Activity* Diagram

Penggambaran alir aktivitas pada *Game* menggambarkan berbagai alir aktivitas dalam *Game* yang sedang dirancang, bagaimana masing-masing alir berawal, *decision* yang mungkin terjadi, dan bagaimana mereka berakhir.

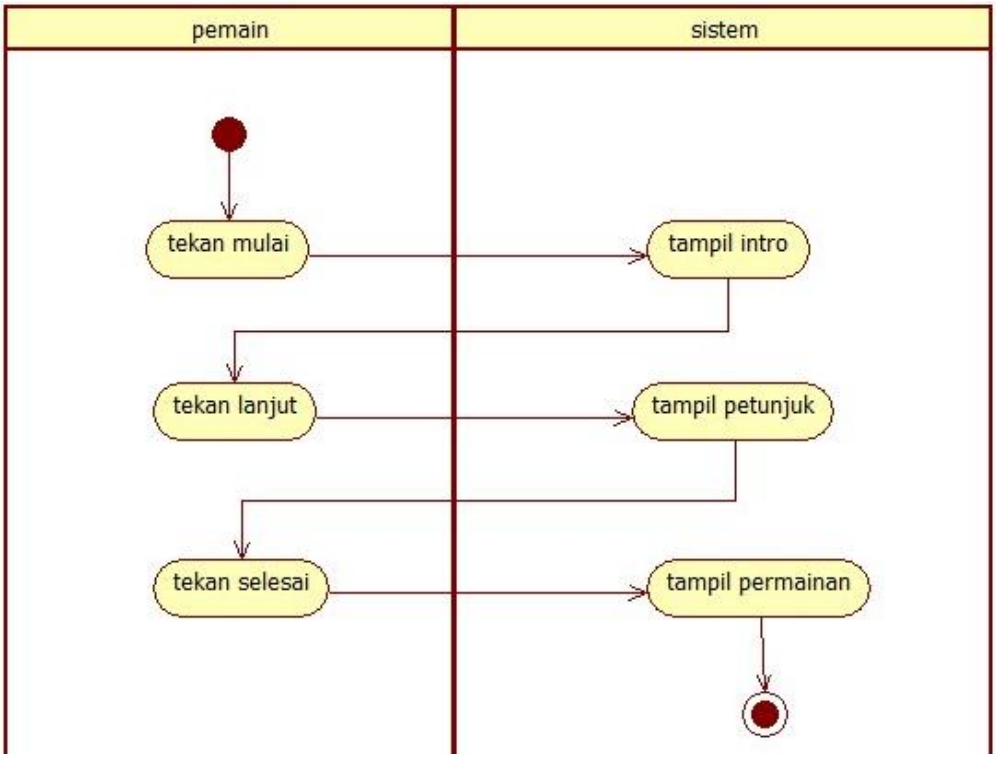
Penggambaran alir aktivitas pada *Game* digambarkan menggunakan *Activity Diagram*. *Activity diagram* merupakan *state diagram* khusus, dimana sebagian besar *state* adalah *action* dan sebagian besar transisi di-*trigger* oleh selesainya *state* sebelumnya (*internal processing)*. Oleh karena itu *activity diagram* tidak menggambarkan behavior internal sebuah sistem (dan interaksi antar subsistem) secara eksak, tetapi lebih menggambarkan proses-proses dan jalur-jalur aktivitas dari level atas secara umum. Sebuah aktivitas dapat direalsasikan oleh satu *Use Case* atau lebih. Aktivitas menggambarkan proses yang berjalan, sementara *Use Case* menggambarkan bagaimana actor menggunakan sisitem untuk melakukan aktivitas. *Activity diagram* ditunjukan seperti berikut.



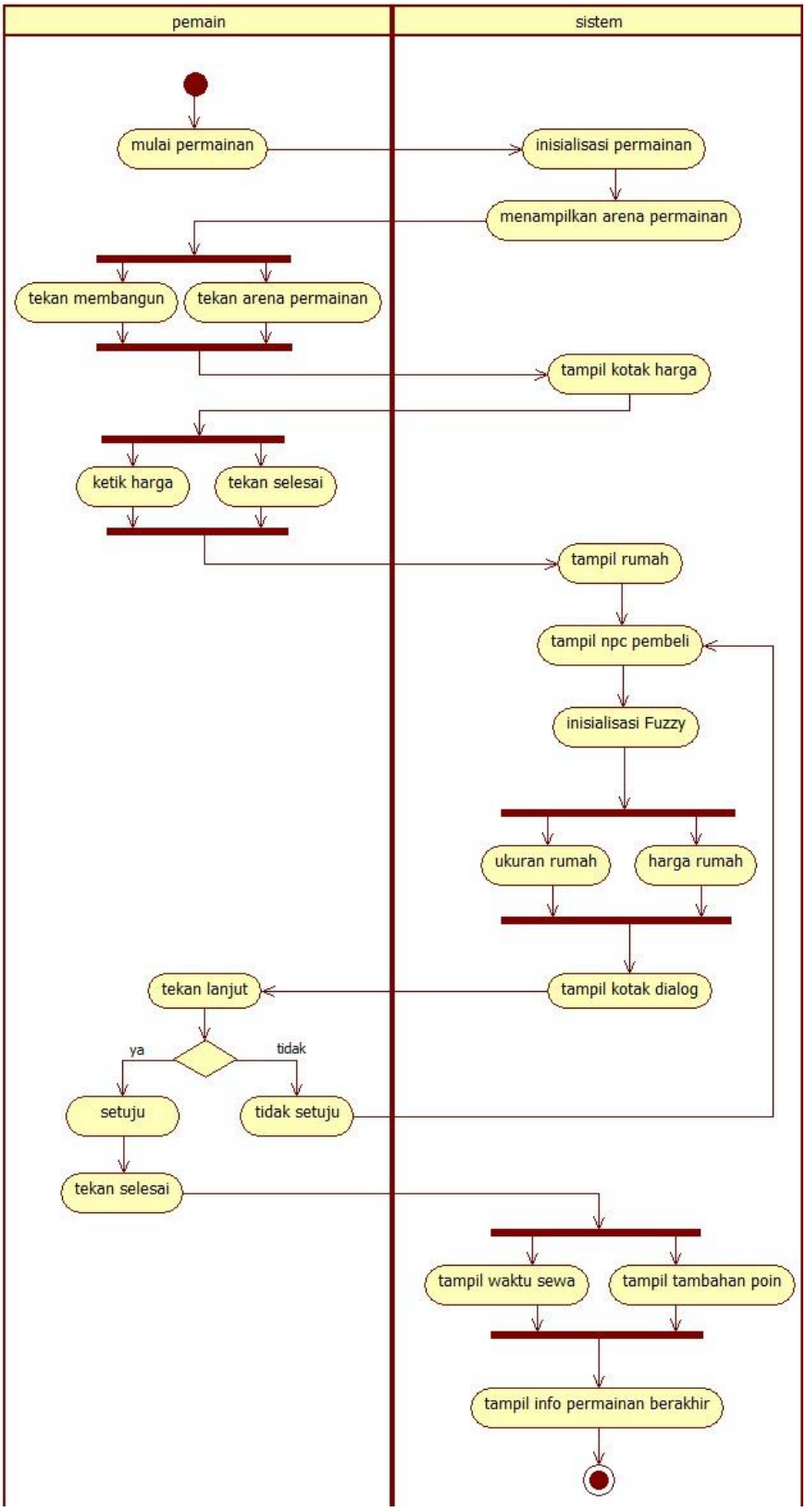
Gambar 3. 10 *Activity* Diagram Mulai



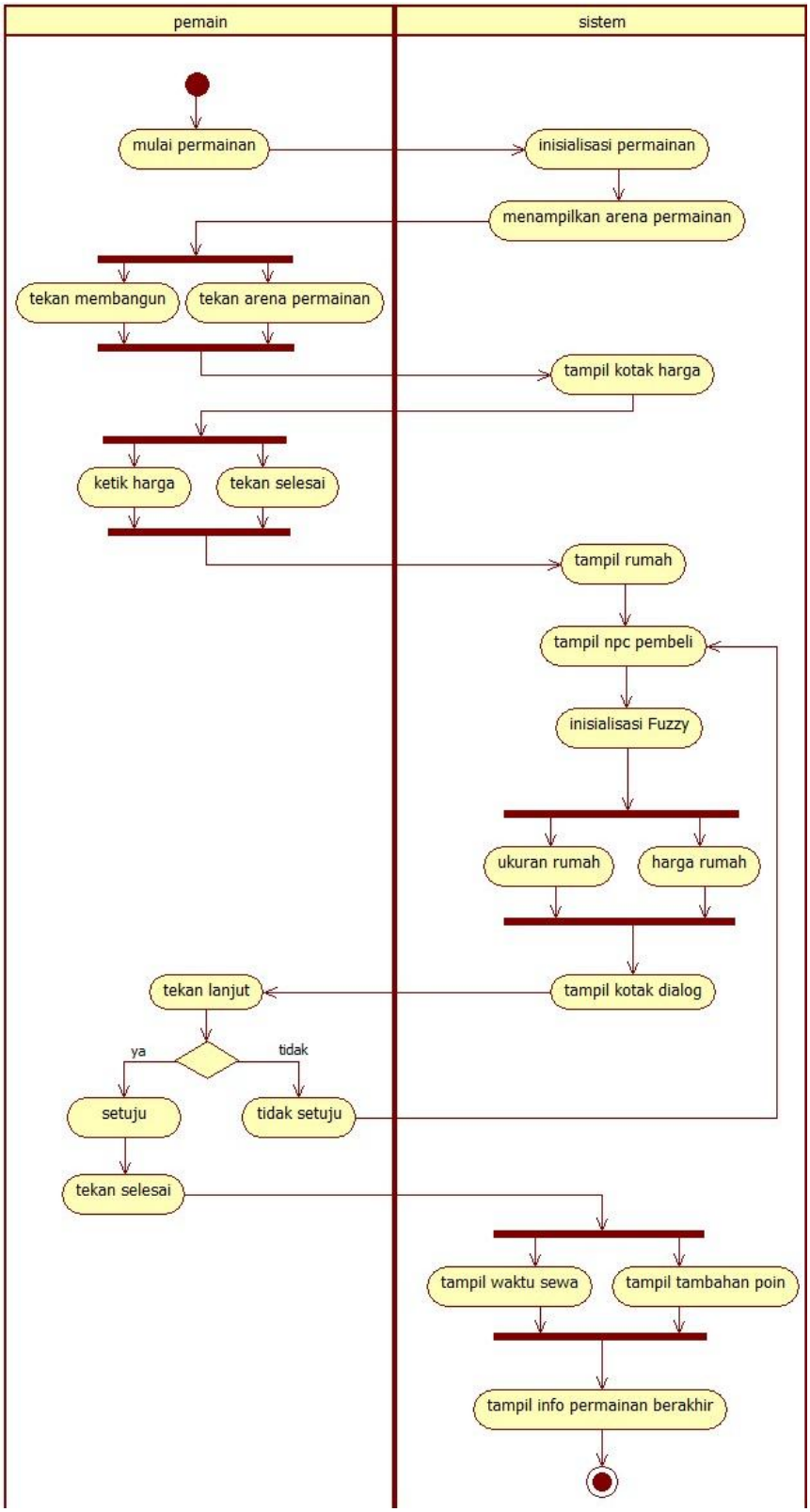
Gambar 3. 11 *Activity* Diagram Tentang



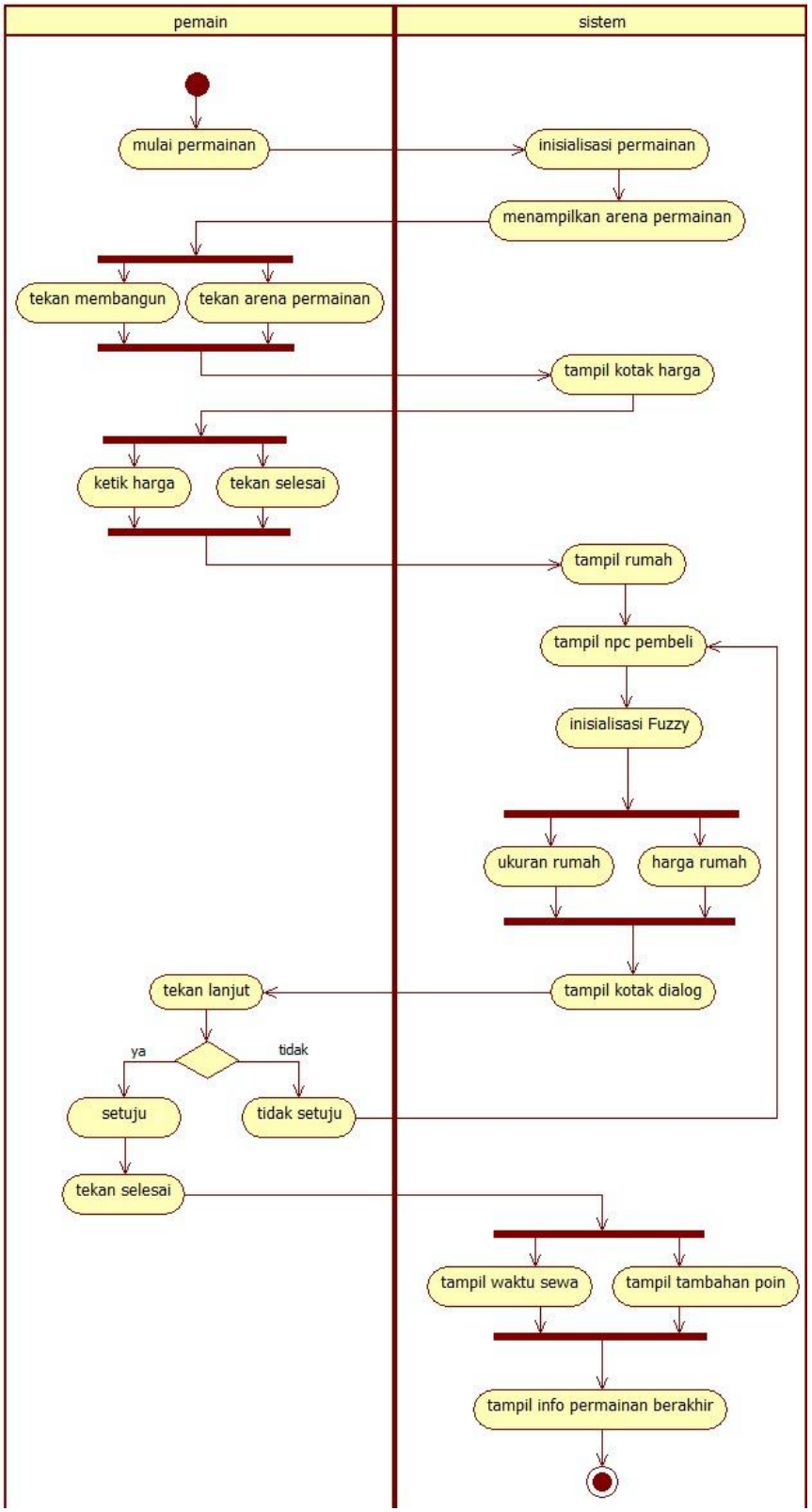
Gambar 3. 12 *Activity* Diagram *Intro*



Gambar 3. 13 *Activity* Diagram *Level* 1



Gambar 3. 14 *Activity* Diagram *Level* 2



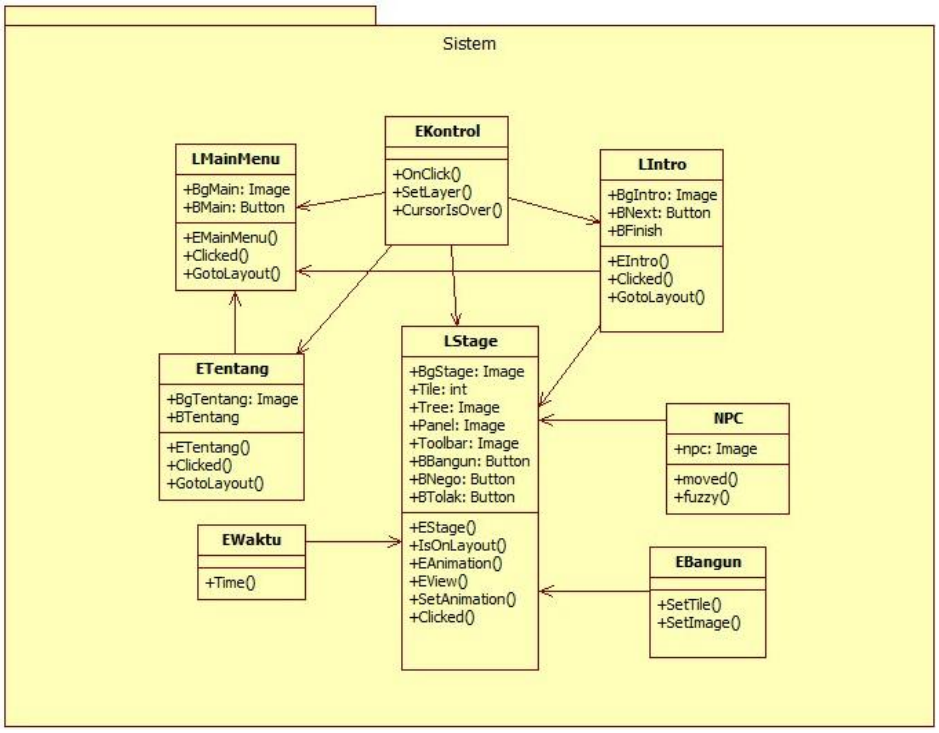
Gambar 3. 15 *Activity* Diagram *Level* 3

3.7.3 Penggambaran Kelas

Kelas digambarkan dengan *class diagram*. Diagram kelas atau *class diagram* menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem. Kelas memiliki apa yang disebut atribut dan metode atau operasi. *Class diagram* ditunjukan seperti berikut:

Tabel 3. 30 Definisi Kelas

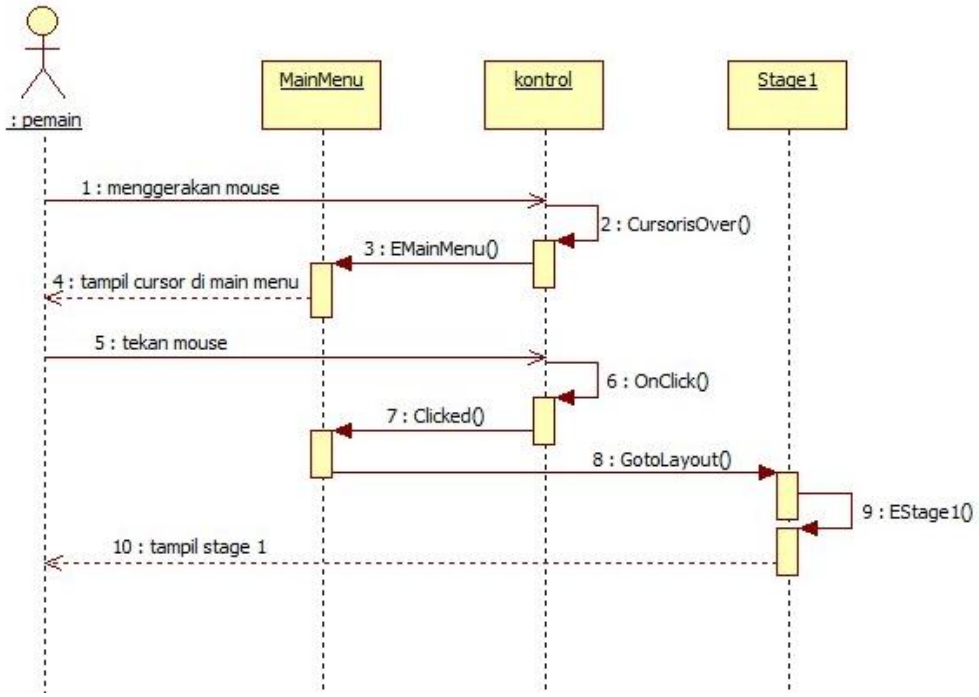
Jenis Kelas	Nama Kelas
Boundary	LMainMenu
	LIntro
	LStage
	EKontrol
	EWaktu
	NPC
	EBangun
	ETentang



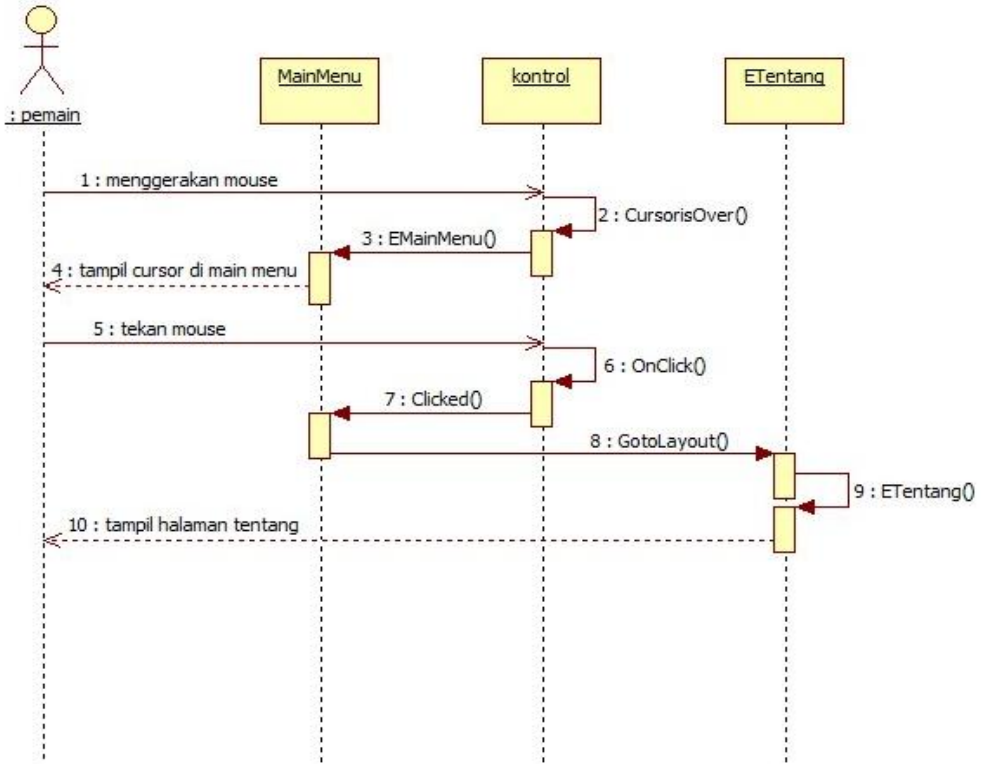
Gambar 3. 16 *Class* Diagram

3.7.4 Penggambaran interaksi objek pada *Game*

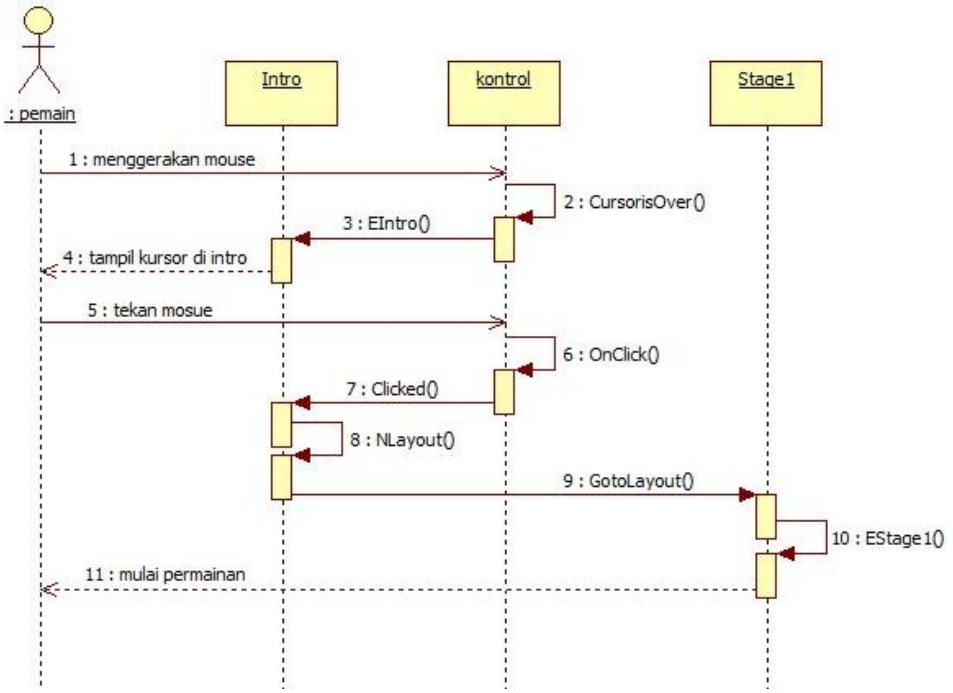
Penggambaran interaksi antar objek menggambarkan interaksi antar objek yang terjadi di dalam dan di luar sistem pada *Game* (termasuk *player*, *display*, dan sebagainya). Penggambaran interaksi antar objek pada *Game* menggunakan *Sequence Diagram*. *Sequence diagram* terdiri atas dimensi *vertical* (waktu) dan dimensi *horizontal* (objek-objek yang terkait). Pada *sequence diagram* ini menggambarkan scenario atau rangkaian langkah-langkah yang dilakukan pada sistem sebagai respon dari sebuah *event* untuk menghasilkan *output* tertentu. Berikut ini adalah *Sequence diagram* pada *Game* ini.



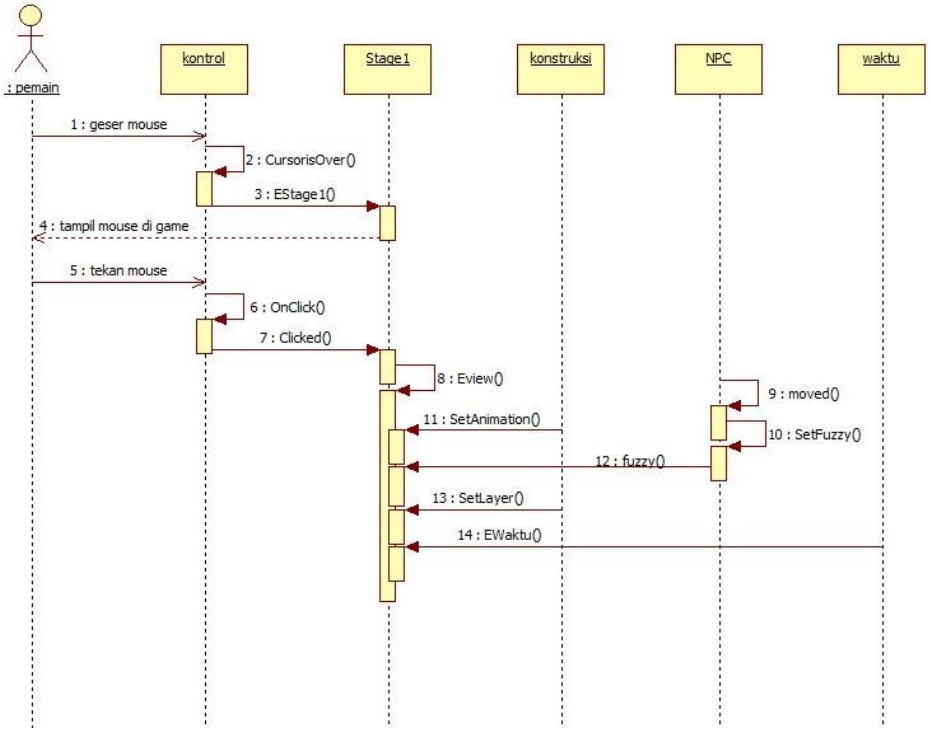
Gambar 3. 17 *Sequence* Diagram Mulai



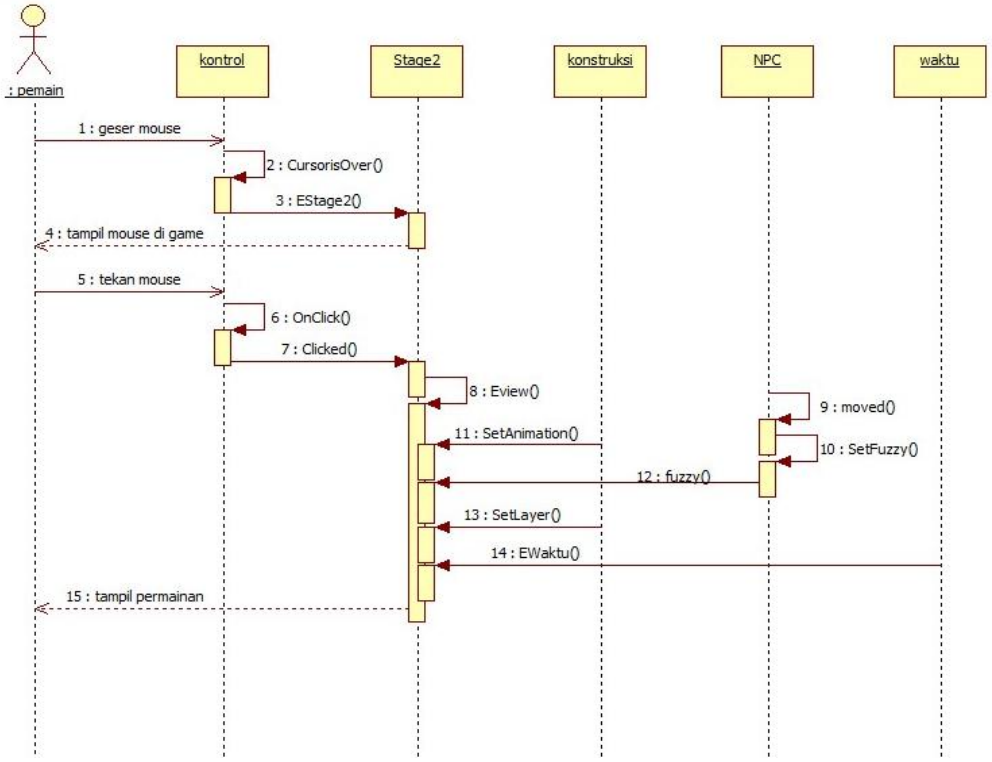
Gambar 3. 18 *Sequence* Diagram Tentang



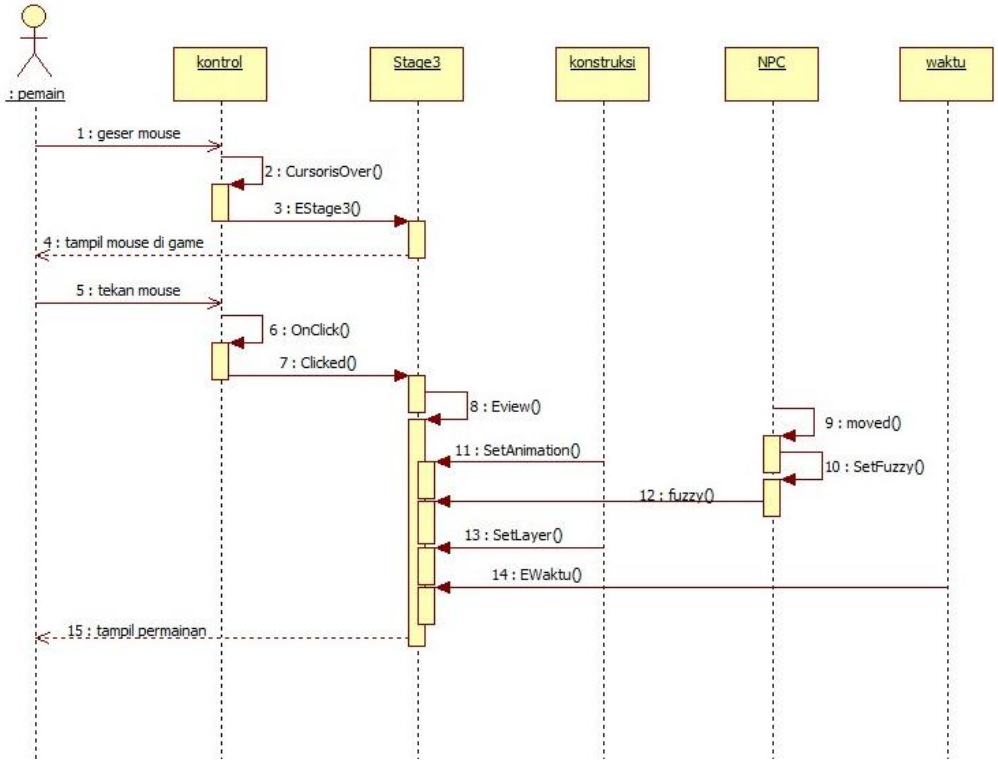
Gambar 3. 19 *Sequence* Diagram *Intro*



Gambar 3. 20 *Sequence* Diagram *Level* 1



Gambar 3. 21 *Sequence* Diagram *Level* 2



Gambar 3. 22 *Sequence* Diagram *Level* 3

3.8 Perancangan Sistem

Tahap perancangan merupakan kelanjutan dari proses analisis dimana dilakukan perubahan-perubahan terhadap sistem yang sedang berjalan. Hal ini dilakukan untuk mengatasi kekurangan yang ada, memudahkan pekerjaan yang dilakukan oleh orang yang terlibat dan menghemat waktu pekerjaan.

Perancangan *Game* ini meliputi perancangan struktur menu, perancangan antarmuka, perancangan pesan, jaringan semantik, perancangan prosedural. Semua perancangan yang telah disebutkan diatas harus saling berhubungan agar menciptakan suatu sistem yang baik.

3.8.1 Perancangan Komponen Permainan

Perancangan komponen permainan merupakan bagian pendeskripsian setiap komponen yang ada pada *Game* baik karakter, bangunan, dan *story board*. Perancangan komponen permainan terdiri dari perancangan karakter dan perancangan *story board*.

3.8.1.1 Perancangan Karakter

Perancangan karakter adalah bagian dimana karakter pada *Game* dijelaskan ciri-cirinya baik itu secara fungsionalitas peran maupun secara desain. Pada *Game* ini, karakter merupakan setiap unit pada *Game*, sehingga pada perancangan karakter akan mendeskripsikan setiap kelas unit pada setiap fraksi. Perancangan karakter pada *Game* ini dijabarkan seperti berikut.

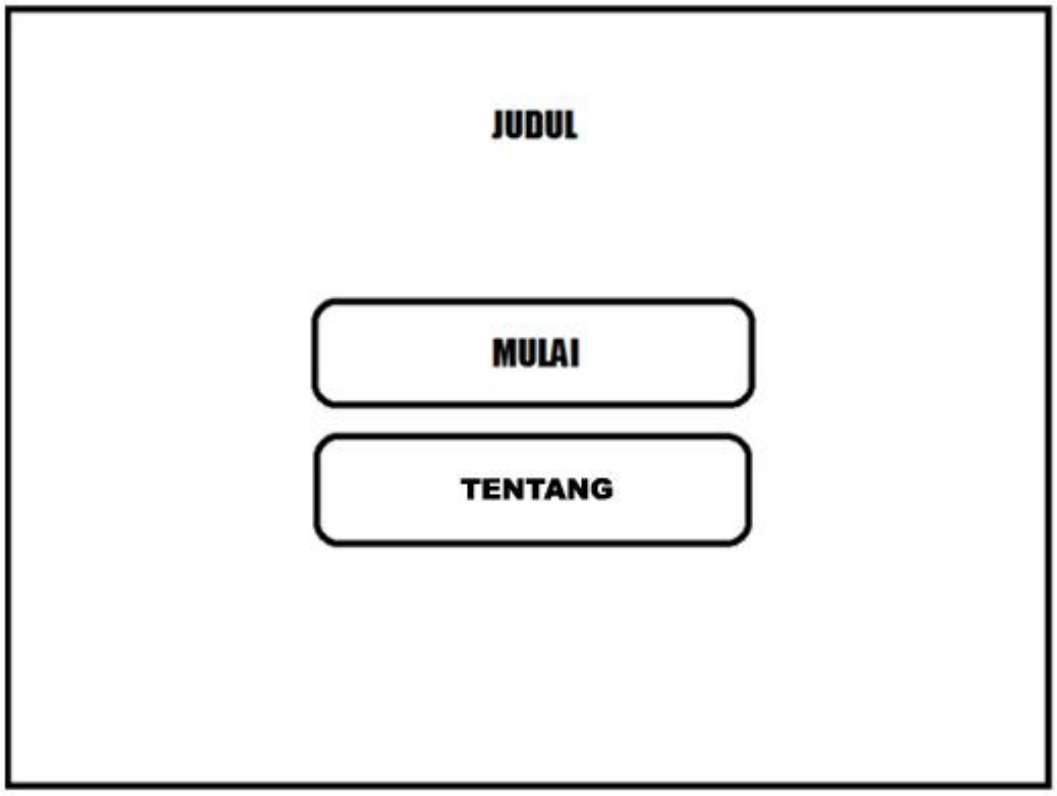
Tabel 3. 31 Perancangan Karakter

No	Nama	Gambar	Keterangan
1	Rumah tipe A		Ini adalah salah satu karakter rumah yang dapat dibangun. Rumah ini memilki ukuran kecil dan mempunyai harga bangun yaitu 150.
2	Rumah tipe B		Ini adalah salah satu karakter rumah yang dapat dibangun. Rumah ini memilki ukuran sedang dan mempunyai harga bangun yaitu 350.
3	Rumah tipe S		Ini adalah salah satu karakter rumah yang dapat dibangun. Rumah ini memilki ukuran besar dan mempunyai harga bangun yaitu 550.
4	Pelanggan		Ini adalah karakter NPC. Karakter ini dapat mencari rumah yang telah dibangun.

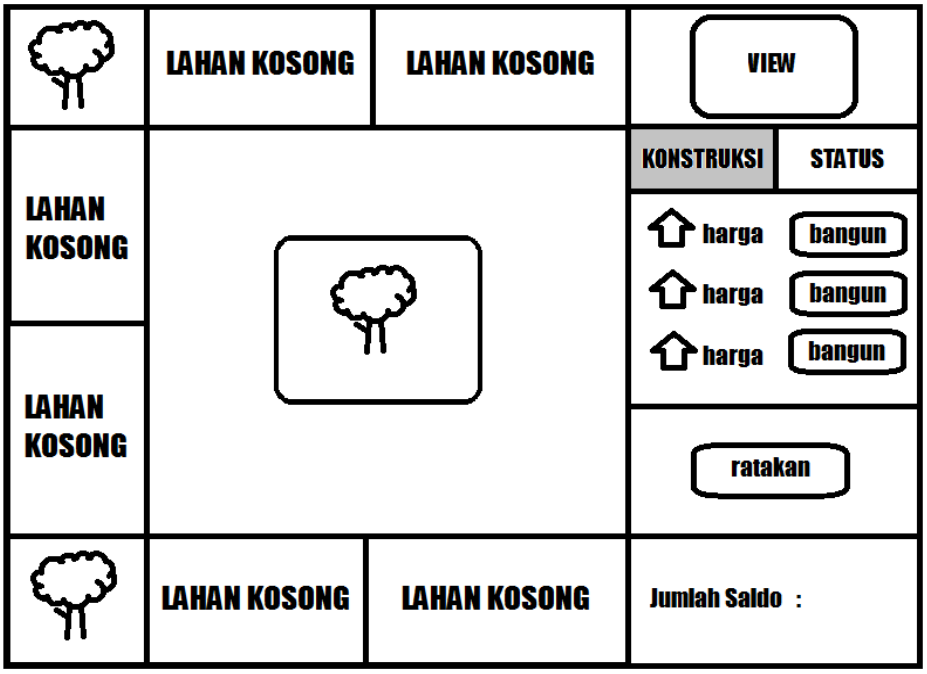
5	Pekerja bangunan		Ini adalah karakter NPC. Karakter ini dapat bekerja ketika sedang membangun rumah.
6	Sold out		Ini adalah navigasi atau penunjuk bahwa rumah telah tersewa.
7	Area		Ini adalah area pemukiman untuk pembangunan rumah.

3.8.1.2 Perancangan *Storyboard*

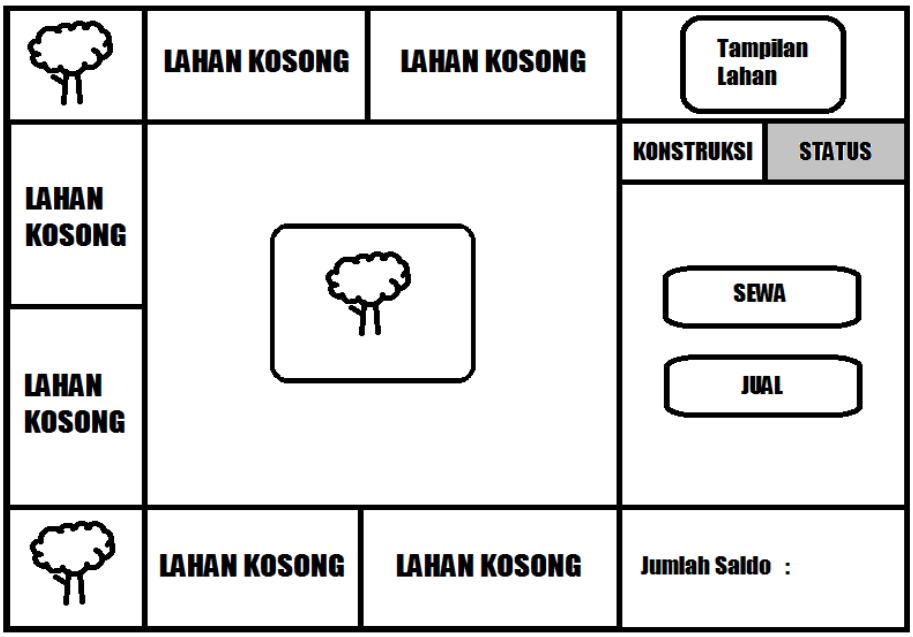
Storyboard merupakan sketsa gambar yang disusun secara berurutan berdasarkan naskah untuk menggambarkan lingkungan pada *Game*. Pada *Game* ini, storyboard merupakan sebuah *map* (peta) yang akan digunakan untuk bermain dan diterapkan pada sebuah stage. *Map* pada *Game* ini digambarkan seperti berikut.



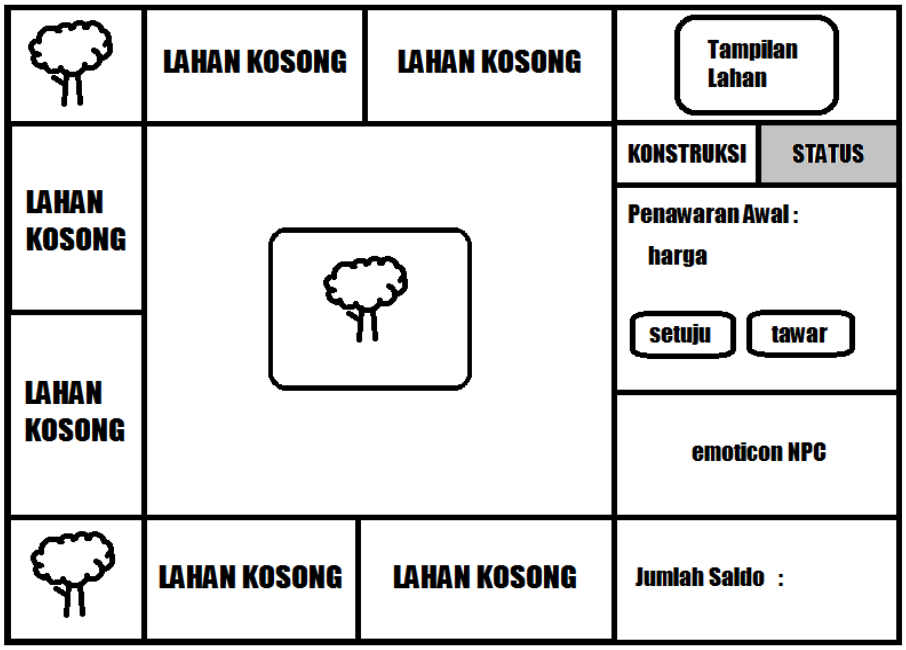
Gambar 3. 23 Menu Utama



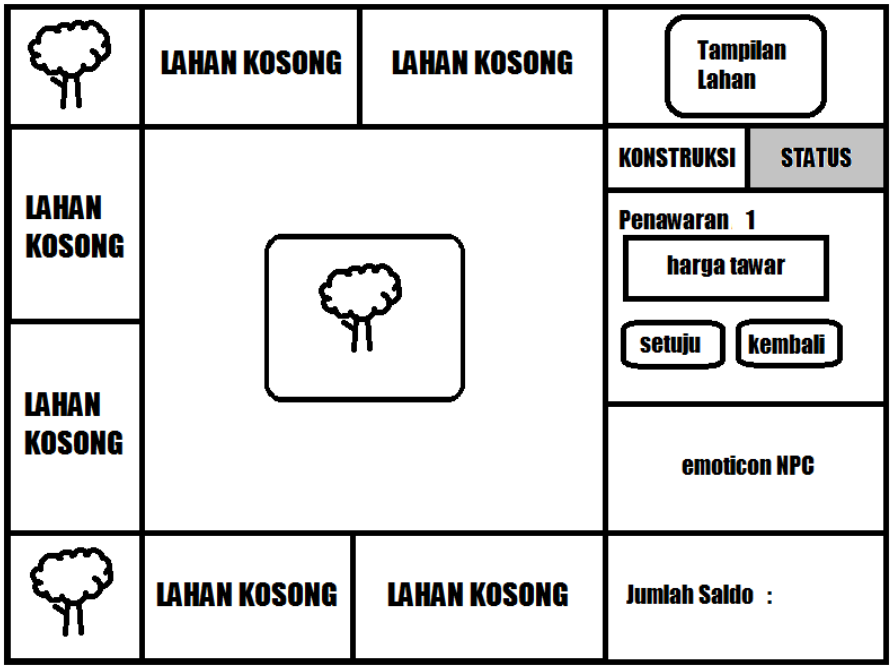
Gambar 3. 24 *Map* 1 dan *Panel* Konstruksi



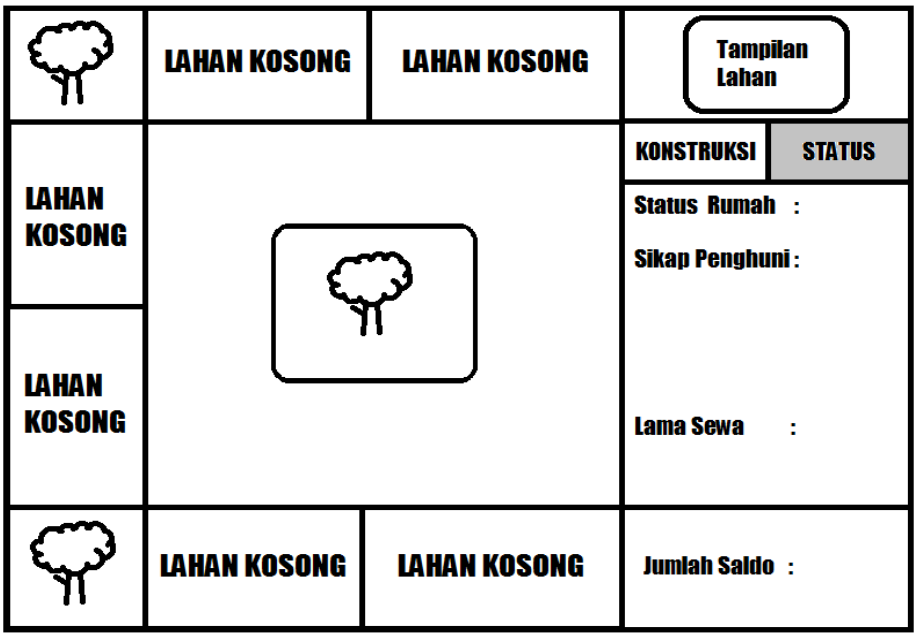
Gambar 3. 25 *Map* 1 dan *Panel* Status



Gambar 3. 26 *Map* 1 dan *Panel* Status Negosiasi 1



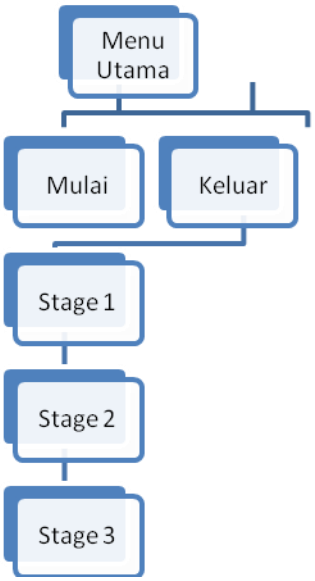
Gambar 3. 27 *Map* 1 dan *Panel* Status Negosiasi 2



Gambar 3. 28 *Map* 1 dan Status Rumah Terisi

3.8.1.3 Perancangan Struktur Menu

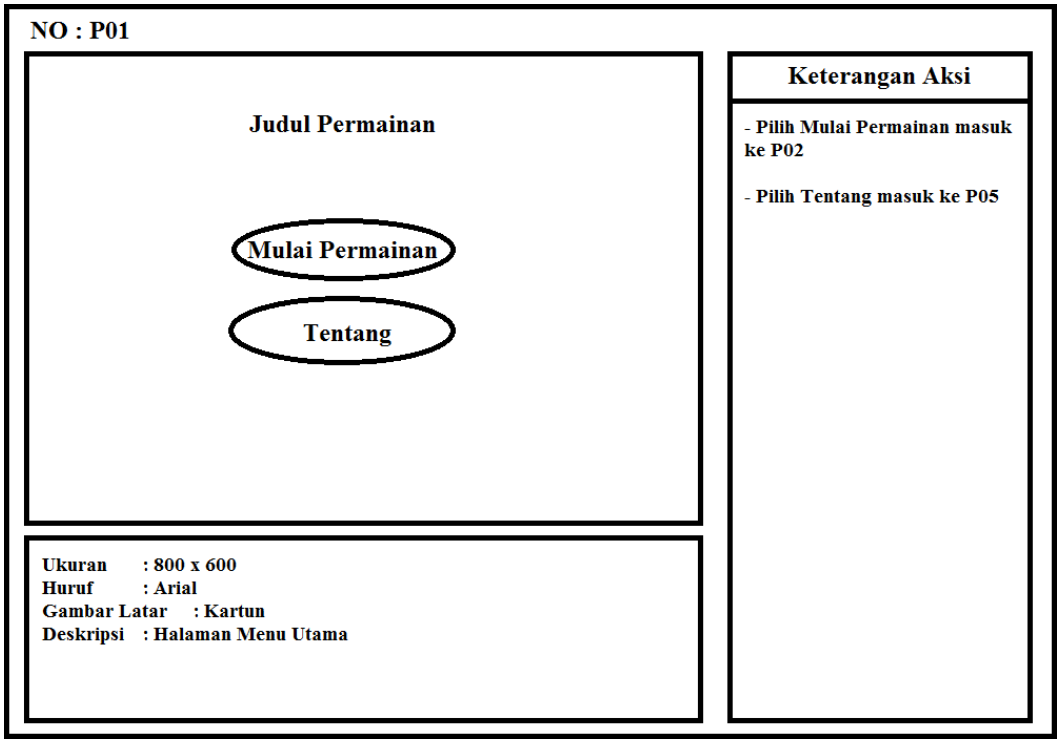
Dalam pembangunan aplikasi diperlukan sebuah struktur menu untuk menunjukan keterkaitan dan tingkatan antara menu utama dan sub menu, sehingga dapat lebih mudah di mengerti. Perancangan trusktur menu digambarkan seperti berikut.



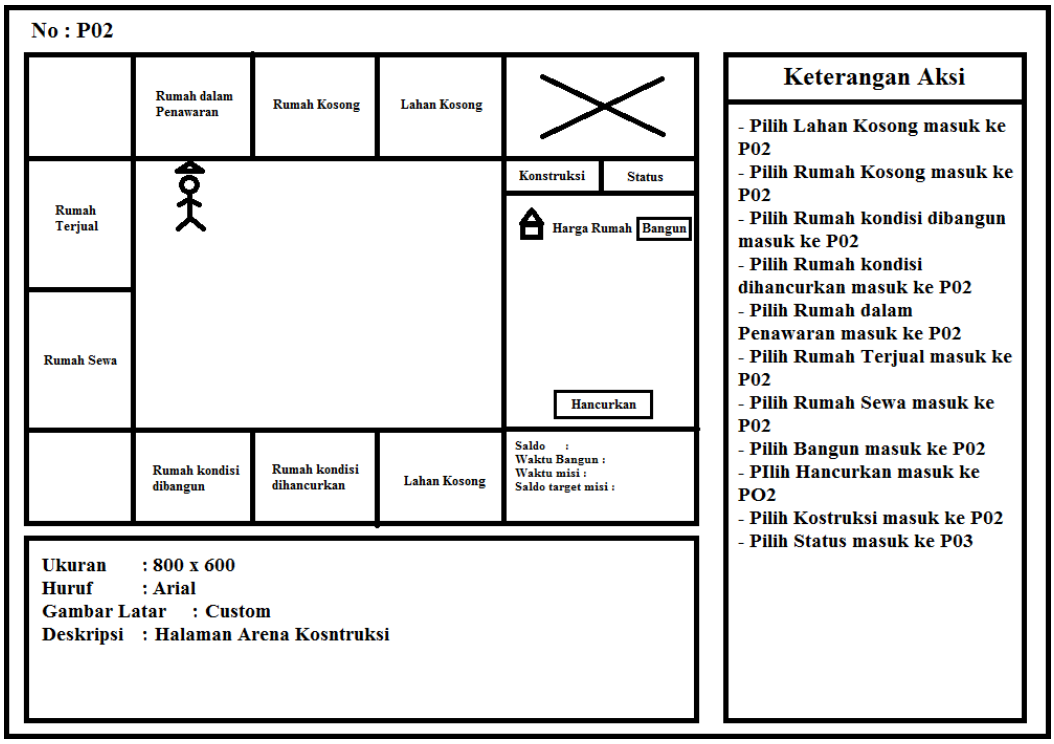
Gambar 3. 29 Struktur Menu

3.8.1.4 Perancangan Antar Muka

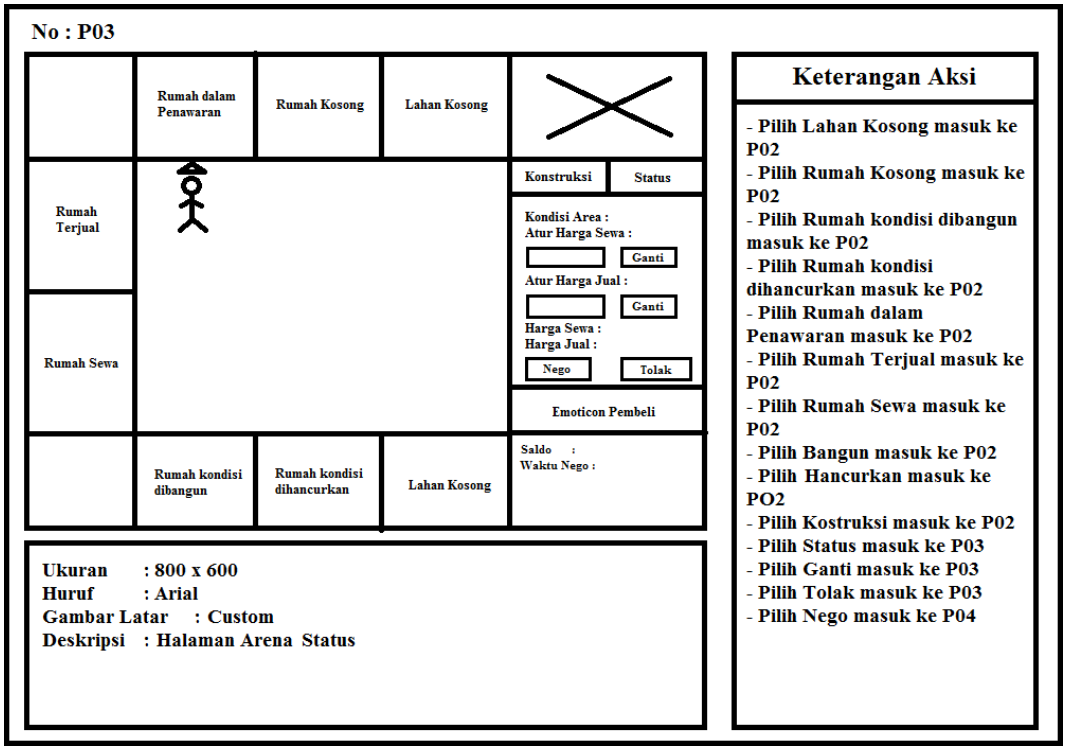
Perancangan antarmuka merupakan salah satu bagian penting dalam perancangan sistem karena nantinya antarmuka tersebut akan menjadi fasilitas yang menjembatani interaksi manusia dengan sistem. Perancangan antarmuka pada *Game* ini digambarkan pada Gambar 3.30 sampai dengan Gambar 3.34.



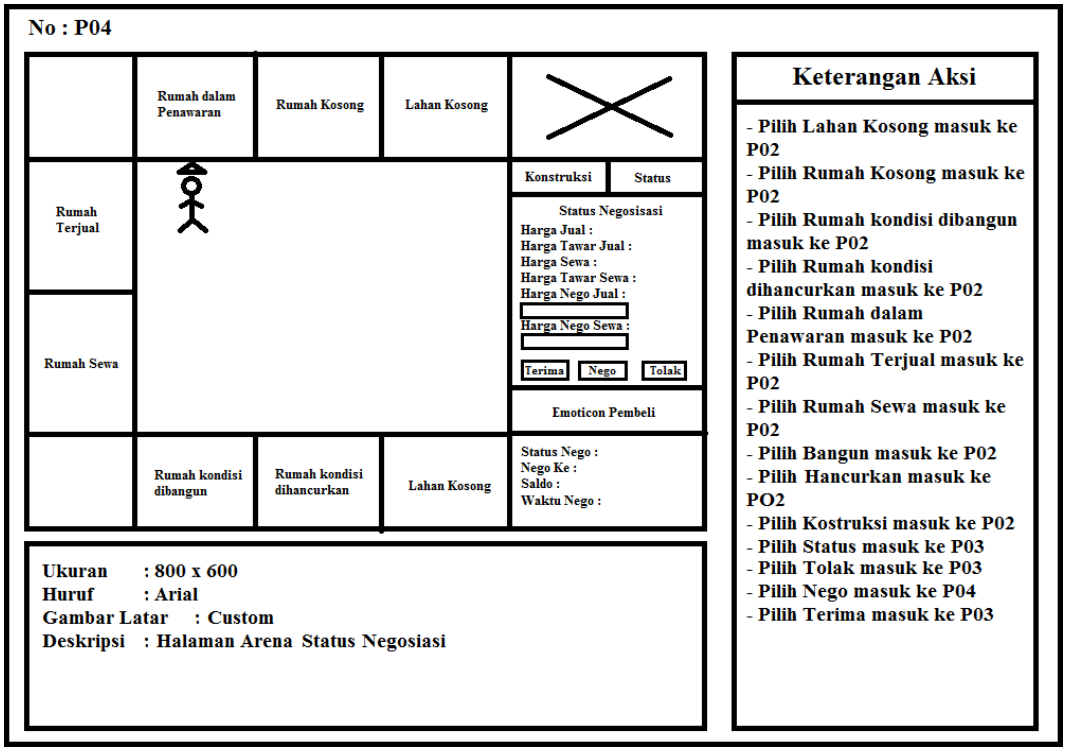
Gambar 3. 30 Perancangan Antarmuka Menu Utama



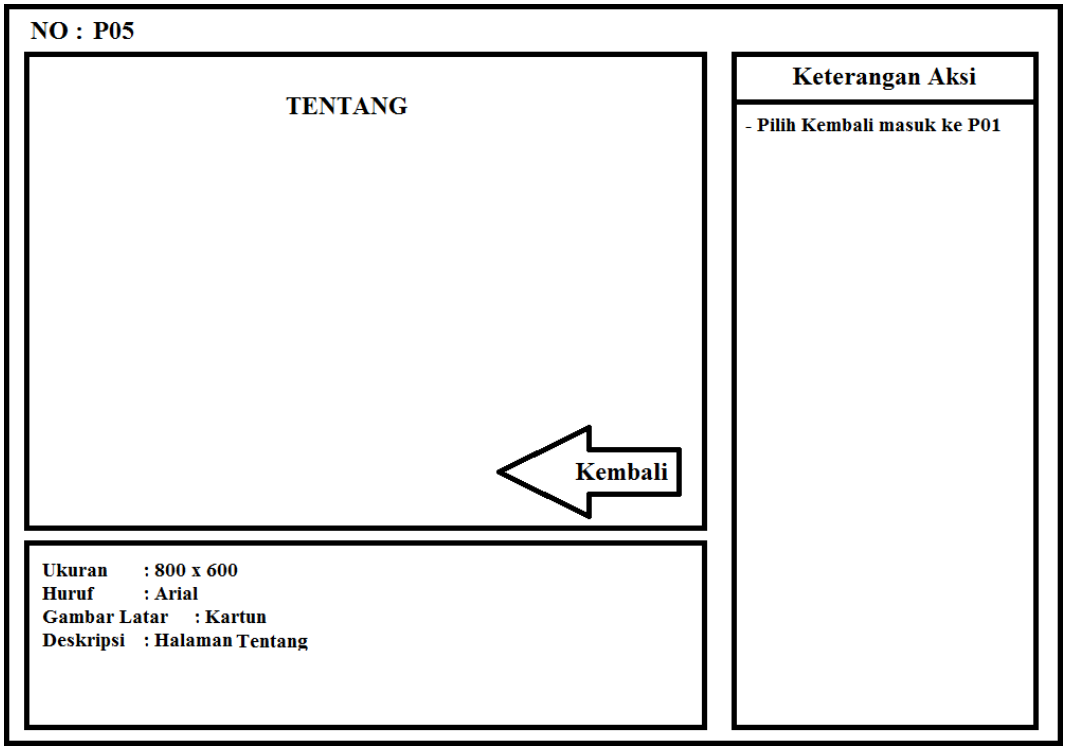
Gambar 3. 31 Perancangan Antarmuka Area dan *Panel* Konstruksi



Gambar 3. 32 Perancangan Antarmuka Area dan *Panel* Status



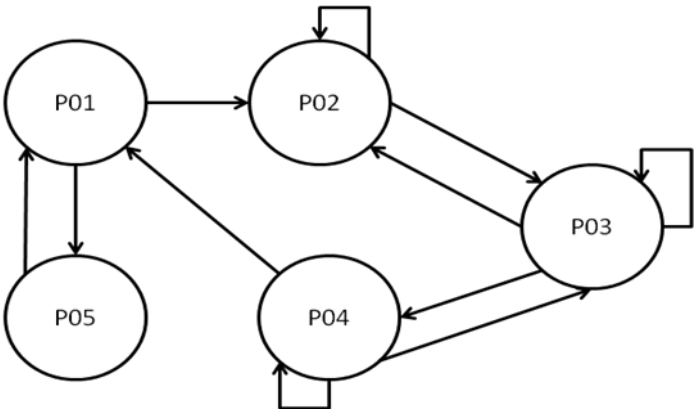
Gambar 3. 33 Perancangan Antarmuka Area dan *Panel* Negosiasi



Gambar 3. 34 Perancangan Antarmuka Tentang

3.8.1.5 Jaringan Semantik

Jaringan semantik merupakan alat yang mangkus untuk merepresentasikan pemetaan data, yang bertujuan mencegah terjadinya duplikasi data. Jaringan semantik pada *Game* ini digambarkan seperti berikut.



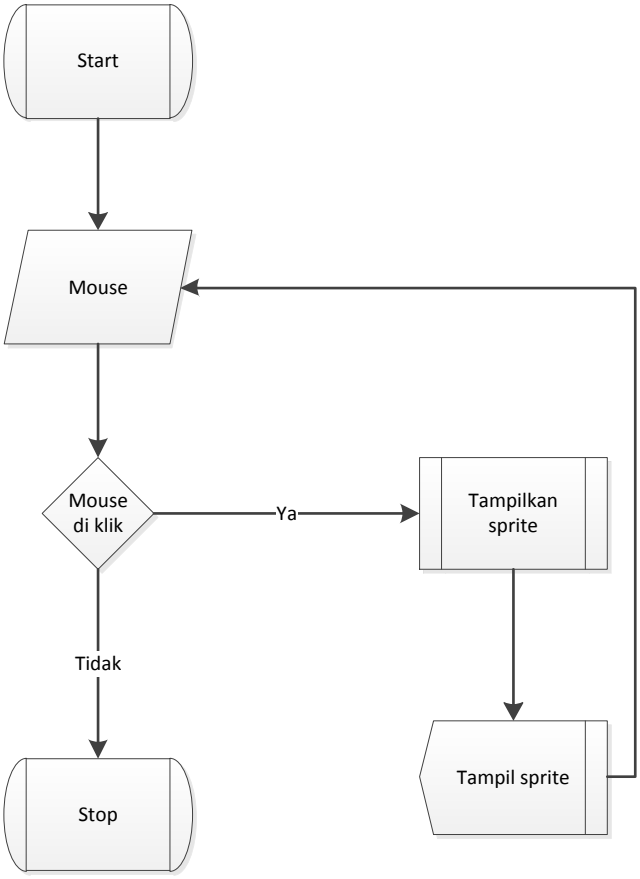
Gambar 3. 35 Jaringan Semantik

3.8.2 Perancangan *Method*

Perancangan *method* merupakan pemaparan proses pada *method-method* yang digunakan pada setiap kelas. Perancangan *method* digambarkan dengan menggunakan *Flow Chart*.

a. *Method* controls()

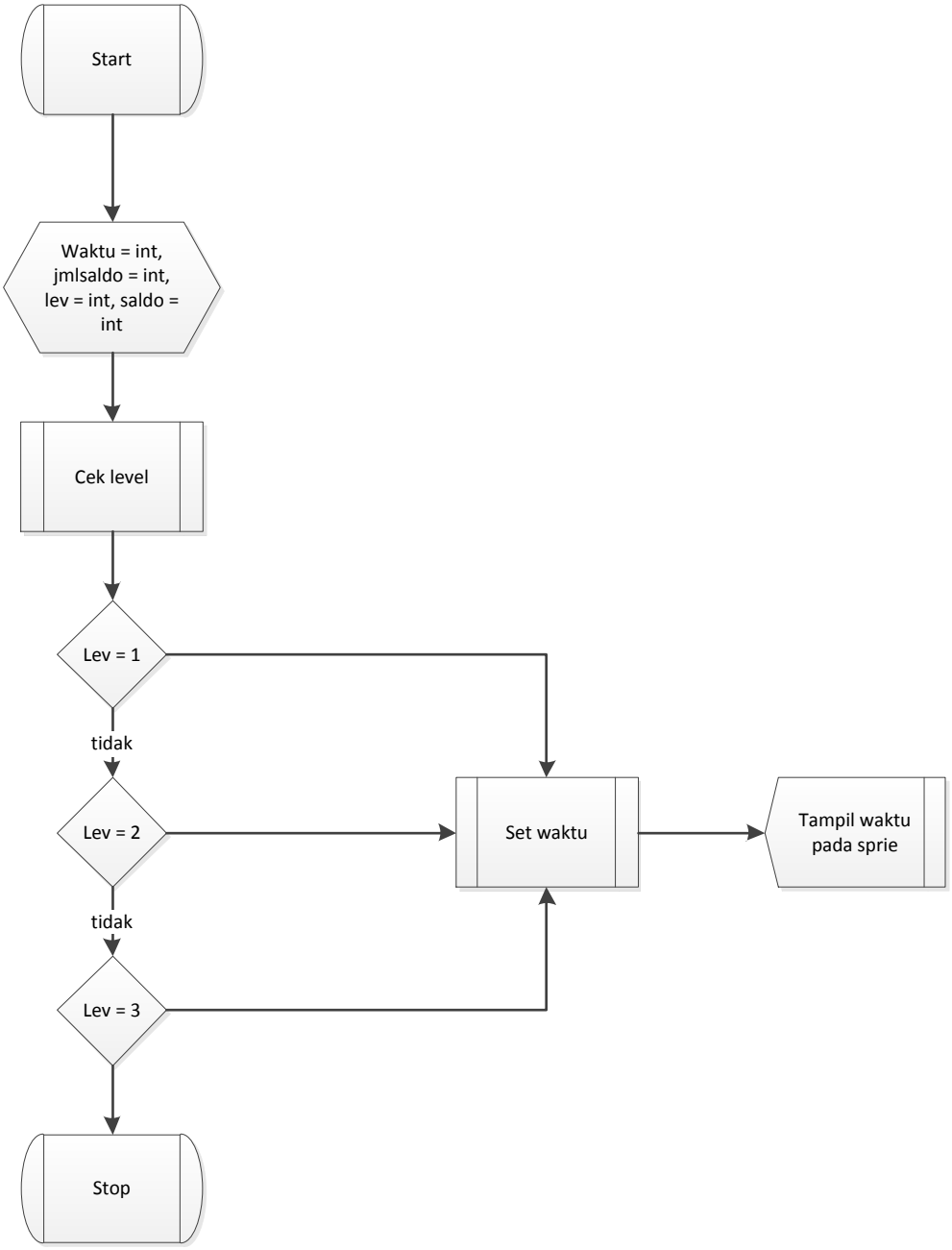
Method controls() dipanggil untuk mengoperasikan proses-proses yang ada di arena permainan, dapat dilihat pada gambar 3.36.



Gambar 3. 36 *Method Controls*()

b. *Method* time()

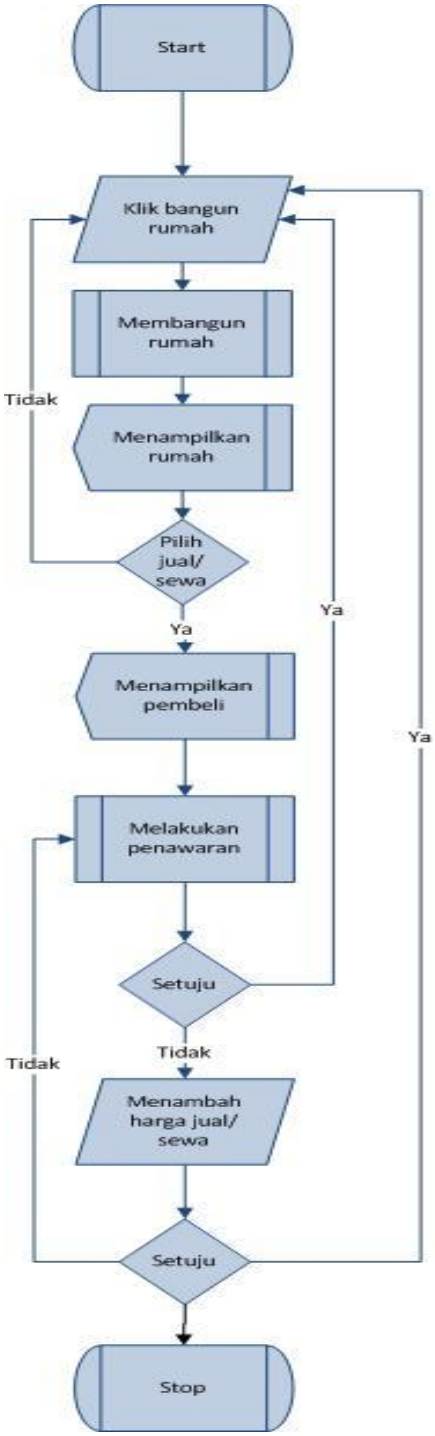
Method time() ini digunakan pada saat memulai misi pada awal permainan. *Method* time() dapat dilihat pada gambar 3.37.



Gambar 3. 37 *Method Time*()

c. *Method* Negosiasi()

Method Negosiasi() ini dipanggil ketika pemain sedang melakukan proses negosiasi dengan pembeli, *method* ini dapat dilihat pada gambar 3.38.



Gambar 3. 38 *Method* Negosiasi()