# BAB 3
# ANALISIS DAN PERANCANGAN

## 3.1 Analisis Sistem

Analisis sistem dapat didefinisikan sebagai penguraian dari suatu sistem yang utuh ke dalam bagian-bagian komponennya dengan maksud untuk mengidentifikasi dan mengevaluasi kebutuhan-kebutuhan yang diharapkan sehingga dapat diusulkan perbaikan-perbaikannya.

## 3.2 Analisis Masalah

Berdasarkan analisis masalah yang ada pada latar belakang sebelumnya terdapat beberapa permasalahan yang telah di analisis yaitu permasalahan penyalahgunaan narkoba dikalangan remaja usia 15-24 tahun. Data dari Bareskrim Polri & Badan Narkotika Nasional 2013 menyebutkan tersangka narkoba berdasarkan jenjang pendidikan, setingkat SMU menduduki peringkat paling tinggi yaitu 60,13 persen. Dengan teknologi *mobile* yang berkembang dikalangan remaja saat ini yang bertujuan untuk mempermudah dalam proses penyampaian informasi, maka dari itu dibangunlah *game Say No To Drugs* berbasis android sebagai media pengenalan tentang bahaya narkoba terutama untuk kalangan remaja.

### 3.2.1 Analisis *Game* Sejenis

Pada analisis *game* sejenis ini akan menjelaskan proses identifikasi mengenai *game* sejenis dan *game* yang akan dibangun. *Game* sejenis yang menjadi acuan yaitu *game adventure*.

*Game* dengan tipe *adventure* banyak dimainkan oleh orang-orang karena *game adventure* memberikan tantangan dan menarik untuk dimainkan. Banyak sekali *game* yang menggunakan tipe *adventure*, beberapa contoh diantaranya adalah “Muffin Knight”.

#### 3.2.1.1 Pengenalan Game Muffin Knight

*Game* ini merupakan salah satu *game action* yang menarik. Setiap sesi sangat pendek, jadi setiap level mampu dilewati dengan cepat. *Game* ini juga

didesain dengan *cross platform* sehingga bisa dimainkan dengan sesama pengguna Android, maupun dengan iOS dan pengguna *web browser Chrome*. Tampilan *game muffin knight* dapat dilihat pada Gambar 3.1 dan Gambar 3.2




**Gambar 3. 1 Tampilan Menu Utama Muffin Knight**



**Gambar 3. 2 Tampilan Level Muffin Knight**

### 3.2.1.2 Gameplay Muffin Knight

*Player* berperan sebagai karakter Muffin knight dengan menggunakan tombol-tombol yang telah digunakan yaitu tombol gerakan kiri dan kanan, lompat dan aksi, Muffin Knight bertugas untuk mengumpulkan muffin-muffin yang bertebaran disetiap level-nya. Dalam mengumpulkan muffin-muffin tersebut karakter dihalangi oleh musuh yang dapat membunuh karakter jika menyentuh karakter, selain itu setelah mengambil muffin karakter akan berubah menjadi bentuk lain yang mempunyai kemampuan yang berbeda-beda, ini menuntut player

untuk mencari jalan alternatif untuk mencapai muffin-muffin tersebut. Untuk melanjutkan ke level selanjutnya player harus mendapatkan sejumlah muffin yang telah ditentukan, selain itu muffin juga digunakan untuk menaikkan level karakter-karakter dalam Muffin Knight, guna dari menaikkan level karakter tersebut adalah untuk mendapatkan keahlian-keahlian khusus dari karakter sehingga memudahkan *player* dalam mengatasi permasalahan dalam *game*.

#### 3.2.1.3 Kelebihan dan Kekurangan Game Muffin Knight

Adapaun kelebihan dan kekurangan *game* muffin knight adalah sebagai berikut :

a. Kelebihan *Game* Muffin Knight
    1. Adanya fitur *unlockables*, yaitu fitur ini hanya dapat dibuka tahap per-tahap seiring perjalanan pada saat bermain .
    2. Karakter yang bisa berubah-ubah setelah mengambil muffin.
b. Kekurangan *Game* Muffin Knight
    1. Di setiap level pada *game* muffin knight cepat diselesaikan karena hanya mengambil muffin dan di setiap level hanya terdapat 1 atau 2 muffin saja.
    2. *Game* muffin knight di setiap level memiliki kriteria yang sama yaitu mengumpulkan muffin.

Tabel 3.1 menunjukkan perbandingan antara *game* Muffin Knight dengan *game* yang akan dibangun.

**Tabel 3. 1 Perbandingan dengan game yang akan dibangun**

| **Komponen** | **Game Muffin Knight** | **Game yang akan dibangun** |
|---|---|---|
| Genre | *Adventure* | *Adventure* |
| Grafis | 2 Dimensi | 2 Dimensi |
| Jumlah pemain | *Single player* dan *Multi player* | *Single player* |
| Aksi Karakter | Kanan, kiri, melompat, aksi, tembak | Kanan, kiri, melompat, aksi, tending |

| | | |
|---|---|---|
| Level | Terdiri dari banyak level | 3 Level |
| Gameplay | Mengumpulkan muffin | Menyelamatkan penderita narkoba dan menangkap pengedarnya. |

### 3.2.2 Analisis *Game* yang Akan Dibangun

*Game* yang akan dibangun adalah jenis game yang bergenre *adventure* dan bersifat *single player*, dengan karakter utama "ANO", disetiap level atau stage akan melawan sekumpulan musuh-musuh yang berupa monster yang berupa narkoba untuk dimusnahkan. Di setiap perjalanan petualangan dari masing-masing *level* atau *stage* dan diakhir dari masing-masing *level* atau *stage* akan di tampilkan pesan berupa bahaya dan dampak negatif dari narkoba. Pemain diharuskan untuk mengambil herbal atau obat untuk diberikan kepada penderita/pecandu narkoba, serta menangkap pengedar dan memusnahkan pengedar besar yaitu raja Narkoba di stage akhir permainan. Di setiap level akan di berikan hati untuk menambah darah karakter utama. Jika misi berhasil, itu artinya ANO berhasil membebaskan orang-orang dari jeratan Narkoba dan akan menampilkan pesan *"Say no To Drugs".*

*Game* ini memiliki beberapa fitur diantaranya:

1. Bergrafis 2D dan bersifat *offline*.
2. Sistem *single player*.
3. Karakter utama "ANO".
4. Memiliki 3 level dengan tingkat kesulitan dan musuh yang berbeda disetiap levelnya.
5. Mengangkat tema tentang bahaya Narkoba.
6. *Control* karakter / sistem kendali menggunakan *touchscreen.*

#### 3.2.2.1 Story Line

*Game* ini menceritakan seorang anak muda yang bernama "ANO" dalam menyelamatkan penderita narkoba dan menangkap pengedar serta bos besar narkoba. Tujuan dan misi dari *game* ini adalah menyelamatkan pecandu narkoba

dan menangkap pengedar narkoba sebagai bentuk simbol bebas dari jeratan "Narkoba" dan akan menampilkan pesan "*Say No To Drugs*".

#### 3.2.2.2 Gameplay

Berikut ini adalah beberapa bagian *game* play dari *game say no to drugs* :

1. *Level 1*

Sebelum memulai level 1, akan ada sedikit intro mengenai apa itu narkoba, serta dampaknya bagi kehidupan generasi muda. Pada level 1, karakter utama ditugaskan untuk menyelamatkan orang yang terkena narkoba dengan cara mengambil atau mencari herbal/obat terlebih dahulu untuk diberikan kepada penderita / pecandu narkoba. Musuh yang harus dilawan pada *level* 1 ini adalah musuh daun ganja. Karakter utama tidak memiliki senjata khusus untuk melawan musuh, karakter utama menggunakan tendangan untuk mengalahkan musuh-musuh tersebut. Setelah menang maka karakter utama akan melanjutkan misi ke *level* selanjutnya dengan tujuan menangkap pengedar narkoba.

2. *Level 2*

Petualangan dilanjutkan ke *level* 2 dengan misi yaitu menangkap beberapa orang pengedar narkoba yang sudah meracuni generasi muda saat ini. Musuh yang meghadang di level 2 ini berbeda dengan level 1, jika level 1 musuh nya berupa daun ganja, maka di level 2 ini musuh yang dihadapi adalah virus HIV yang merupakan salah satu penyakit yang ditimbulkan akibat jarum suntik yang disalahgunakan oleh para pengguna narkoba. Ada beberapa orang pengedar yang harus ditangkap di *level* 2 ini, jika semua sudah tertangkap itu berarti misi di *level* 2 ini berhasil dan karakter utama akan melanjutkan ke *level* 3.

3. Level 3

*Level* 3 merupakan gabungan dari *level* 1 dan 2, pada *level* 3 ini karakter utama ditugaskan untuk menyelamatkan penderita / pecandu serta menangkap beberapa pengedar narkoba. Dalam area ini terdapat lagi seorang musuh yang berupa monster jarum suntik dan yang terakhir Raja Narkoba. Raja pengedar narkoba akan dilawan saat misi menyelamatkan pecandu dan menangkap

pengedar narkoba sudah dilakukan. Dengan menerapkan algoritma A* (star), raja pengedar narkoba itu akan terus mengikuti atau mendekati karakter utama. Di akhir *level* akan ditampilkan pesan tentang jenis-jenis narkoba yang harus dihindari dan pesan "*Say No To Drugs*".

Dalam setiap *level* atau *stage* terdapat hati yang harus diambil untuk menambah darah, jika darah karakter utama sudah berkurang. Hal ini agar membantu pemain dalam bermain serta membantu karakter utama "ANO" dalam menyelesaikan misi-misi nya.

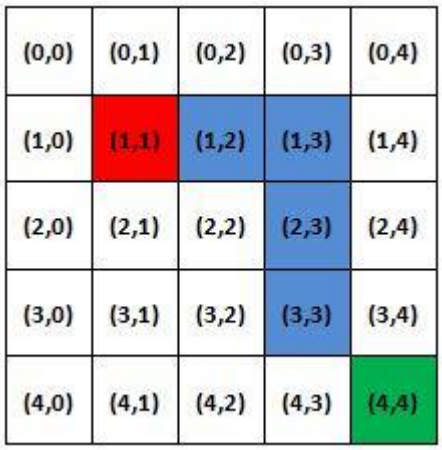
#### 3.2.2.3 Scoring

Pada bagian ini akan dijelaskan bagaimana pemberian skor yang ada pada *game*. Berikut cara memperoleh nilai atau *score* pada *game "Say No To Drugs"* yaitu :

1. Jika berhasil mengalahkan musuh, maka memperoleh *score* 10.
2. Jika berhasil menyelesaikan *level* 1 akan mendapatkan *score* 70.
3. Jika berhasil menyelesaikan *level* 2 akan mendapatkan *score* 160.
4. Jika berhasil menyelesaikan *level* 3 akan mendapatkan *score* 240..
5. Jika berhasil mengalahkan raja akan mendapatkan *score* 500.
6. Skor tertinggi dari *level* 1 sampai dengan melawan raja adalah 740.

### 3.2.3 Analisis Algoritma

Dalam *game* ini akan digunakan algoritma A* pada NPC (*Non Playable Character*) musuh sebagai penentuan jalur terpendek atau *pathfinding* menuju target yaitu player. Untuk lebih jelasnya pada Gambar 3.3 diberikan posisi awal dimana objek musuh di posisi (1,1) dan player ada di posisi (4,4).




**Gambar 3. 3 Posisi Awal Algoritma A***

Pada Gambar 3.3 pencarian wilayah dibagi ke dalam node-node yang terdiri dari node berwarna merah merupakan node awal, node berwarna hijau merupakan node akhir dan yang berwarna biru merupakan penghalang (*unwalkable node*). Dengan menggunakan fungsi heuristic Non-Manhattan Distance, dengan fungsi heuristic yang digunakan adalah sebagai berikut:

h_diagonal(n) = min(abs(n.x-goal.x) + abs(n.y-goal.y))

h_orthogonal(n)= (abs(n.x-goal.x) + abs(n.y-goal.y))

h(n) = h_diagonal(n) + (h_orthogonal (n) – (2 * h_diagonal(n))).

Langkah perhitungannya adalah sebagai berikut:

Koordinat n(0,0)

Nilai n.x = 0

Nilai n.y = 0

Simpul tujuan(4,4) sehingga goal.x = 4, goal.y = 4

g (0,0) = 1,4

h_orthogonal(n) = (abs(n.x-goal.x) + abs(n.y-goal.y))

h_orthogonal(0,0) = (abs(0 - 4) + abs(0 - 4))

= (abs(-4) + abs(-4))

= 8

h_diagonal(n) = -(abs(n.x-goal.x) + abs(n.y-goal.y))

h_diagonal(0,0) = -(abs(0 - 4)+abs(0 - 4))

= -(abs(-4)+abs(-4))

= -8

h(n) = h_diagonal(n) + (h_orthogonal (n) – (2 * h_diagonal(n)))

h(0,0) = (-8) + (8-(2*(-8)))

= (-8) + 24

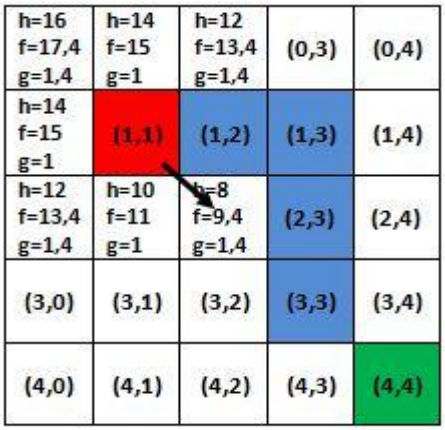
= 16

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

= 1,4 + 16

= 17,4

Hasil perhitungan dari langkah pertama dapat dilihat pada Gambar 3.4 merupakan penjelasan langkah pertama pencarian jalur dari algoritma A*.



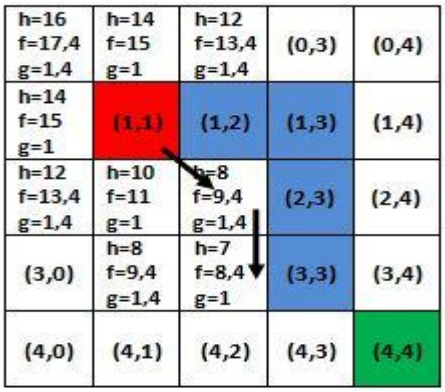

**Gambar 3. 4 Penjelasan A* Pada Langkah Pertama**

Langkah pertama menghasilkan node-node di dalam *open list* dan *closed list* yang dapat dilihat pada Tabel. 3.2.

**Tabel 3. 2 Perhitungan Langkah Pertama**

| | |
|---|---|
| *Open list* | (2,0), (2,1),(2,2) |
| *Closed list* | (1,1) |
| *Parent* | (1,1) |
| *Current node* | (2,2) |

Hasil perhitungan dari langkah kedua dapat dilihat pada Gambar 3.5 merupakan penjelasan langkah kedua pencarian jalur dari algoritma A*.



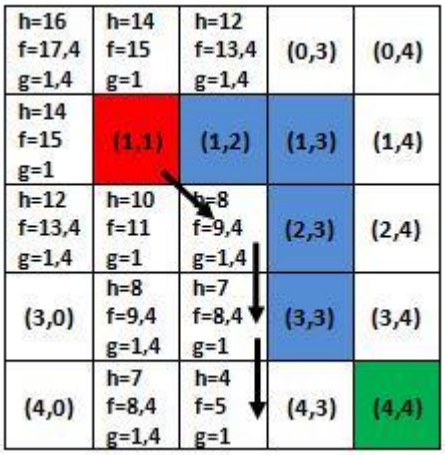

**Gambar 3. 5 Penjelasan A* Pada Langkah Kedua**

Langkah kedua menghasilkan node-node di dalam *open list* dan *closed list* yang dapat dilihat pada Tabel. 3.3

**Tabel 3. 3 Perhitungan Langkah Kedua**

| | |
|---|---|
| *Open list* | (2,1),(3,1),(3,2) |
| *Closed list* | (2,2),(2,3),(3,3) |
| *Parent* | (2,2) |
| *Current node* | (3,2) |

Hasil perhitungan dari langkah ketiga dapat dilihat pada Gambar 3.6 merupakan penjelasan langkah ketiga pencarian jalur dari algoritma A*.



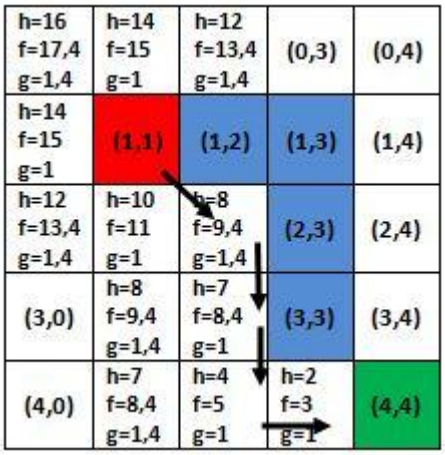

**Gambar 3. 6 Penjelasan A* Pada Langkah Ketiga**

Langkah ketiga menghasilkan node-node di dalam *open list* dan *closed list* yang dapat dilihat pada Tabel. 3.4

**Tabel 3. 4 Perhitungan Langkah Ketiga**

| | |
|---|---|
| *Open list* | (3,1),(4,1),(4,2) |
| *Closed list* | (3,3),(4,3) |
| *Parent* | (3,2) |
| *Current node* | (4,2) |

Hasil perhitungan dari langkah keempat dapat dilihat pada Gambar 3.7 merupakan penjelasan langkah keempat pencarian jalur dari algoritma A*.



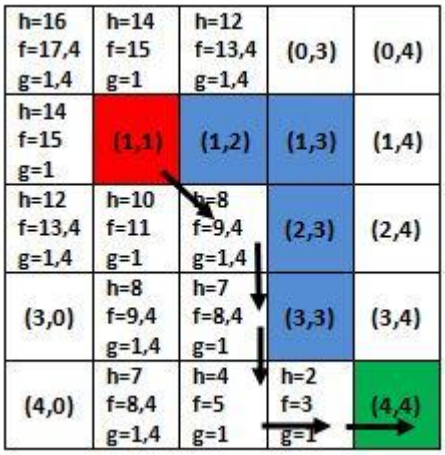

**Gambar 3. 7 Penjelasan A* Pada Langkah Keempat**

Langkah keempat menghasilkan node-node di dalam *open list* dan *closed list* yang dapat dilihat pada Tabel. 3.5

**Tabel 3. 5 Perhitungan Langkah Keempat**

| | |
|---|---|
| *Open list* | (4,1),(4,2) |
| *Closed list* | (4,3), |
| *Parent* | (4,2) |
| *Current node* | (4,3) |

Hasil perhitungan dari langkah kelima dapat dilihat pada Gambar 3.8 merupakan penjelasan langkah kelima pencarian jalur dari algoritma A*.




**Gambar 3. 8 Penjelasan A* Pada Langkah Kelima**

Langkah kelima menghasilkan node-node di dalam *open list* dan *closed list* yang dapat dilihat pada Tabel. 3.6

**Tabel 3. 6 Perhitungan Langkah Kelima**

| | |
|---|---|
| *Open list* | (3,3),(4,4) |
| *Closed list* | (4,3), |
| *Parent* | (4,3) |
| *Current node* | (4,4) |

### 3.2.4 Analisis Kebutuhan Non Fungsional

Analisis kebutuhan fungsional menggambarkan proses kegiatan yang akan diterapkan dalam sebuah sistem dan menjelaskan analisis kebutuhan perangkat lunak, analisis dan kebutuhan perangkat keras, serta analisis dan kebutuhan user.

#### 3.2.4.1 Analisis dan Kebutuhan Perangkat Lunak

Perangkat lunak diguunakan dalam sebuah sistem merupakan perintah-perintah yang diberikan kepada perangkat keras agar bisa dapat berinteraksi diantara keduanya. Adapun perangkat lunak yang dibutuhkan dalam membangun aplikasi *game* ini adalah sebagai berikut :

**1. Dari sisi pengembang**

Perangkat lunak yang digunakan untuk mengembangkan *game* ini adalah sebagai berikut:

1. Sistem Operasi Windows
2. *Adobe Flash Professional CS6*
3. *Adobe Air*

**2. Dari sisi pemain**

*Game adventure* "*Say No To Drugs*" ini dibutuhkan perangkat lunak untuk menjalankannya, yaitu Sistem Operasi Android 2.2.

### 3.2.4.2 Analisis dan Kebutuhan Perangkat Keras

Suatu komputer terdiri dari perangkat keras dan perangkat lunak yang saling berinteraksi dan terintregrasi. Perangkat lunak memberikan instruksi-instruksi kepada perangkat keras untuk melakukan suatu tugas tertentu, sehingga dapat menjalankan suatu sistem di dalamnya. Agar aplikasi dapat berjalan dengan baik, maka dibutuhkan perangkat keras yang sesuai dengan kebutuhan aplikasi.

Di bagian ini akan dijelaskan analisis kebutuhan perangkat keras dari sisi pengembang dan dari sisi pemain. Pada aplikasi *game* "*Say No To Drugs*" ini, perangkat keras yang digunakan adalah sebagai berikut :

**1. Dari sisi pengembang**

Perangkat keras yang digunakan untuk membangun *game* ini adalah sebagai berikut:

1. Prosesor 2.1 GHz
2. Memori RAM DDR2 2 GB
3. VGA card 512 MB
4. Harddisk 160 GB
5. Mouse
6. Keyboard
7. Display LCD Monitor dengan resolusi 1366 x 768

2. **Dari sisi pemain**

Spesifikasi minimum perangkat keras yang dibutuhkan agar dapat menjalankan *game* ini adalah sebagai berikut:

1. *Smartphone* Android 2.2
2. Layar *touchscreen.*

#### 3.2.4.3 Analisis Pengguna

Selain dibutuhkannya perangkat lunak dan perangkat keras, user juga sangat dibutuhkan dalam penggunakaan aplikasi *game* ini. Adapun spesifikasi user yang dibutuhkan :

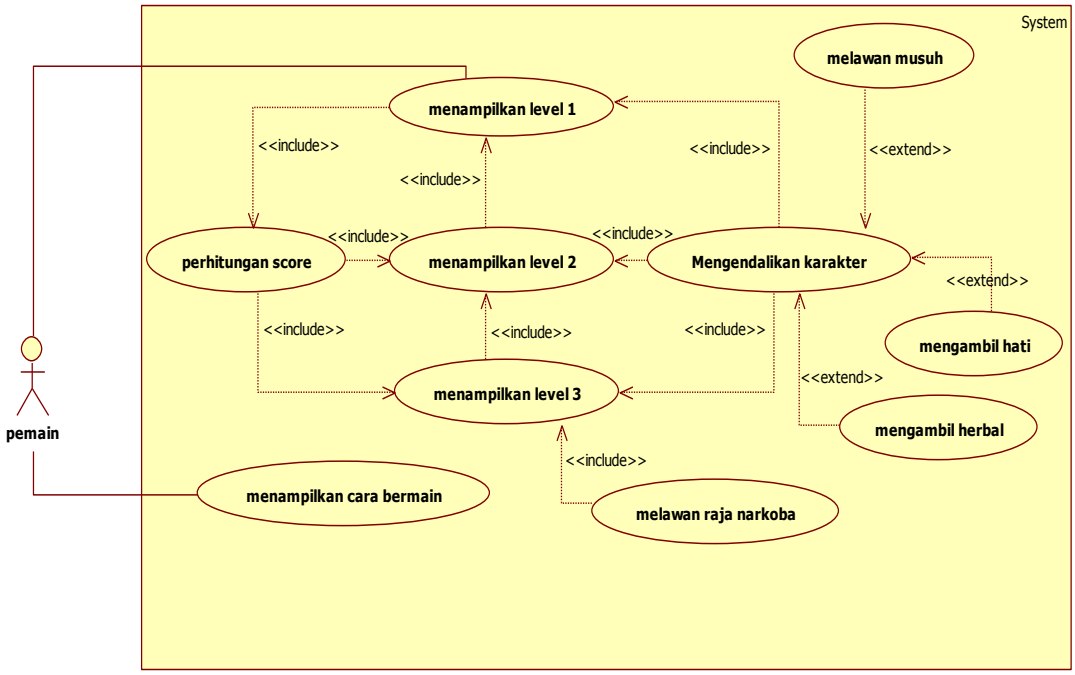
1. Memiliki pengalaman dalam mengoperasikan *game mobile phone*
2. Mengetahui cara menggunakan *mobile phone*
3. Usia 15-24 tahun ( Laki-laki dan Perempuan )
4. Telah mampu membaca
5. Tidak buta warna

### 3.2.5 Analisis Kebutuhan Fungsional

Analisis kebutuhan fungsional menggambarkan proses kegiatan yang akan diterapkan dalam sebuah sistem dan menjelaskan kebutuhan yang diperlukan sistem agar sistem dapat berjalan dengan baik sesuai kebutuhan. Analisis yang dilakukan dimodelkan dengan menggunakan UML *(Unified Modeling Language).* Pada bagian ini akan dibahas tentang *use case diagram, activity diagram, class diagram, dan sequence diagram.*

#### 3.2.5.1 *Use case diagram*

Pada sub bab berikut ini dijelaskan bagaimana *Use Case* yang terjadi dalam pembangunan *game* ini yang dimodelkan dalam sekumpulan *Use Case* dan *Actor* dan bagaimana hubungan-hubungannya yang akan dipetakan dalam diagram *Use Case*, dilengkapi dengan skenario untuk menjelaskan dari gambaran *Use Case* yang ada dengan menggambarkan skenario per *Use Case*. *Use case diagram* dapat dilihat pada Gambar 3.9 berikut ini.




**Gambar 3. 9** ***Use Case Diagram Game Adventure "Say No To Drugs"***

### 3.2.5.2 *Actor Definiton*

*Actor Definition* berfungsi untuk menjelaskan *Actor* yang terdapat pada *Use case diagram*. *Actor Definition* diterangkan pada Tabel 3.7.

**Tabel 3. 7** ***Actor Definition***

| No | *Actor* | Deskripsi |
|---|---|---|
| 1 | Pemain | Orang yang memainkan *game* |

### 3.2.5.3 *Use Case Definition*

*Use Case Definition* berfungsi untuk menjelaskan fungsi *Use Case* yang terdapat pada *Use case diagram. Use Case Definition* diterangkan pada Tabel 3.8.

**Tabel 3. 8** ***Use Case Definition***

| No | *Use Case* | Deskripsi |
|---|---|---|
| 1 | Menampilkan *level* 1 | Proses untuk memulai permainan dan menampilkan *level* 1 |

| | | |
|---|---|---|
| 2 | Menampilkan *level* 2 | Proses untuk menampilkan *level* 2 |
| 3 | Menampilkan *level* 3 | Proses untuk menampilkan *level* 3 |
| 4 | Menampilkan Cara bermain | Proses untuk menampilkan cara bermain |
| 5 | Mengambil herbal | Proses untuk mengambil herbal |
| 6 | Mengambil hati | Proses untuk mengambil hati |
| 7 | Melawan musuh | Proses untuk melawan musuh |
| 8 | Melawan raja | Proses untuk melawan raja |
| 9 | Mengendalikan karakter | Proses yang dilakukan oleh pemain untuk mengendalikan karakter utama |
| 10 | Perhitungan Skor | Proses untuk menghitung skor |

#### 3.2.5.4 *Use Case Scenario*

*Use Case Scenario* mendeskripsikan urutan langkah-langkah dalam proses bisnis baik yang dilakukan aktor terhadap sistem maupun yang dilikakukan oleh sistem terhadap aktor. Berdasarkan *use case diagram* pada Gambar 3.9 diatas maka *use case scenario* untuk aplikasi yang dibangun akan dijelaskan sebagai berikut.

***a. Use Case Scenario* Menampilkan *Level* 1**

*Use Case Scenario* dari *Use Case* Menampilkan *level* 1 dapat dilihat pada Tabel 3.9.

**Tabel 3. 9 *Use Case Scenario* Menampilkan *Level* 1**

| **Identifikasi** | |
|---|---|
| **Nomor** | 1 |
| **Nama** | Menampilkan *Level* 1 |
| **Tujuan** | Proses yang dilakukan pemain untuk memulai |

| | |
|---|---|
| | permainan dan menampilkan *level* 1 |
| **Aktor** | Pemain |
| **Skenario Utama** | |
| **Kondisi Awal** | Pemain berada di menu utama |
| **Aksi Aktor** | **Reaksi Sistem** |
| 1. Memilih tombol *Play* pada menu utama | |
| | 2. Sistem menampilkan prolog |
| 3. Memilih tombol *Play* yang ada pada prolog | |
| | 4. Sistem menampilkan arena permainan level 1 |
| **Kondisi Akhir** | Pemain berada di arena permainan level 1 |

### *b. Use Case Scenario* Menampilkan *Level* 2

*Use Case Scenario* dari *Use Case* Menampilkan *Level* 2 dapat dilihat pada Tabel 3.10.

**Tabel 3. 10 *Use Case Scenario* Menampilkan *Level* 2**

| **Identifikasi** | |
|---|---|
| **Nomor** | 2 |
| **Nama** | Menampilkan Level 2 |
| **Tujuan** | Menampilkan permainan pada level 2 |
| **Aktor** | Pemain |
| **Skenario Utama** | |
| **Kondisi Awal** | Sistem menampilkan prolog |

| **Aksi Aktor** | **Reaksi Sistem** |
|---|---|
| 1. Memilih tombol *play* pada prolog | |
| | 2. Sistem menampilkan arena permainan level 2 |
| **Kondisi Akhir** | Pemain berada di arena permainan level 2. |

### *c. Use Case Scenario* Menampilkan *Level* 3

*Use Case Scenario* dari *Use Case* Menampilkan *level* 3 dapat dilihat pada Tabel 3.11.

**Tabel 3. 11 *Use Case Scenario* Menampilkan *Level* 3**

| **Identifikasi** | |
|---|---|
| **Nomor** | 3 |
| **Nama** | Menampilkan Level 3 |
| **Tujuan** | Menampilkan permainan pada level 3 |
| **Aktor** | Pemain |
| **Skenario Utama** | |
| **Kondisi Awal** | Sistem menampilkan prolog |
| **Aksi Aktor** | **Reaksi Sistem** |
| 1. Memilih tombol *play* pada prolog | |
| | 2. Sistem menampilkan arena permainan level 3 |
| **Kondisi Akhir** | Pemain berada di arena permainan level 3 |

### *d. Use Case Scenario* Cara Bermain

*Use Case Scenario* dari *Use Case* Cara Bermain dapat dilihat pada Tabel 3.12.

**Tabel 3. 12 *Use Case Scenario* Cara Bermain**

| **Identifikasi** | |
|---|---|
| **Nomor** | 4 |
| **Nama** | Cara Bermain |
| **Tujuan** | Menampilkan cara bermain |
| **Aktor** | Pemain |
| **Skenario Utama** | |
| **Kondisi Awal** | Berada di cara bermain |
| **Aksi Aktor** | **Reaksi Sistem** |
| 1. Pemain memilih menu cara bermain | |
| | 2. Sistem akan merespon dan menampilkan menu cara bermain |
| **Kondisi Akhir** | Sistem menampilkan cara bermain |

### *e. Use Case Scenario* Mengambil Herbal

*Use Case Scenario* dari *Use Case* Mengambil Herbal dapat dilihat pada Tabel 3.13.

**Tabel 3. 13 *Use Case Scenario* Mengambil Herbal**

| **Identifikasi** | |
|---|---|
| **Nomor** | 5 |
| **Nama** | Mengambil Herbal |

| | |
|---|---|
| **Tujuan** | Mengambil Herbal |
| **Aktor** | Pemain |
| **Skenario Utama** | |
| **Kondisi Awal** | Karakter Bergerak |
| **Aksi Aktor** | **Reaksi Sistem** |
| 1. Karakter berjalan mendekati herbal | |
| | 2. Sistem akan merespon dan menambahkan jumlah herbal |
| | 3. Herbal menghilang |
| 4. Karakter bergerak mendekati penderita | |
| | 5. Herbal diberikan kepada penderita |
| | 6. Sistem akan merespon dan penderita sembuh dari narkoba |
| **Kondisi Akhir** | Penderita bebas dari narkoba |

### *f. Use Case Scenario* Mengambil Hati

*Use Case Scenario* dari *Use Case* Mengambil Hati dapat dilihat Tabel 3.14.

**Tabel 3. 14 *Use Case Scenario* Mengambil Hati**

| **Identifikasi** | |
|---|---|
| **Nomor** | 6 |
| **Nama** | Mengambil Hati |
| **Tujuan** | Menambah Darah |
| **Aktor** | Pemain |

| Skenario Utama | |
|---|---|
| **Kondisi Awal** | Darah Berkurang |
| **Aksi Aktor** | **Reaksi Sistem** |
| 1. Karakter utama berjalan mendekati dan mengambil hati | |
| | 2. Sistem akan merespon dan darah karakter utama akan bertambah |
| | 3. Hati akan menghilang |
| **Kondisi Akhir** | Darah karakter utama bertambah |

### *g. Use Case Scenario* Melawan Musuh

*Use Case Scenario* dari *Use Case* Melawan Musuh dapat dilihat pada Tabel 3.15.

**Tabel 3. 15 *Use Case Scenario* Melawan Musuh**

| Identifikasi | |
|---|---|
| **Nomor** | 7 |
| **Nama** | Melawan Musuh |
| **Tujuan** | Mengalahkan Musuh |
| **Aktor** | Pemain |
| **Skenario Utama** | |
| **Kondisi Awal** | Karakter Bergerak |
| **Aksi Aktor** | **Reaksi Sistem** |
| 1. Karakter berjalan mendekati musuh | |

| | |
|---|---|
| | 2. Sistem akan merespon dan musuh bergerak |
| 3. Karakter menyerang musuh | |
| | 4. Sistem akan merespon dan musuh akan menghilang |
| | 5. Jika karakter utama terkena musuh, darah akan berkurang. |
| **Kondisi Akhir** | Mengalahkan musuh |

### *h. Use Case Scenario* Melawan Raja

*Use Case Scenario* dari *Use Case* Melawan Raja dapat dilihat pada Tabel 3.16.

**Tabel 3. 16 *Use Case Scenario* Melawan Raja**

| **Identifikasi** | |
|---|---|
| **Nomor** | 8 |
| **Nama** | Melawan Raja |
| **Tujuan** | Mengalahkan Raja |
| **Aktor** | Pemain |
| **Skenario Utama** | |
| **Kondisi Awal** | Karakter bergerak |
| **Aksi Aktor** | **Reaksi Sistem** |
| 1. Pemain berjalan mendekati raja | |
| | 2. Sistem akan merespon dan raja bergerak mendekati karakter |
| 3. Karakter memukul raja | |

| | |
|---|---|
| | 4. Sistem akan merespon dan darah raja akan berkurang |
| | 5. Jika karakter terkena pukulan darah berkurang |
| **Kondisi Akhir** | Mengalahkan Raja |

### *i. Use Case Scenario* Mengendalikan Karakter

*Use Case Scenario* dari *Use Case* Mengendalikan Karakter dapat dilihat pada Tabel 3.17.

**Tabel 3. 17 *Use Case Scenario* Mengendalikan Karakter**

| **Identifikasi** | |
|---|---|
| **Nomor** | 9 |
| **Nama** | Mengendalikan Karakter Pemain |
| **Tujuan** | Proses yang dilakukan pemain untuk mengendalikan karakter pemain diarena permainan |
| **Aktor** | Pemain |
| **Skenario Utama** | |
| **Kondisi Awal** | Pemain berada di arena permainan |
| **Aksi Aktor** | **Reaksi Sistem** |
| 1. Pemain menekan tombol arah kanan  | |
| | 2. Sistem akan merespon dan karakter utama berjalan maju / kanan |

| | |
|---|---|
| 3. Pemain menekan tombol arah kiri  | |
| | 4. Sistem akan merespon dan karakter utama berjalan ke arah kiri |
| 5. Pemain menekan tombol arah atas  | |
| | 6. Sistem akan merespon dan karakter utama akan meloncat |
| 7. Pemain menekan tombol tinju  | |
| | 8. Sistem akan merespon dan karakter utama akan bergerak menyerang musuh |
| **Kondisi Akhir** | Karakter akan bergerak, jika pemain tidak menekan tombol maka karakter utama diam. |

### *j. Use Case Scenario* Perhitungan *Score*

*Use Case Scenario* dari *Use Case* Perhitungan *Score* dapat dilihat pada Tabel 3.18.

**Tabel 3. 18 *Use Case Scenario* Perhitungan *Score***

| **Identifikasi** | |
|---|---|
| **Nomor** | 10 |
| **Nama** | Perhitungan *Score* |

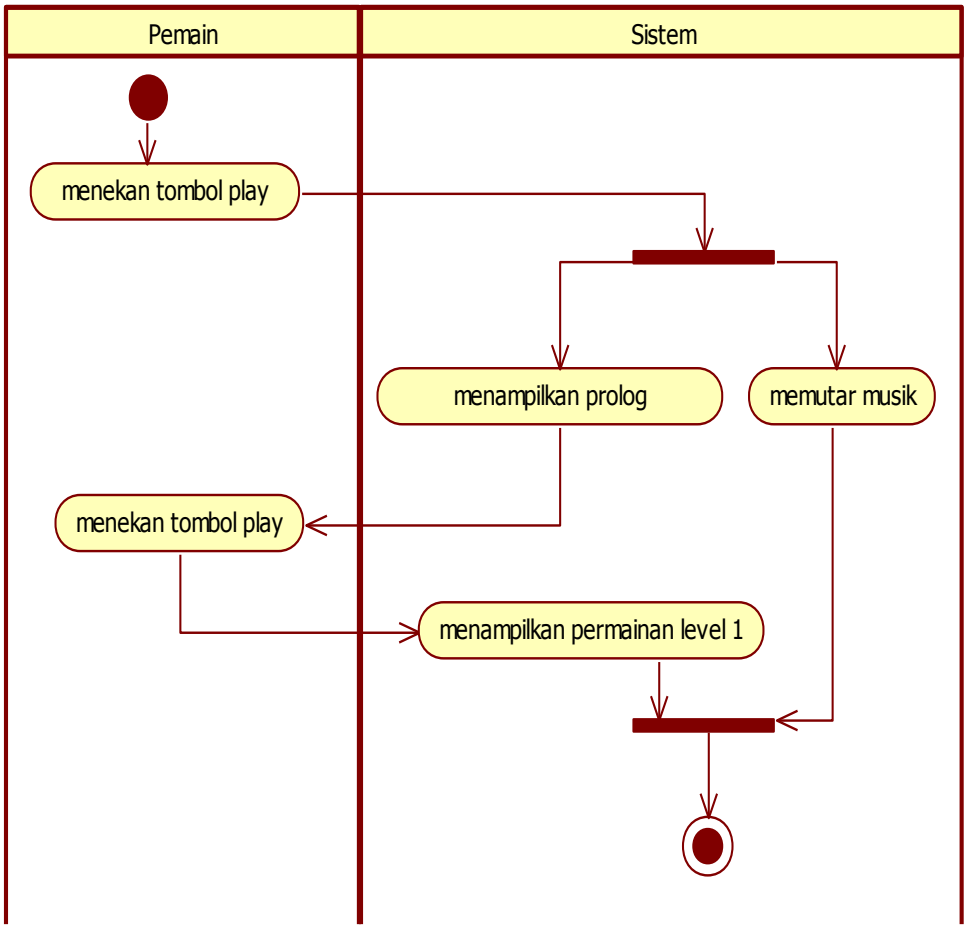
| | |
|---|---|
| **Tujuan** | Proses yang akan menginformasikan nilai skor pemain |
| **Aktor** | Pemain |
| **Skenario Utama** | |
| **Kondisi Awal** | Pemain berada di arena permainan |
| **Aksi Aktor** | **Reaksi Sistem** |
| 1. Pemain menyerang musuh | |
| | 2. Objek musuh yang terkena serangan akan hilang dari arena permainan |
| | 3. Skor Pemain akan bertambah 10 |
| **Skenario Alternatif** | |
| 1. Pemain menyerang musuh | |
| | 2. Objek musuh tidak terkena serangan |
| | 3. Skor pemain tidak akan bertambah |

### 3.2.5.5 *Activity Diagram*

*Activity diagram* menggambarkan berbagai alur 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. Berikut ini *activity diagram* pada aplikasi game *adventure* “Say No To Drugs”.

### 1. *Activity Diagram* Menampilkan *Level* 1

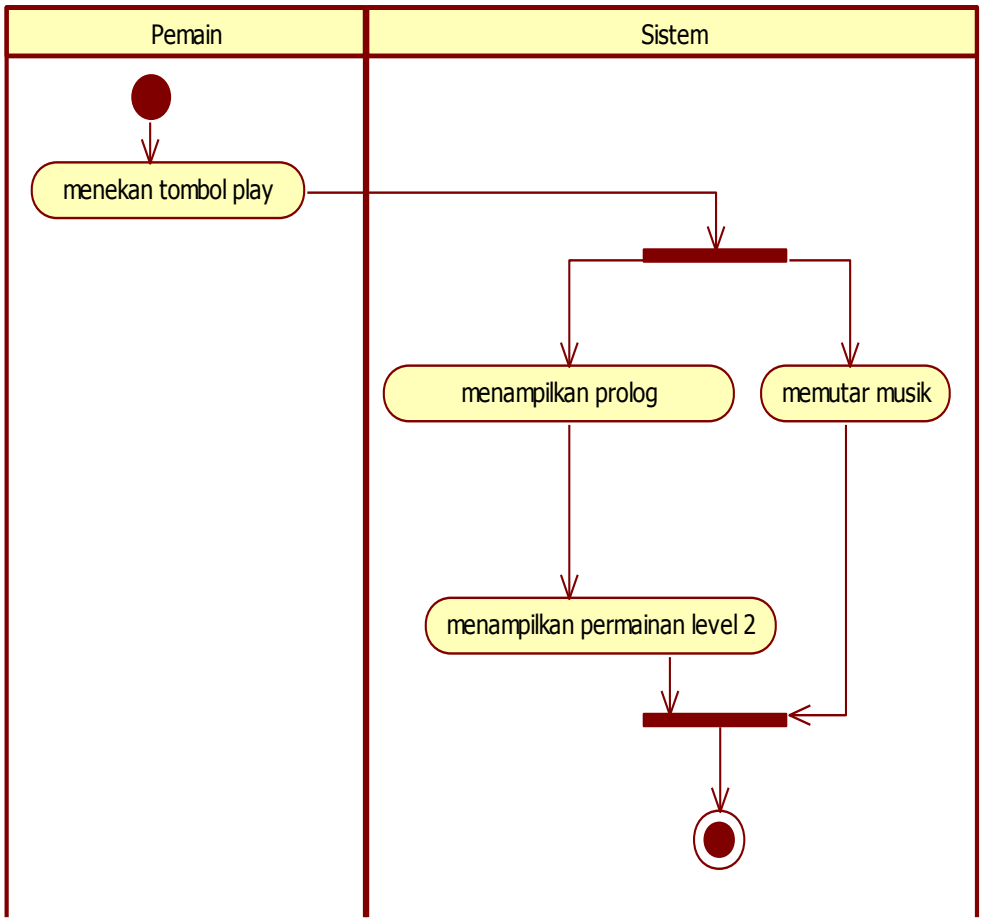
*Activity Diagram* Menampilkan *Level* 1 pada *game adventure "Say No To Drugs"* dapat dilihat pada Gambar 3.10.




**Gambar 3. 10 *Activity Diagram* Menampilkan *Level* 1**

**2. *Activity Diagram* Menampilkan *Level* 2**

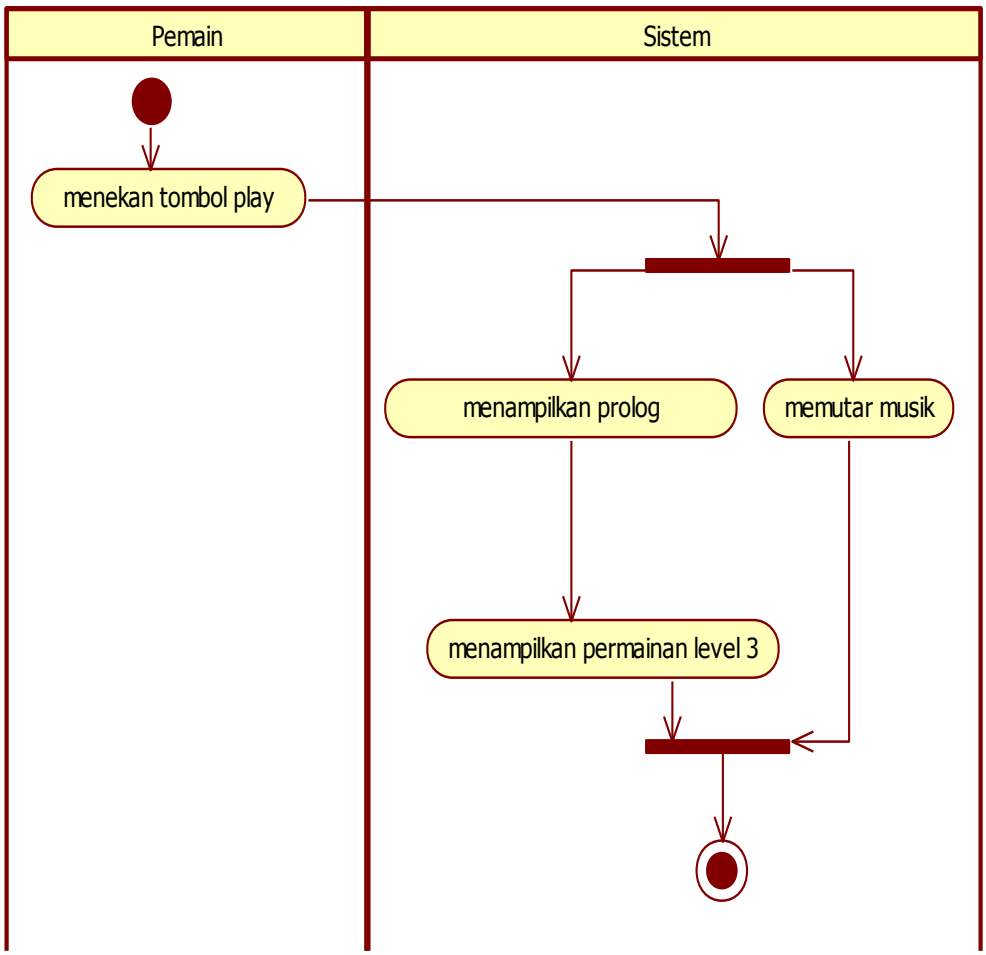
*Activity Diagram* Menampilkan *Level* 2 pada *game adventure "Say No To Drugs"* dapat dilihat pada Gambar 3.11.




**Gambar 3. 11 *Activity Diagram* Menampilkan *Level* 2**

### 3. *Activity Diagram* Menampilkan *Level* 3

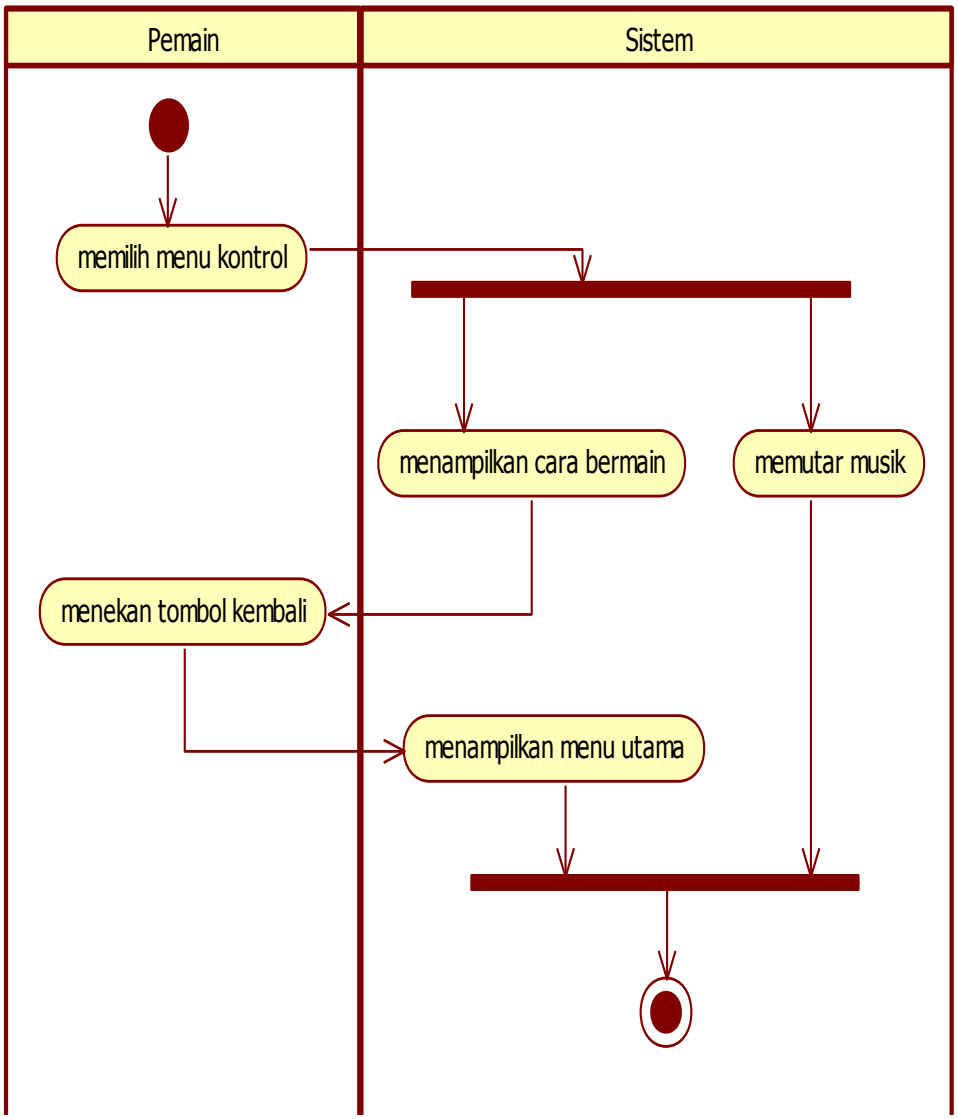
*Activity Diagram* Menampilkan *Level* 3 pada *game adventure "Say No To Drugs"* dapat dilihat pada Gambar 3.12.




**Gambar 3. 12 *Activity Diagram* Menampilkan *Level* 3**

### 4. *Activity Diagram* Cara Bermain

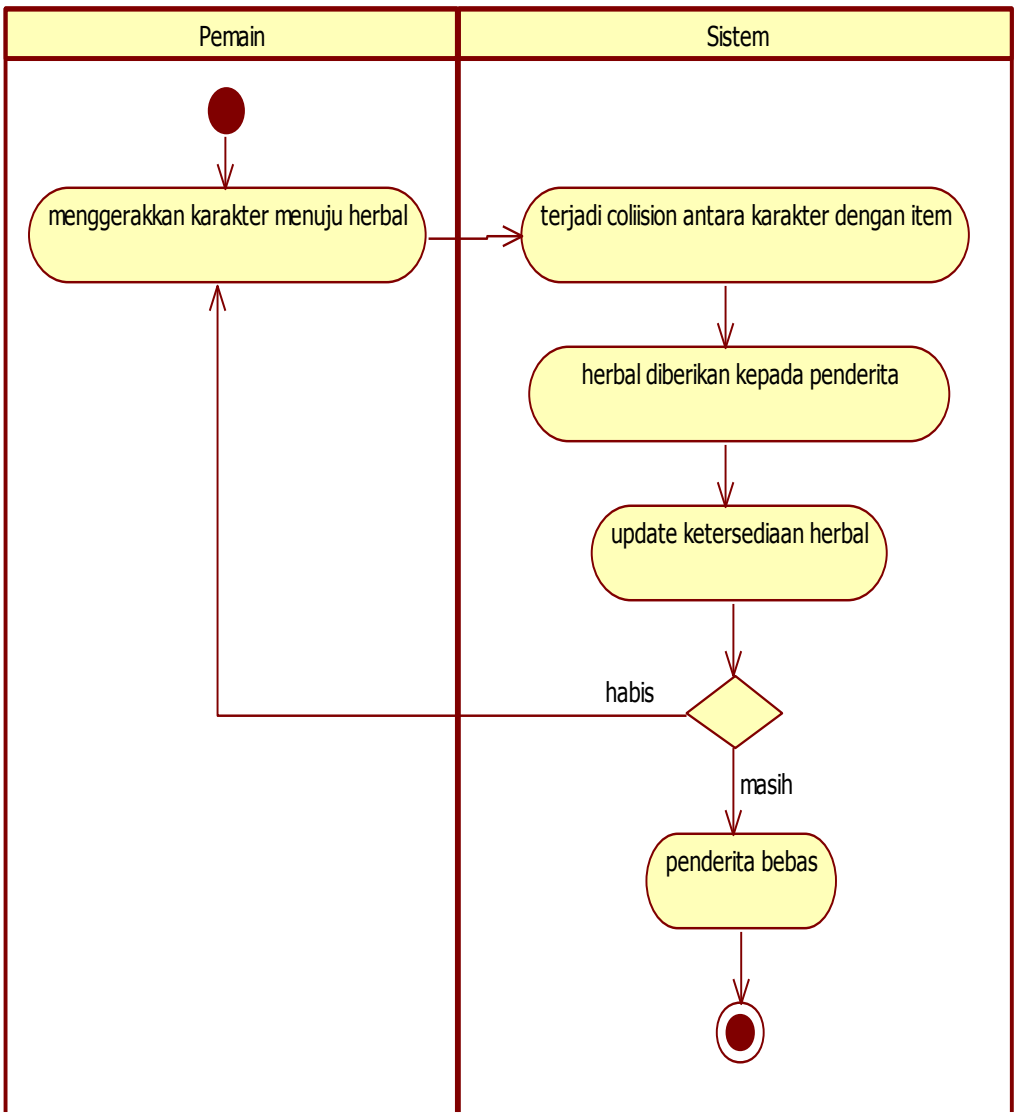
*Activity Diagram* Cara Bermain pada *game adventure "Say No To Drugs"* dapat dilihat pada Gambar 3.13.




**Gambar 3. 13 *Activity Diagram* Cara Bermain**

**5. *Activity Diagram* Mengambil Herbal**

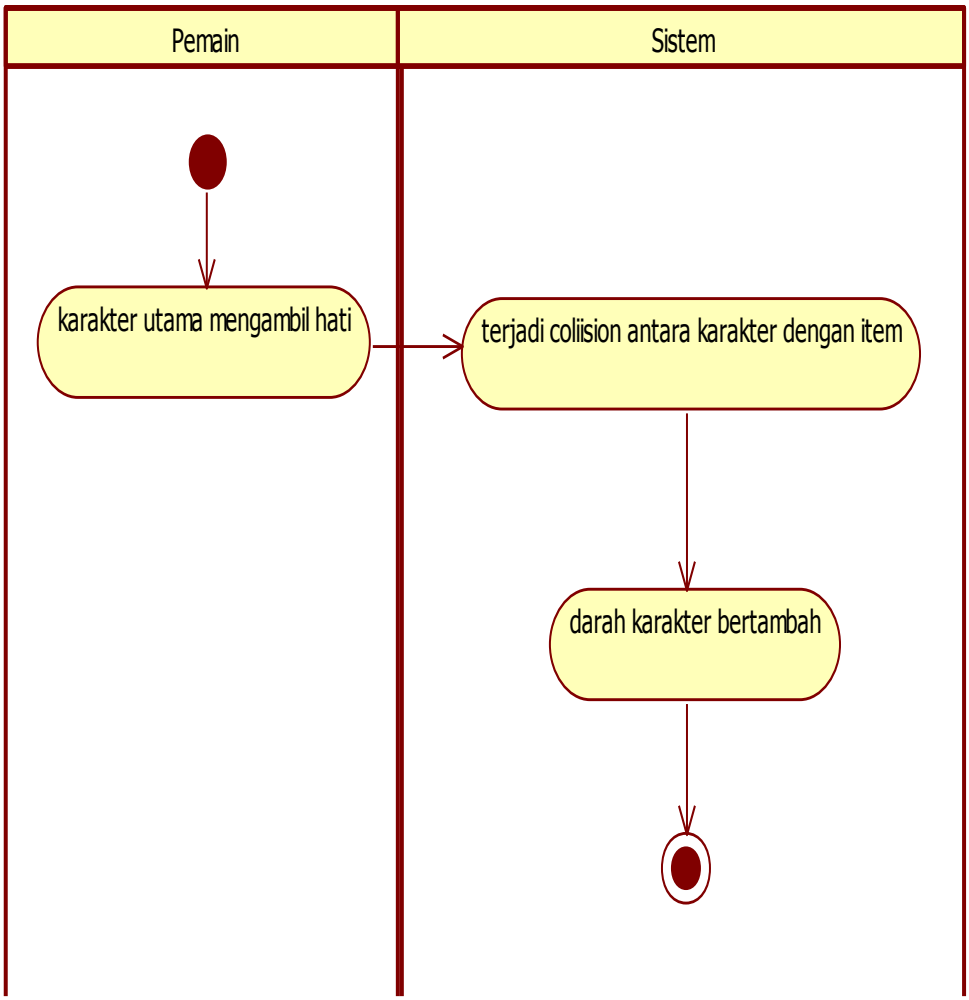
*Activity Diagram* Mengambil Herbal pada *game adventure "Say No To Drugs"* dapat dilihat pada Gambar 3.14.




**Gambar 3. 14 *Activity Diagram* Mengambil Herbal**

### 6. *Activity Diagram* Mengambil Hati

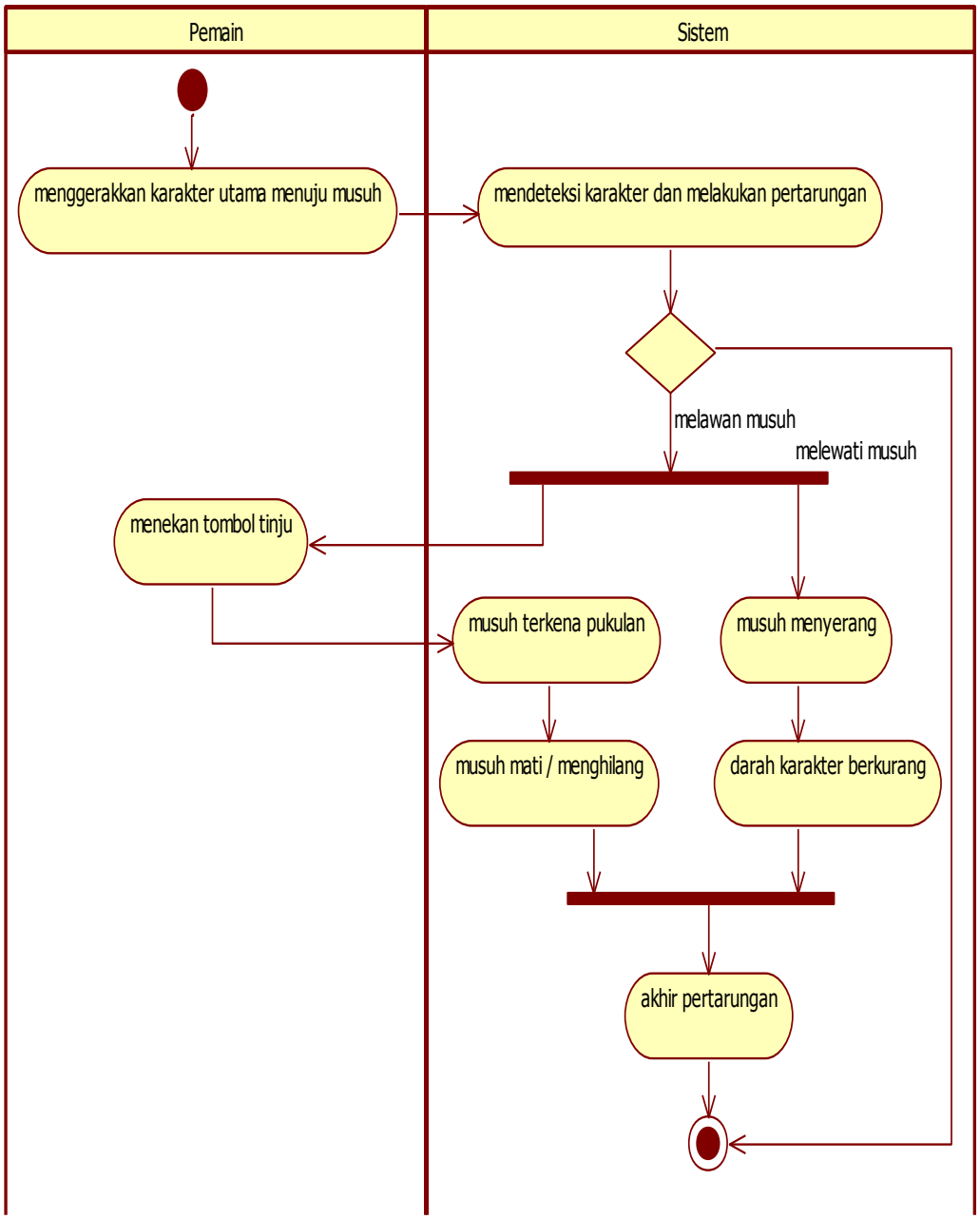
*Activity Diagram* Mengambil Hati pada *game adventure "Say No To Drugs"* dapat dilihat pada Gambar 3.15.




**Gambar 3. 15 *Activity Diagram* Mengambil Hati**

### 7. *Activity Diagram* Melawan Musuh

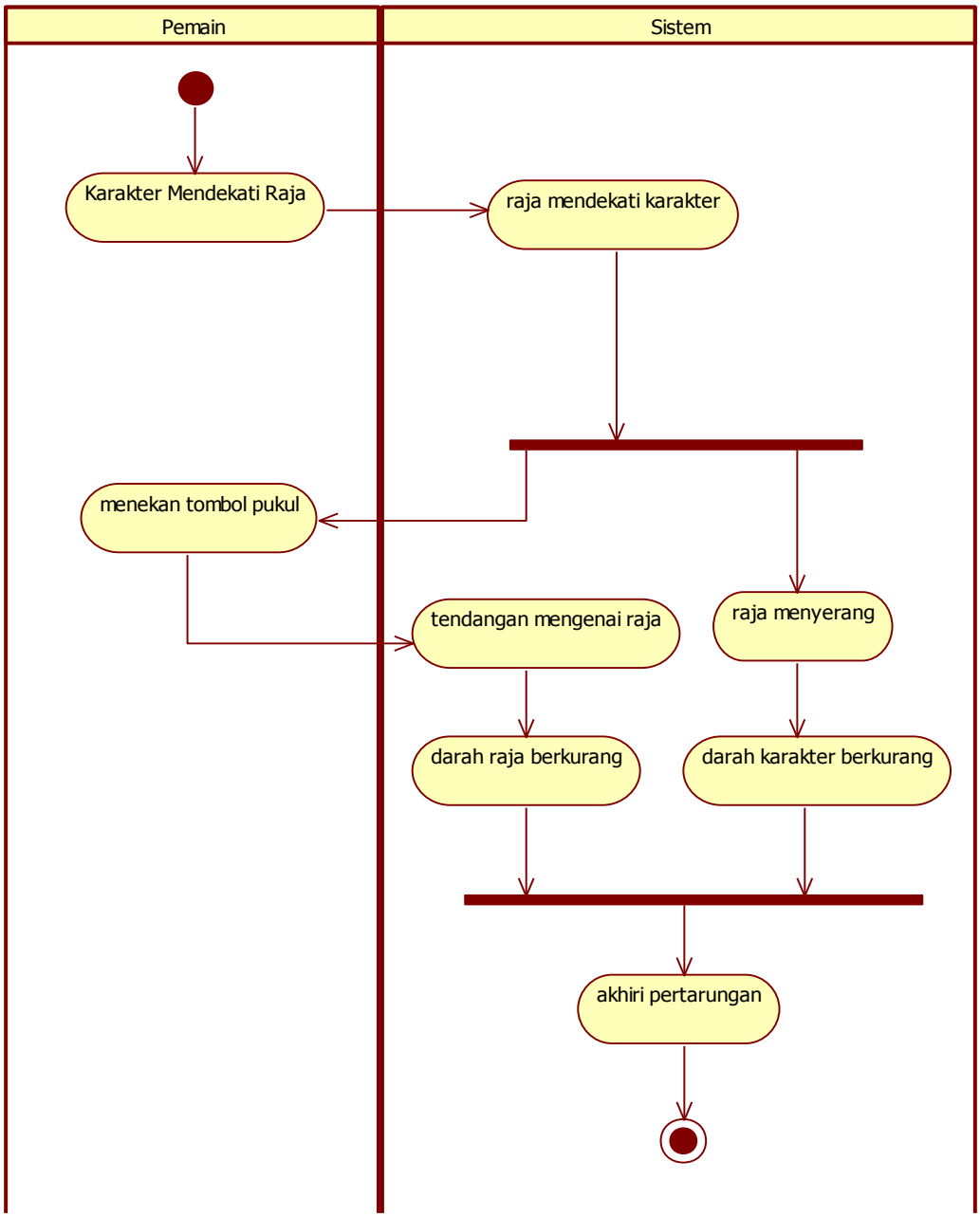
*Activity Diagram* Melawan Musuh pada *game adventure "Say No To Drugs"* dapat dilihat pada Gambar 3.16.




**Gambar 3. 16 *Activity Diagram* Melawan Musuh**

### 8. *Activity Diagram* Melawan Raja

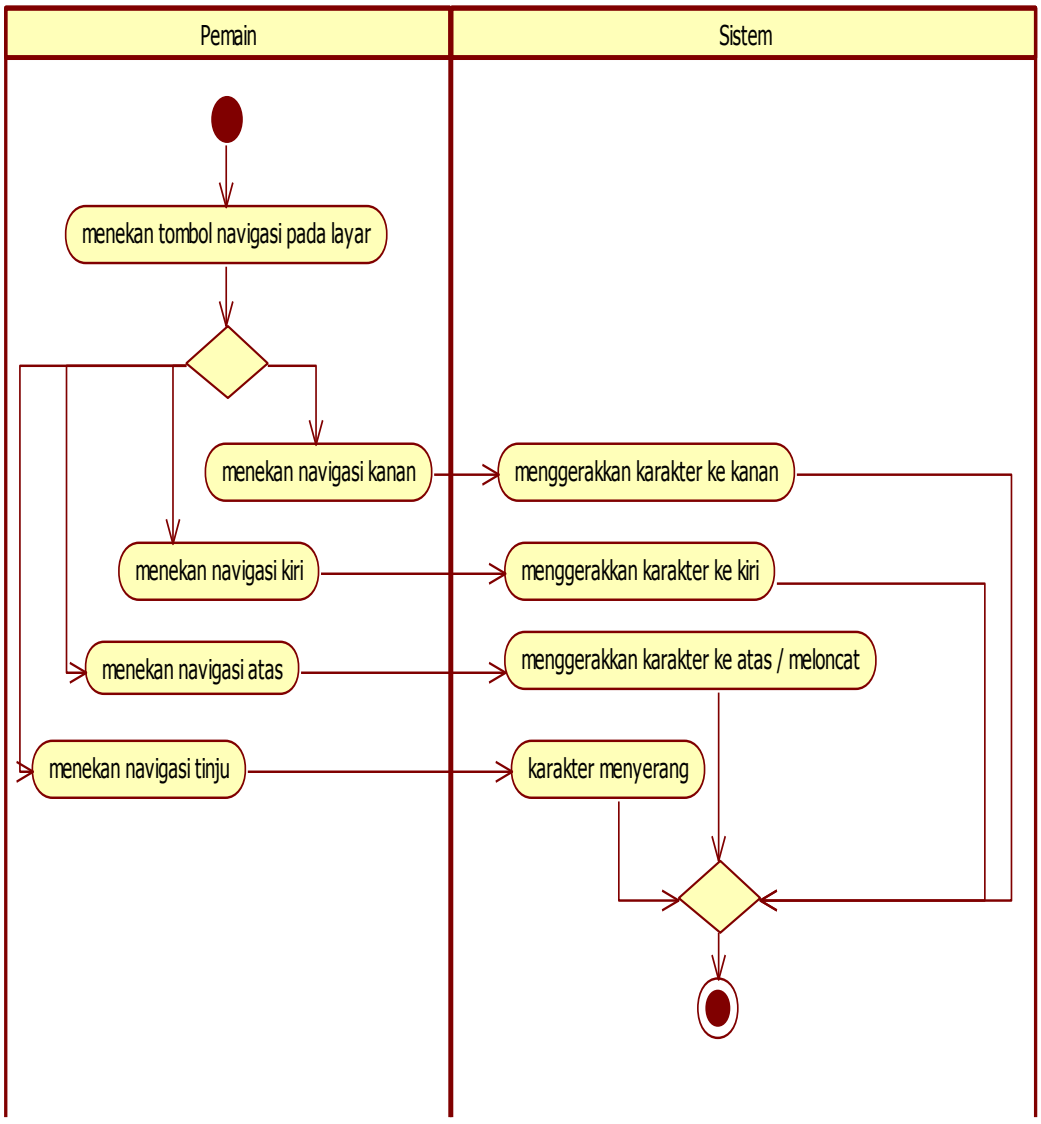
*Activity Diagram* Melawan Raja pada *game adventure "Say No To Drugs"* dapat dilihat pada Gambar 3.17.




**Gambar 3. 17 *Activity Diagram* Melawan Raja**

### 9. *Activity Diagram* Mengendalikan Karakter

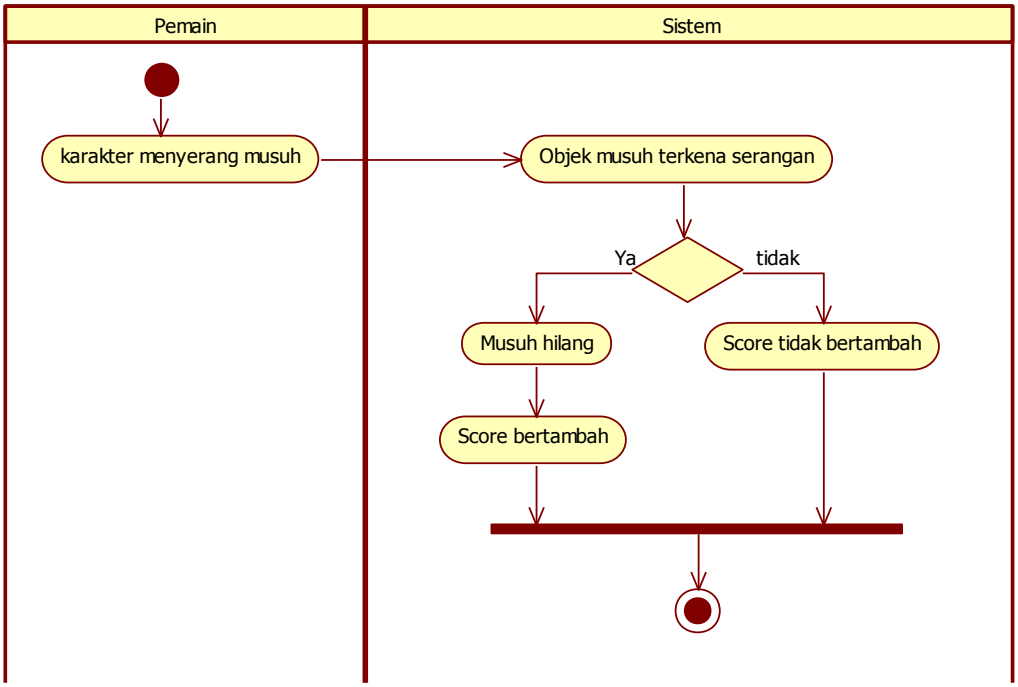
*Activity Diagram* Mengendalikan Karakter pada *game adventure "Say No To Drugs"* dapat dilihat pada Gambar 3.18.




**Gambar 3. 18 *Activity Diagram* Mengendalikan Karakter**

**10. *Activity Diagram* Perhitungan *Score***

*Activity Diagram* Perhitungan *Score* pada *game adventure "Say No To Drugs"* dapat dilihat pada Gambar 3.19.



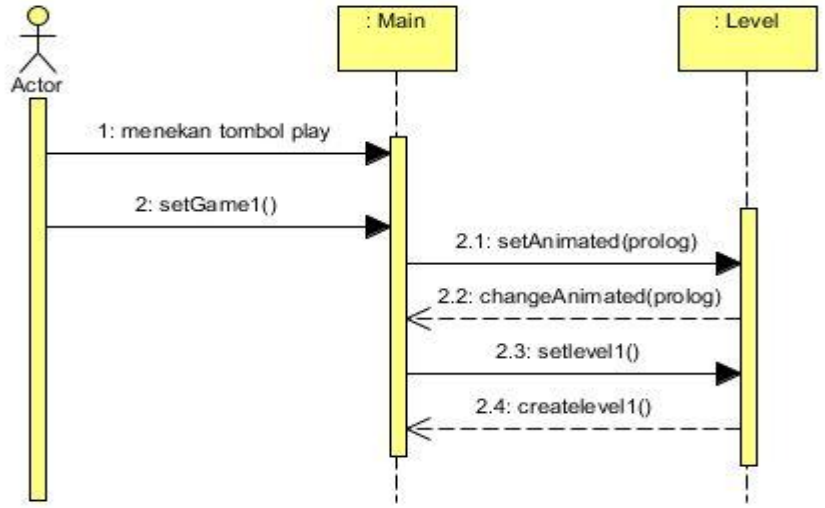

**Gambar 3. 19 *Activity Diagram* Perhitungan *Score***

### 3.2.5.6 *Sequence Diagram*

*Sequence diagram* yaitu menggambarkan interaksi antar masing-masing objek pada setiap *use case*. *Sequence diagram* pada *game adventure "Say No To Drugs"* terdiri dari *sequence diagram* menampilkan *sequence diagram* mulai permainan, *sequence diagram* tentang pembuat, *sequence diagram* cara bermain, dan *sequence diagram* keluar. Penjelasan *sequence diagram game adventure "Say No To Drugs"* dapat dilihat sebagai berikut :

### 1. *Sequence Diagram* Menampilkan *Level* 1

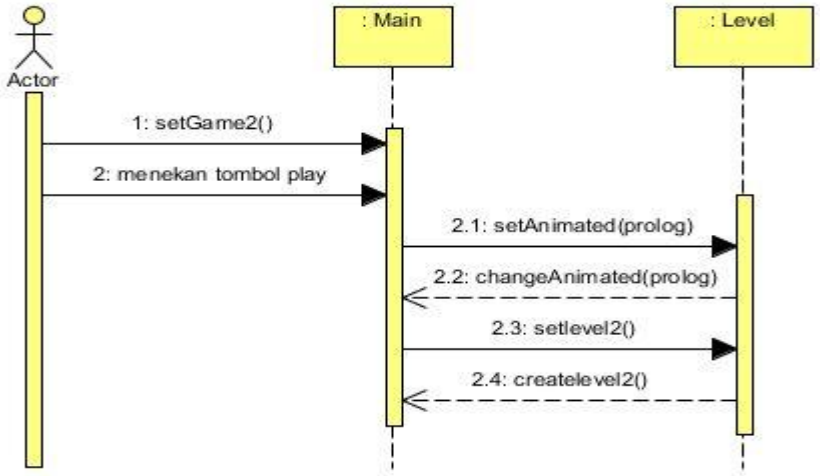
*Sequence Diagram* Menampilkan *Level* 1 pada *game "Say No To Drugs"* dapat dilihat pada gambar 3.20.




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

### 2. *Sequence Diagram* Menampilkan *Level* 2

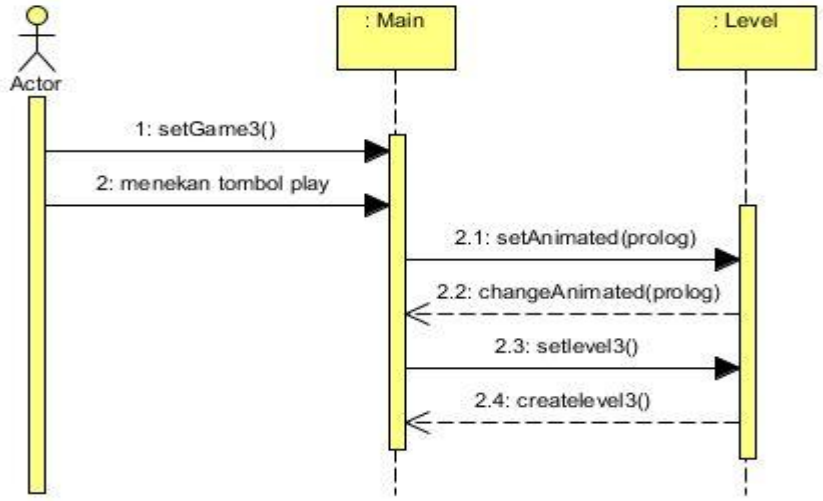
*Sequence Diagram* Menampilkan *Level* 2 pada *game "Say No To Drugs"* dapat dilihat pada gambar 3.21.




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

### 3. *Sequence Diagram* Menampilkan *Level* 3

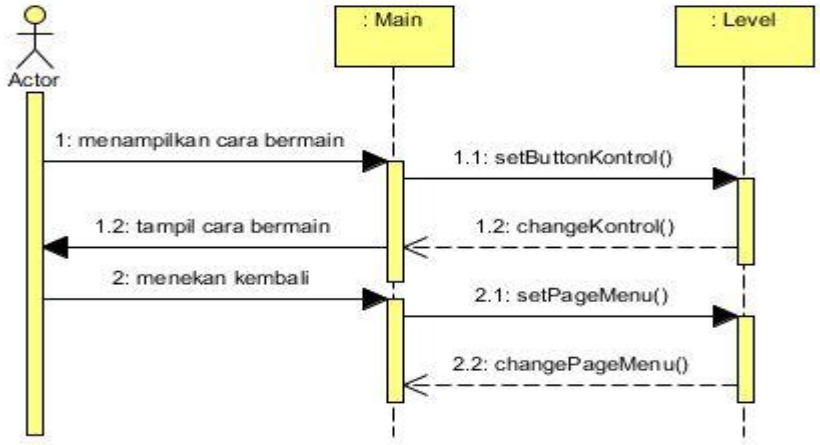
*Sequence Diagram* Menampilkan *Level* 3 pada *game "Say No To Drugs"* dapat dilihat pada gambar 3.22.




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

### 4. *Sequence Diagram* Cara Bermain

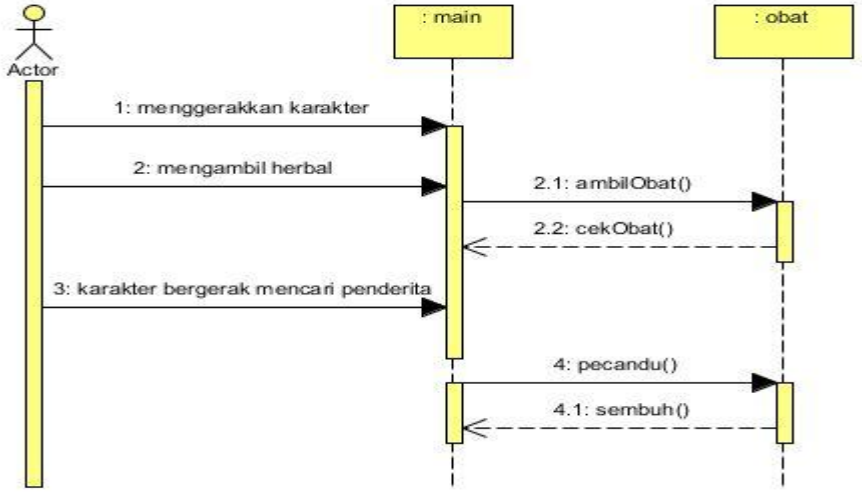
*Sequence Diagram* Cara Bermain pada *game "Say No To Drugs"* dapat dilihat pada gambar 3.23.




**Gambar 3. 23 *Sequence Diagram* Cara Bermain**

### 5. *Sequence Diagram* Mengambil Herbal

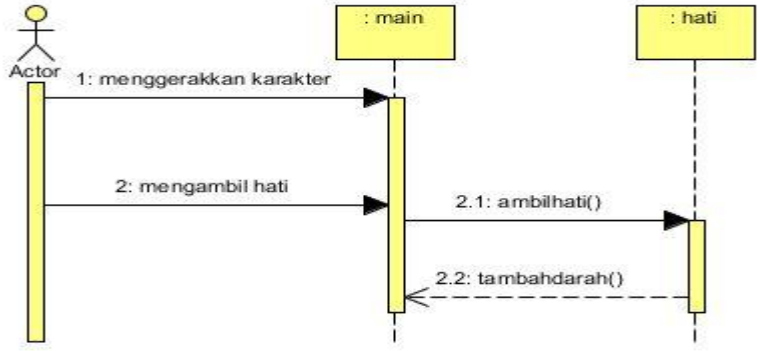
*Sequence Diagram* Mengambil Herbal pada *game* "*Say No To Drugs"* dapat dilihat pada gambar 3.24.




**Gambar 3. 24 *Sequence Diagram* Mengambil Herbal**

### 6. *Sequence Diagram* Mengambil Hati

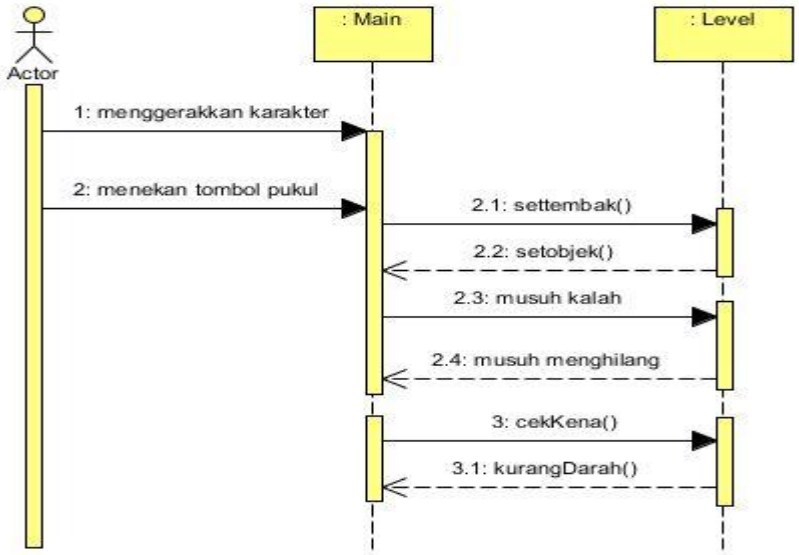
*Sequence Diagram* Mengambil Hati pada *game* "*Say No To Drugs*" dapat dilihat pada gambar 3.25.




**Gambar 3. 25 *Sequence Diagram* Mengambil Hati**

### 7. *Sequence Diagram* Melawan Musuh

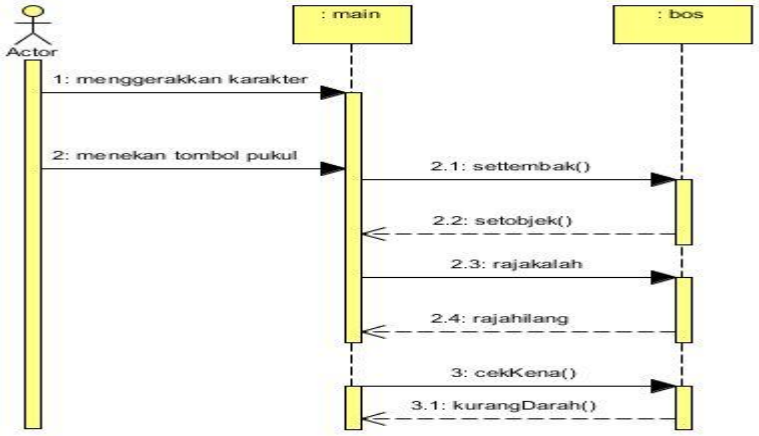
*Sequence Diagram* Melawan Musuh pada *game* "*Say No To Drugs*" dapat dilihat pada gambar 3.26.




**Gambar 3. 26 *Sequence Diagram* Melawan Musuh**

### 8. *Sequence Diagram* Melawan Raja

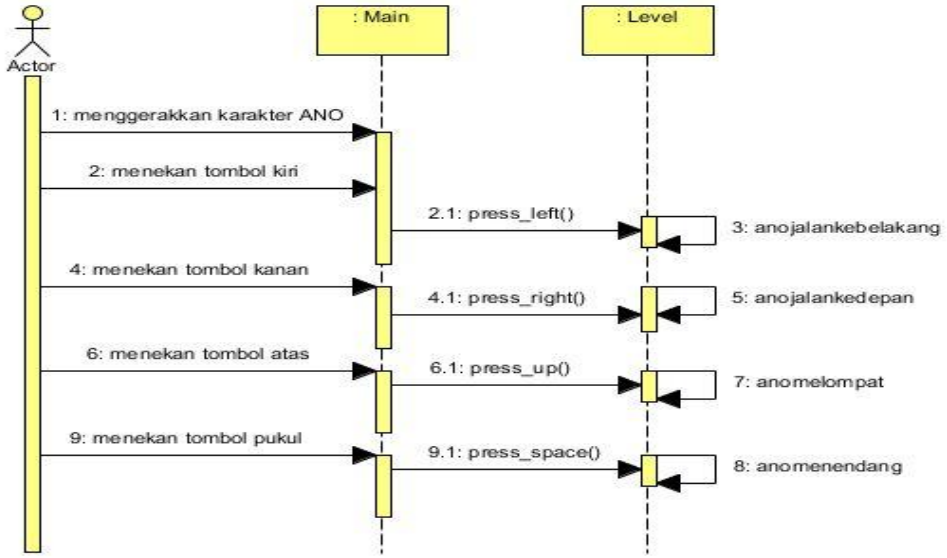
*Sequence Diagram* Melawan Raja pada *game* "*Say No To Drugs*" dapat dilihat pada gambar 3.27.




**Gambar 3. 27 *Sequence Diagram* Melawan Raja**

### 9. *Sequence Diagram* Mengendalikan Karakter

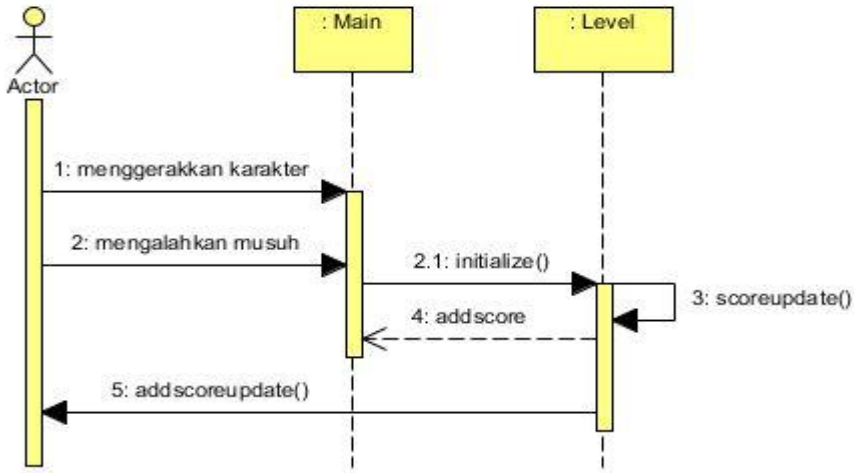
*Sequence Diagram* Mengendalikan Karakter pada *game* "*Say No To Drugs*" dapat dilihat pada gambar 3.28.




**Gambar 3. 28 *Sequence Diagram* Mengendalikan Karakter**

### 10. *Sequence Diagram* Perhitungan *Score*

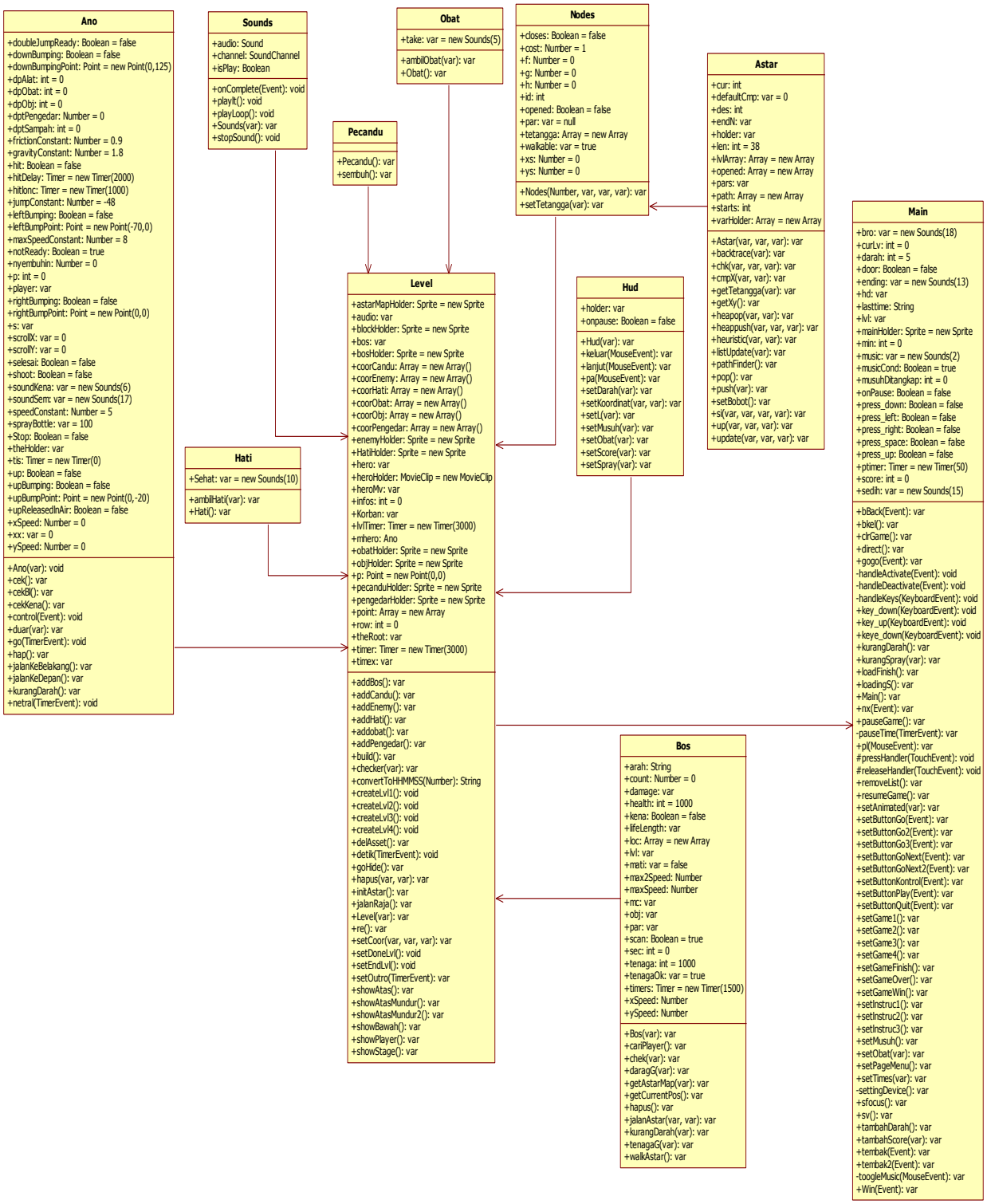
*Sequence Diagram* Perhitungan *Score* pada *game* "*Say No To Drugs*" dapat dilihat pada gambar 3.29.




**Gambar 3. 29 *Sequence Diagram* Perhitungan *Score***

### 3.2.5.7 *Class Diagram*

Diagram kelas atau *class diagram* menggambarkan struktur sistem dari segi pendefinisian kelas yang dibuat pada game "*Say No To Drugs*", struktur yang ada meliputi atribut-atribut dan method-method yang ada pada masing-masing class. *Class Diagram* dari game "*Say No To Drugs*" dapat dilihat pada gambar 3.30.




**Gambar 3. 30** ***Class Diagram Say No To Drugs***

### 3.2.6 Perancangan Sistem

Perancangan sistem merupakan kegiatan untuk merancang aplikasi yang akan di bangun, tahapannya dimulai dari perancangan komponen permainan, perancangan struktur menu, dan perancangan antarmuka dari *game adventure "Say No To Drugs".*

#### 3.2.6.1 Perancangan Komponen Permainan

Di dalam bagian ini akan dijelaskan karakter, pergerakan karakter, dan item yang terdapat pada *game adventure "Say No To Drugs".*

##### 3.2.6.1.1 Karakter *Game Say No To Drugs*

Di dalam bagian ini akan dijelaskan karakter-karakter yang terdapat pada *game* "*Say No To Drugs*" seperti yang bisa dilihat pada tabel 3.19 berikut ini.

**Tabel 3. 19 Karakter *Game Say No To Drugs***

| No | Karakter | Keterangan |
|---|---|---|
| 1 |  | Nama karakter :<br>**ANO** merupakan karakter utama dari *game "Say No To Drugs".* |
| 2 |  | Nama karakter :<br>**Monster Ganja** sebagai musuh ANO yang terdapat di *level* 1. |
| 3 |  | Nama karakter :<br>**Monster HIV** sebagai musuh ANO yang terdapat di *level* 2. |
| 4 |  | Nama karakter :<br>**Monster Jarum Suntik** sebagai musuh ANO yang terdapat di *level* 3. |
| 5 |  | Nama karakter :<br>**Penderita** adalah orang yang terkena narkoba yang harus di sembuhkan oleh ANO, terdapat pada level 1 dan *level* 3**.** |

| | | |
|---|---|---|
| 6 |  | Nama karakter :<br>**Pengedar** yang harus di tangkap oleh ANO di *level* 2 dan *level* 3. |
| 7 |  | Nama karakter :<br>**Raja Narkoba** yang harus dimusnahkan oleh ANO, terdapat di akhir *level*. |

### 3.2.6.1.2 *Item Game Say No To Drugs*

Item pendukung yang digunakan pada aplikasi *game* "*Say No To Drugs*". Berikut ini Tabel 3.20 deskripsi item pada aplikasi *game* "*Say No To Drugs*".

**Tabel 3. 20 Item *Game Say No To Drugs***

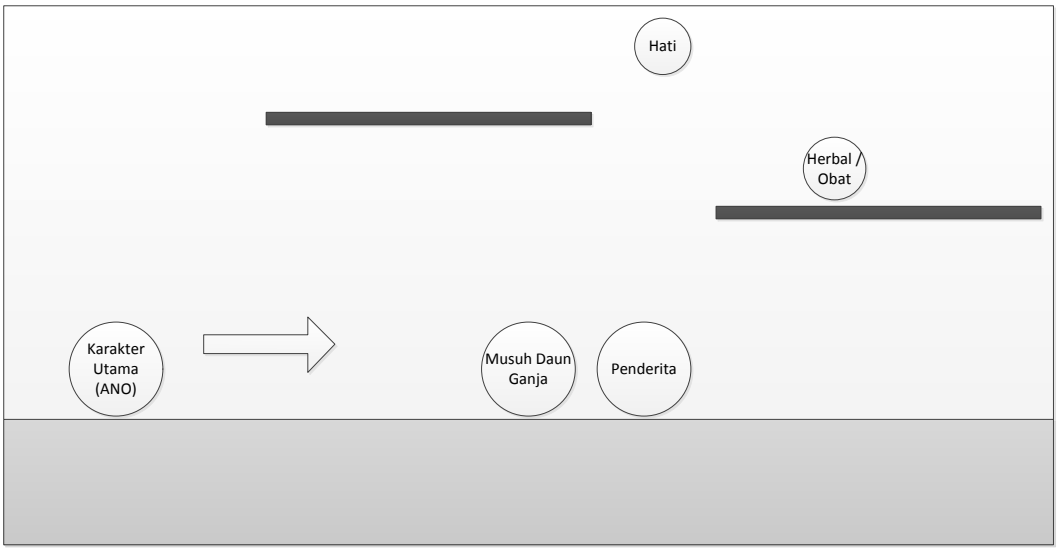
| No | *Item* | Keterangan |
|---|---|---|
| 1 |  | Nama *item*: Herbal / Obat untuk menyembuhkan penderita yang terkena narkoba |
| 2 |  | Nama *item*: Hati digunakan untuk menambah darah karakter utama (ANO). |

### 3.2.6.1.3 *Story Board Game Say No To Drugs*

*Story board* adalah sketsa gambar yang disusun berurutan sesuai dengan naskah, dengan *story board* ide cerita dapat disampaikan kepada orang lain dengan lebih mudah, karena dapat menggiring khayalan seseorang mengikuti gambar-gambar yang tersaji sehingga menghasilkan persepsi yang sama. Berikut merupakan *story board* dari *game adventure* "*Say No To Drugs"*:

### 1. *Story Board Level* 1

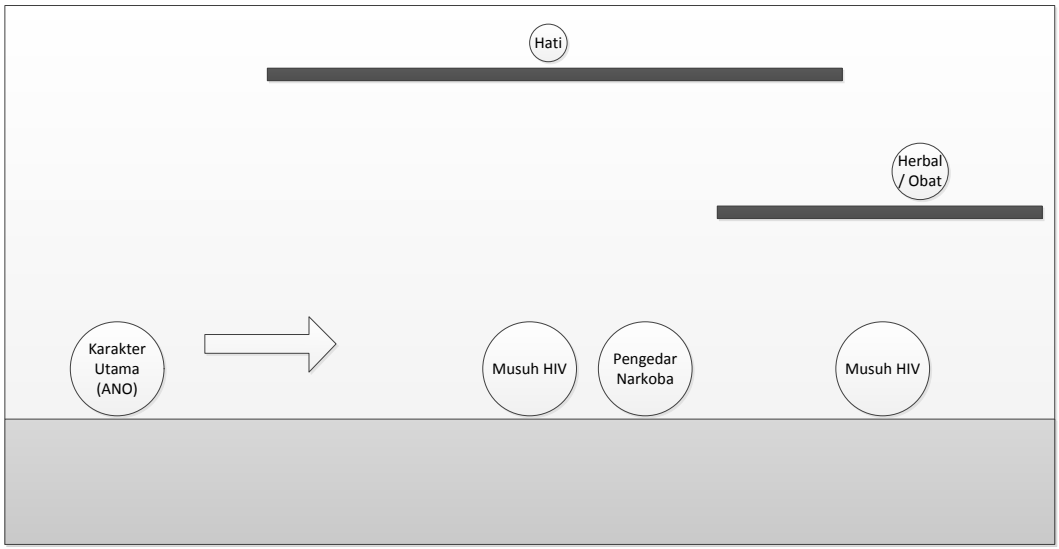
*Story board* aplikasi *game* "*Say No To Drugs*" *level 1* dapat di lihat pada gambar 3.31.




**Gambar 3. 31 *Story Board Game Say No To Drugs Level* 1**

### 2. *Story Board Level* 2

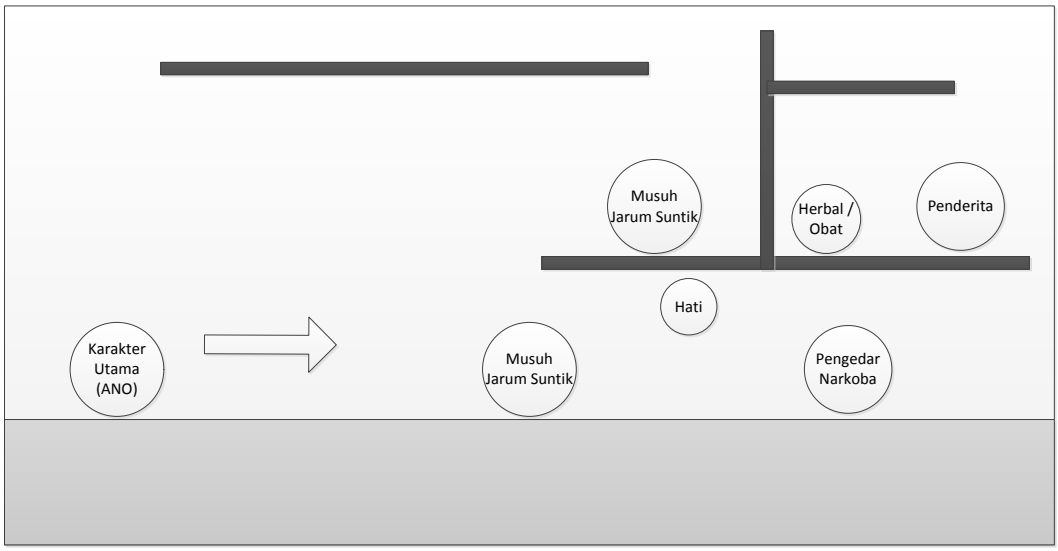
*Story board* aplikasi *game* "*Say No To Drugs*" *level* 2 dapat di lihat pada gambar 3.32.




**Gambar 3. 32 *Story Board Game Say No To Drugs Level* 2**

### 3. *Story Board Level 3*

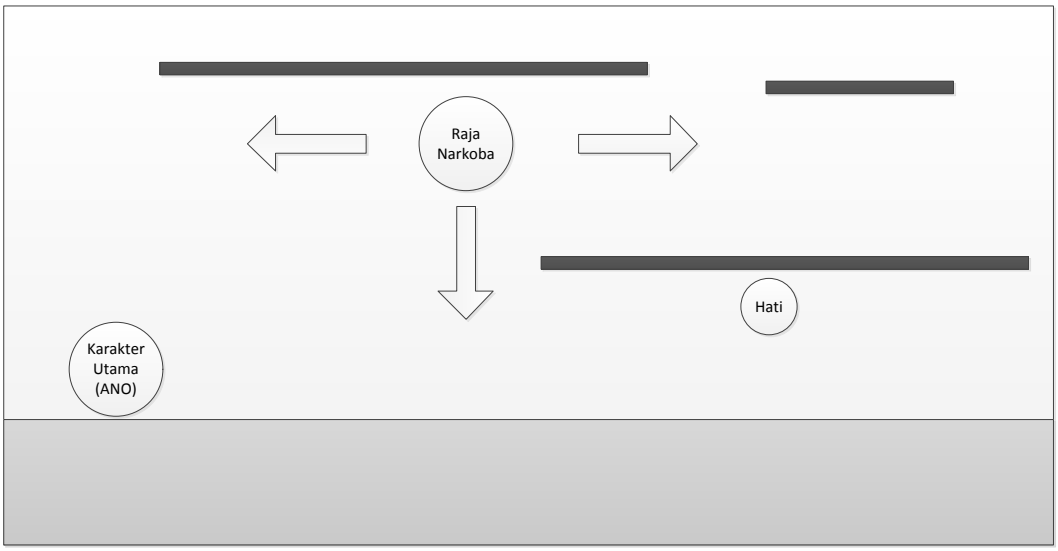
*Story board* aplikasi *game* "*Say No To Drugs*" *level* 3 dapat di lihat pada gambar 3.33.




**Gambar 3. 33 *Story Board Game Say No To Drugs Level* 3**

### 4. *Story Board* Melawan Raja Narkoba

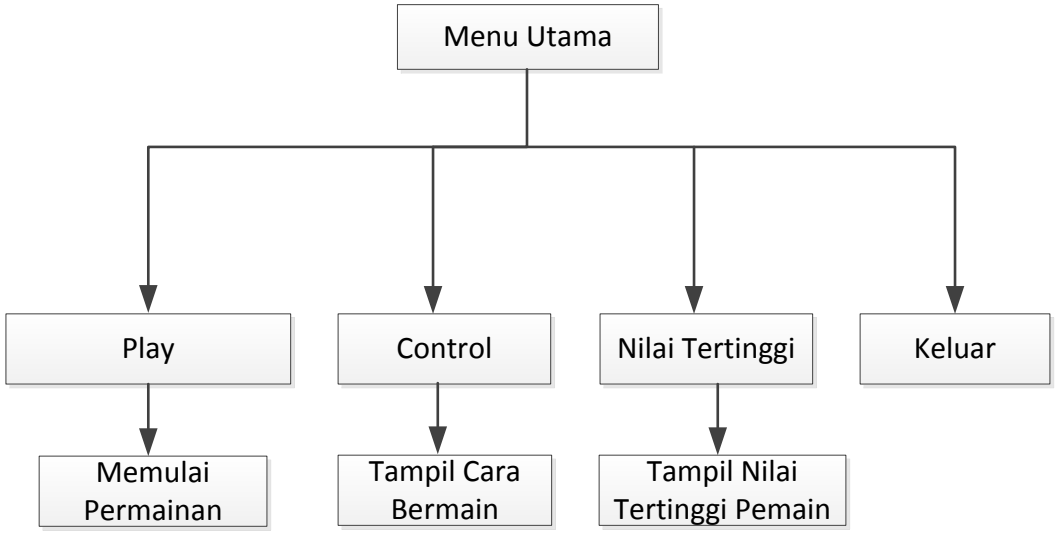
*Story board* aplikasi *game* "*Say No To Drugs*" melawan raja narkoba dapat di lihat pada gambar 3.34.




**Gambar 3. 34 *Story Board Game Say No To Drugs* Melawan Raja Narkoba**

### 3.2.6.2 Perancangan Struktur Menu

Perancangan struktur menu pada *game "Say No To Drugs" dapat dilihat* pada gambar 3.35.




**Gambar 3. 35 Perancangan Struktur Menu Utama**

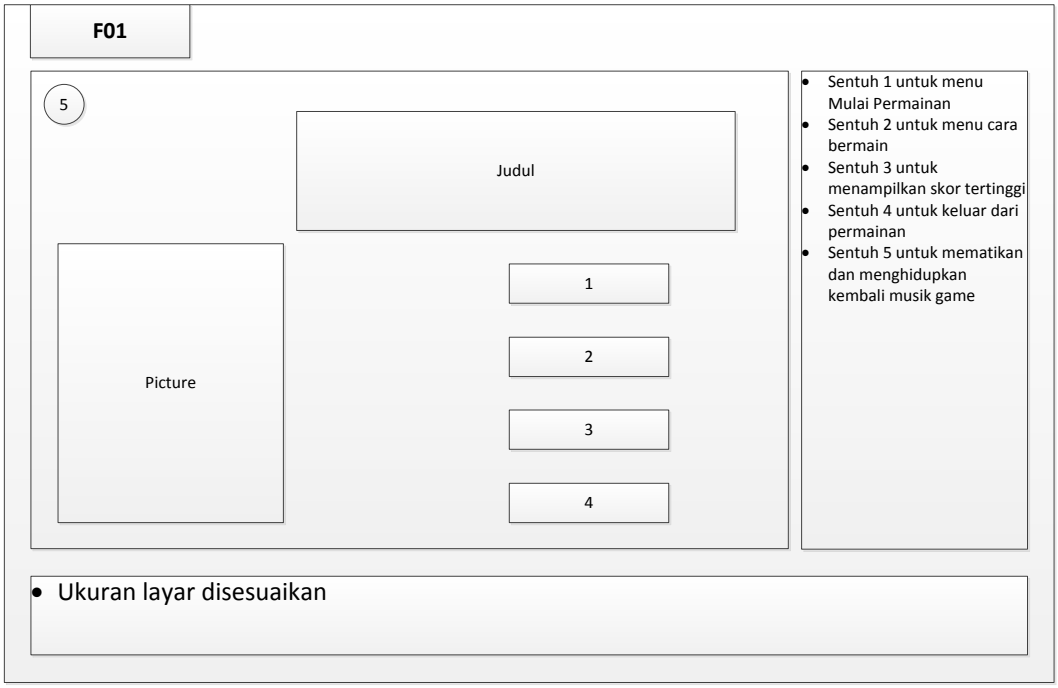
### 3.2.6.3 Perancangan Antar Muka

Perancangan antar muka bertujuan untuk memberikan gambaran tentang aplikasi yang akan dibangun, sehingga akan mempermudah dalam mengimplementasikan aplikasi serta akan memudahkan pembuatan aplikasi.

Pada bagian ini akan dijelaskan perancangan antar muka menu utama, perancangan antar muka *menu play*, perancangan antar muka intro *game*, perancangan antar muka cara bermain, perancangan antar muka nilai tertinggi, perancangan antar muka kalah, dan perancangan antar muka menu permainan.

#### 3.2.6.3.1 Perancangan Antar Muka Menu Utama

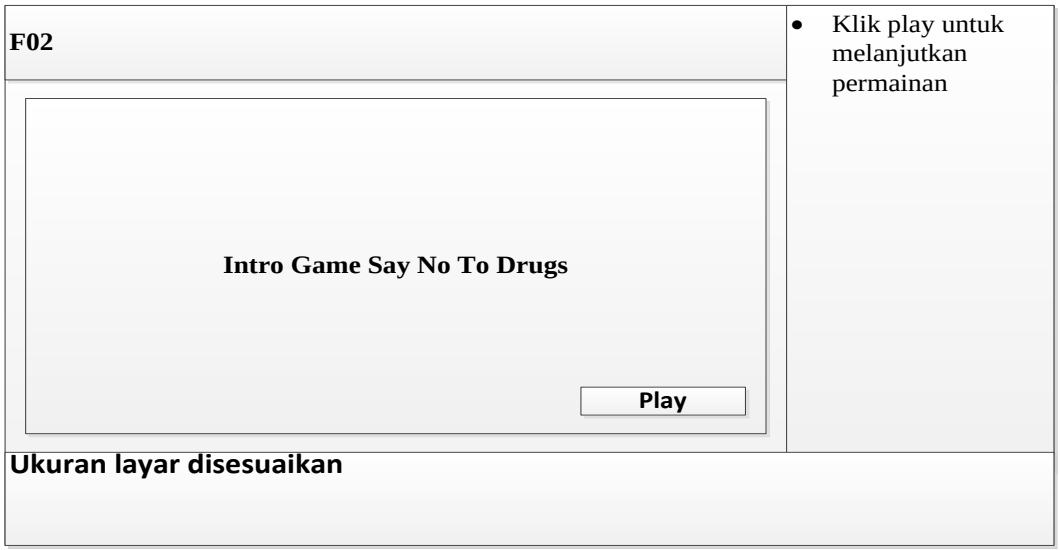
Menu utama merupakan menu yang akan muncul pertama kali pada saat pengguna membuka *game* seperti yang terlihat pada gambar 3.36.




**Gambar 3. 36 Perancangan Antar Muka Menu Utama**

### 3.2.6.3.2 Perancangan Antar Muka Intro *Game*

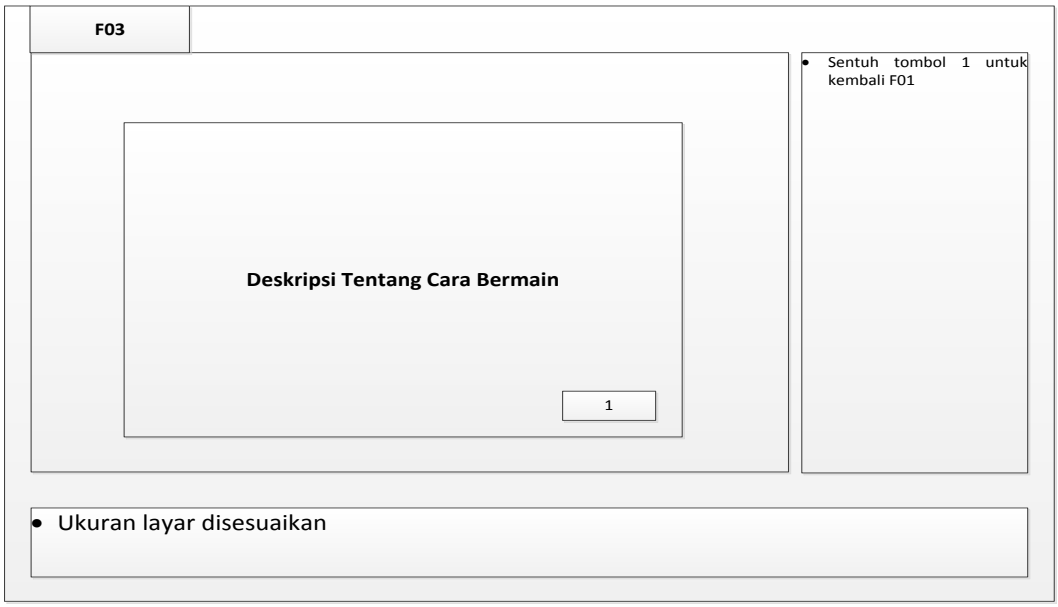
Intro game merupakan menu yang akan muncul sebelum masuk ke arena permainan. Menu intro *game* dapat di lihat pada gambar 3.37.




**Gambar 3. 37 Perancangan Antar Muka Intro *Game***

### 3.2.6.3.3 Perancangan Antar Muka Cara Bermain

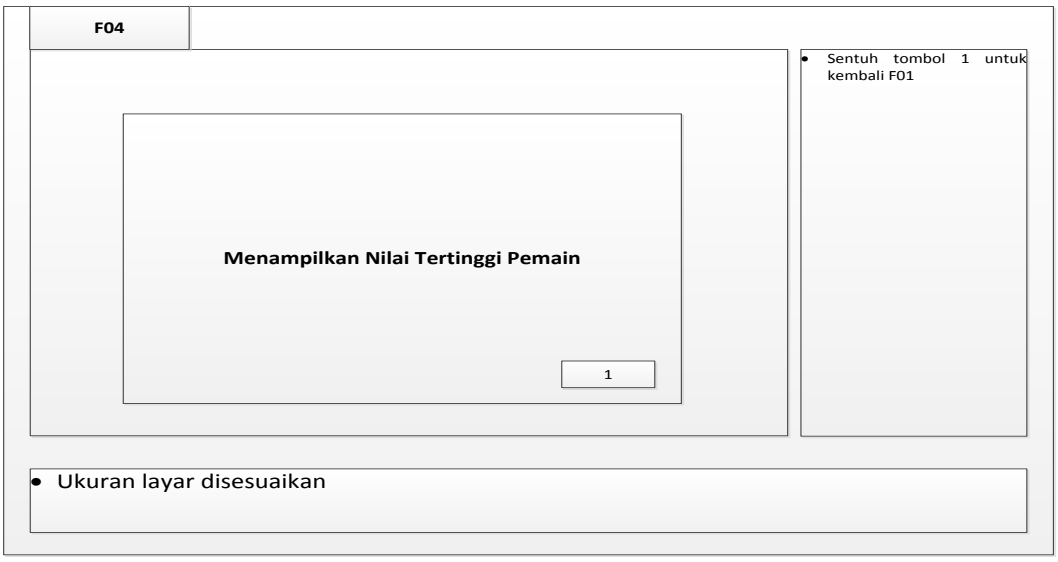
Perancangan antarmuka cara bermain pada *game* "Say No To Drugs" dapat dilihat pada gambar 3.38.




**Gambar 3. 38 Perancangan Antar Muka Cara Bermain**

### 3.2.6.3.4 Perancangan Antar Muka Menu Nilai Tertinggi

Perancangan antarmuka cara bermain pada *game* "Say No To Drugs" dapat dilihat pada gambar 3.39.




**Gambar 3. 39 Perancangan Antar Muka Nilai Tertinggi**

### 3.2.6.3.5 Perancangan Antar Muka Menu *Play*

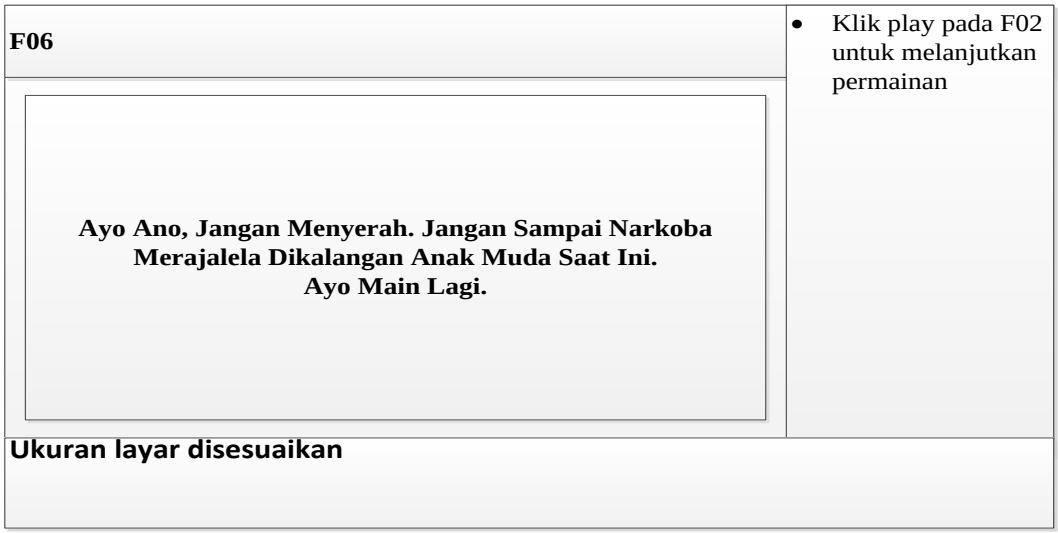
Perancangan menampilkan *Menu Play* pada *game* "*Say No To Drugs*" dapat dilihat pada gambar 3.40.




**Gambar 3. 40 Perancangan Antar Muka Menu *Play***

### 3.2.6.3.6 Perancangan Antar Muka Kalah

Perancangan antar muka kalah pada *game* "*Say No To Drugs*" dapat dilihat pada gambar 3.41.

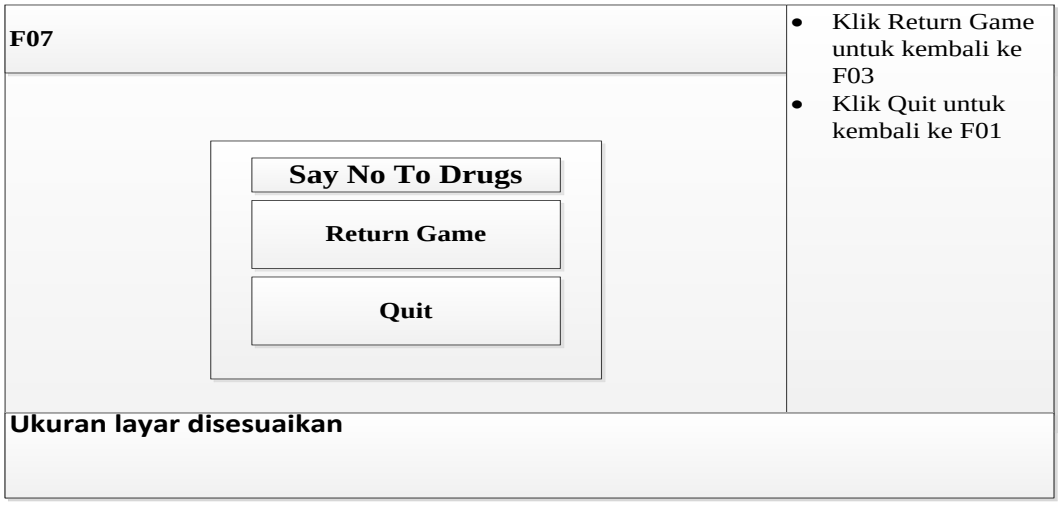



.

**Gambar 3. 41 Perancangan Antar Muka Kalah**

### 3.2.6.3.7 Perancangan Antar Muka Menu Permainan

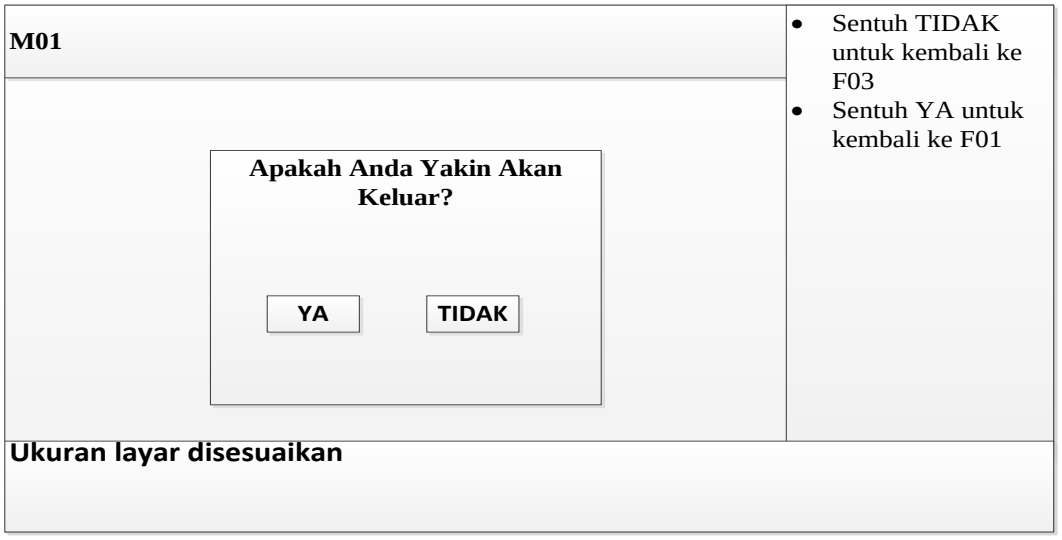
Perancangan antar muka kalah pada *game* "*Say No To Drugs*" dapat dilihat pada gambar 3.42.




**Gambar 3. 42 Perancangan Antar Muka Menu Permainan**

### 3.2.6.4 Perancangan Pesan

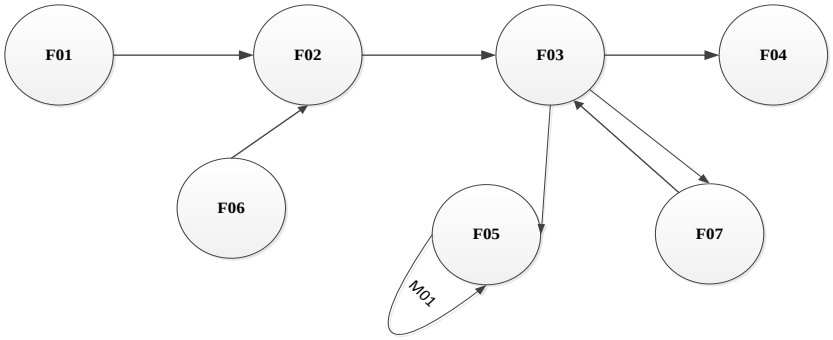
Pesan digunakan untuk memberi pemberitahuan, peringatan atau untuk meminta konfirmasi dari suatu aksi. Pesan dalam *game* "Say No To Drugs" dapat dilihat pada gambar 3.43 di bawah ini:




**Gambar 3. 43 Perancangan pesan *game Say No To Drugs***

### 3.2.7 Jaringan Semantik

Jaringan semantik merupakan jaringan data yang menunjukan hubungan antar berbagai objek. Gambar 3.44 menunjukkan jaringan semantik dari *game* "*Say No To drugs*" yang dibangun.




**Gambar 3. 44 Jaringan Semantik**