BAB 3
ANALISIS DAN PERANCANGAN

1.1 Analisis Sistem

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

1.1.1 Analisis *Game* Sejenis yang Sudah Ada

Sebelum merancang *game* yang akan dibuat, terlebih dahulu dilakukan analisis pada *game* sejenis yang sudah ada. Ini bertujuan agar *game* yang sudah dibuat nanti memiliki solusi dari permasalahan *game* yang sudah ada.

Game City Adventure sudah ada beberapa, diantara nya *Big City Adventure – San Francisco* dan *Big City Adventure – Sidney*.

1. *Big City Adventure – San Francisco*

Game Big City Adventure – San Francisco adalah *game* yang dikembangkan oleh *Big Fish Games*, Inc yang bermarkas di Seattle, Amerika. *Game* ini berburu harta karun dan mencari objek – objek yang tersembunyi diseluruh Kota San Francisco.

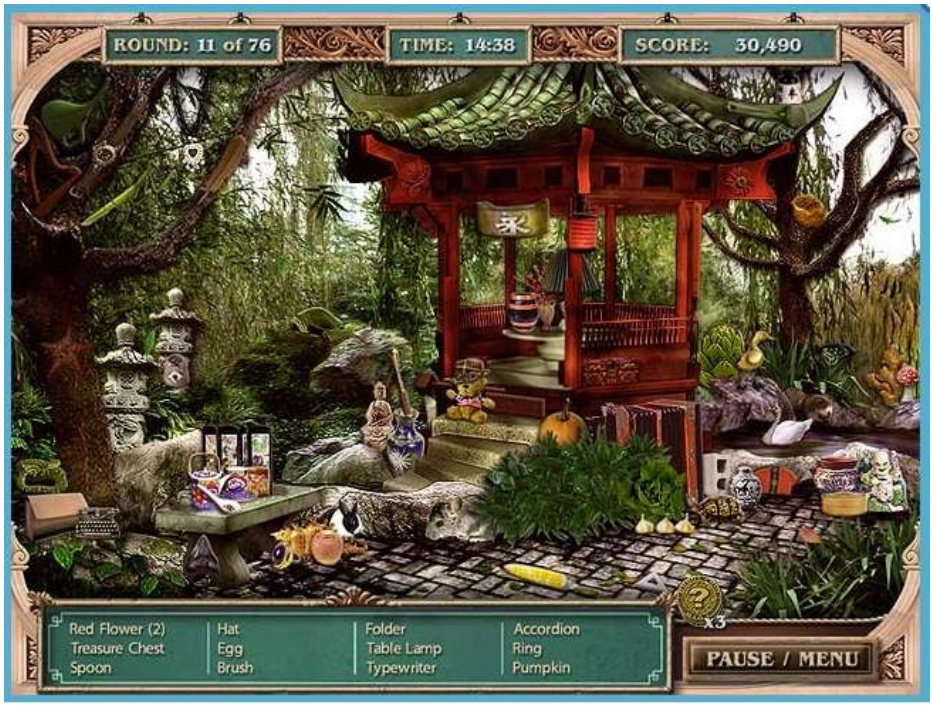


Gambar 3.1 Tampilan Game *Big City Adventure – San Francisco*

Gambar 3.1 adalah salah satu tampilan *game* yang ada pada *Big City Adventure – San Francisco* dimana pemain diminta menyusun 2 atau lebih objek yang sama.

2. Big City Adventure – Sidney

Game Big City Adventure – Sidney adalah *game* yang dikembangkan oleh *Big Fish Games,* Inc yang bermarkas di Seattle, Amerika. *Game* ini berburu harta karun dan mencari objek – objek yang tersembunyi diseluruh Kota Sidney.



Gambar 3.2 Tampilan Game *Big City Adventure – Sidney*

Gambar 3.2 adalah salah satu tampilan *game* yang ada pada *Big City Adventure – Sidney* dimana pemain diminta mencari gambar yang ada di dalam daftar.

Setelah melakukan observasi pada kedua *game* tersebut, maka didapat kesimpulan seperti yang bisa dilihat pada tabel 3.1 berikut :

Table 3.1 Analisis Game yang Sudah Ada

Nama Game	Jenis Game	Kelebihan	Kekurangan
Big City Adventure – Sidney	*Adventure*	Detail gambar sangat bagus, lokasi *game* luas, dan permainan sangat banyak.	Game berbayar, terlalu banyak pengulangan tipe *game* yang sama hanya beda tingkat kesulitan nya.
City Adventure – San Francisco	*Adventure*	Detail gambar sangat bagus, ada piala setiap menang, lokasi *game* luas, dan permainan sangat banyak.	Game berbayar, terlalu banyak pengulangan tipe *game* yang sama hanya beda tingkat kesulitan nya.

1.1.2 Analisis *game* yang akan dibangun

Game yang akan dibangun berupa *game adventure* yang dipadukan dengan mini *game*. *Game* ini dibangun sebagai media pembelajaran dan hiburan alternatif yang lebih interaktif dan menyenangkan. Dengan adanya jalan cerita, *gameplay adventure*, serta beragam *mini game* diharapkan dapat menjadi alternatif solusi dari permasalahan yang ada di *game* sejenis yang sudah ada.

1.1.2.1 Storyline

Pemain akan melakukan perjalanan mengunjungi beberapa landmark yang ada di Bandung, dimulai dari stasiun, lalu mengunjungi kawasan dago, gedung sate, braga, dan jembatan layang pasupati.

1. *Stage* 1

   Lokasi dari *stage* adalah stasiun bandung, dimana karakter utama akan muncul dan mendapat informasi sekilas tentang kota bandung.

2. *Stage* 2

Lokasinya berada di kawasan dago, dimana akan ada *game* acak gambar yang merupakan benda tradisional khas bandung. Pada *stage* ini pengguna akan diuji ketelitian nya dalam menemukan gambar yang dicari.

3. *Stage* 3

Lokasinya ada di gedung sate, dimana akan ada *game* mencari kuliner khas bandung, diamana pemain akan mencari makanan yang tersembunyi yang merupakan kuliner khas dari bandung.

4. *Stage* 4

Di *stage* ini pengguna akan memainkan *memory game*, yaitu mencari dua gambar yang sama.

5. *Stage* 5

Pada *stage* ini akan terdapat puzzle *game* yaitu *sliding puzzle* yang akan menyusun gambar Jembatan Pasupati.

1.1.2.2 Gameplay

Tugas utama pemain adalah mengunjungi semua lokasi dan memainkan mini *game* yang terdapat di setiap *landmark* yang di kunjungi.

1. Permainan *Game* Acak Gambar

Permainan *game* acak gambar adalah permainan diamana pemain diharuskan untuk menemukan gambar yang dicari. Misalnya pemain di minta untuk mencari gambar angklung, maka pemain harus menemukan gambar angklung tersebut.

2. Permainan Mencari Kuliner

Permainan mencari kuliner adalah permainan dimana pemain akan mencari makanan khas kota bandung yang tersembunyi di anatara barang-barang lainnya.

3. Permainan *Puzzle*

Permainana *puzzle* adalah permainan menyusun gambar, dimana pemain akan menyusun gambar yang telah di acak menjadi satu gambar

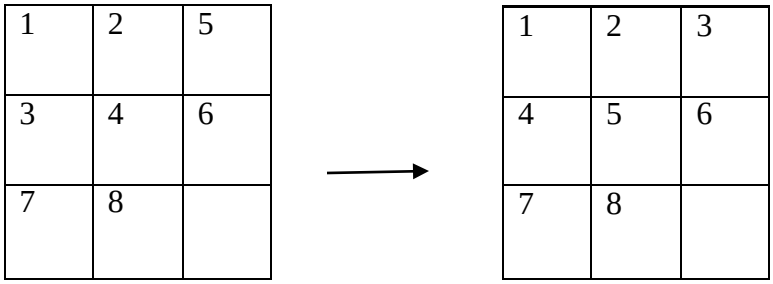
yang utuh. Tipe *game* puzzle yang dimainkan berupa *game sliding puzzle*.

4. Permainan *Memory Game*

Permainan *memory game* adalah permainan dimana pemain diminta untuk mencari dua gambar yang sama yang terdapat di dalam kotak.

1.1.3 Analisis Algoritma

Implementasi algoritma A* dengan ukuran *puzzle* 3x3 dapat dilihat pada gambar 3.3 berikut.



Gambar 3.3 Posisi Awal dan Posisi Akhir *Puzzle*

Pada gambar 3.3 diatas, ada 2 node yang mungkin menjadi node akar yaitu node 6 dan node 8. Posisi awal node 6 adalah (1,2) dan posisi akhirnya adalah (1,2). Posisi awal node 8 adalah (2,2) dan posisi akhirnya adalah (2,2). Untuk menentukan node mana yang akan digerakkan terlebih dahulu, maka digunakan fungsi evaluasi f, namun terlebih dahulu harus dihitung nilai h dari masing-masing node. Nilai g pada keadaan awal akan dianggap 0 namun saat perhitungan fungsi akan ditambahkan 1. Untuk nilai g seterusnya, akan diambil dari nilai g yang sudah dihitung dari fungsi pada node parent.

$h(2,2) = |2-2|+|2-2|= 0$

$h(1,2) = |1-2|+|1-2| = 0$

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

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

Nilai f(2,2) sama dengan f(1,2), maka yang akan dipindahkan adalah node 6. Hal ini ditentukan adanya node yang ditemukan lebih dulu nilai heuristiknya

1	2	5
3	4	
7	8	6

Gambar 3. 4 Posisi Node 6 Setelah Dipindahkan

Node 6 ini masih akan dimasukkan lagi pada *OPEN list* mengingat node ini belum mencapai tujuan. Selain itu, meskipun telah mencapai tujuan, node 6 masih mungkin akan dimasukkan lagi pada *OPEN list* jika node menempuh *path* yang berbeda dari *path* yang ditempuh sebelumnya. Selanjutnya, lakukan perhitungan nilai f untuk node 4 dan 5 dengan node *parent* merupakan node yang telah dipindahkan sebelumnya yaitu node 4. Node 4 terletak pada posisi awal (1,1) dan tujuannya adalah (1,0). Node 5 terletak pada posisi awal (0,2), dan tujuannya adalah (1,2).

$h(2,1) = |1-1|+|1-0| = 1$

$h(0,2) = |0-1|+|2-2| = 1$

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

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

Maka, node yang dipindahkan adalah node 4.

1	2	5
3		4
7	8	6

Gambar 3. 5 Posisi Node 4 Setelah Dipindahkan

Node-node selanjutnya dipindahkan menurut nilai f yang dihitung berdasarkan contoh diatas sehingga tercapai posisi akhir *puzzle*.

1.1.4 Spesifikasi Kebutuhan Perangkat Lunak (SKPL)

Spesifikasi kebutuhan perangkat lunak berisi deskripsi dari kebutuhan perangkat lunak yang akan dibangun. SKPL dari *game* edukasi *adventure* teori musik bisa dilihat pada tabel 3.2.

Table 3.2 Spesifikasi Kebutuhan Perangkat Lunak (SKPL)

Kode Kebutuhan	Kebutuhan
SKPL-F-1	Sistem dapat menampilkan materi
SKPL-F-2	Sistem dapat menampilkan *adventure game*
SKPL-F-3	Sistem dapat menampilkan menu *about*

1.1.5 Analisis Kebutuhan Non-Fungsional

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

Pada analisis kebutuhan sistem non-fungsional ini dijelaskan analisis kebutuhan perangkat lunak, analisis kebutuhan perangkat keras, dan analisis pengguna.

1.1.5.1 Analisis Kebutuhan Perangkat Keras

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.

1. Dari sisi pengembang

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

1. Prosesor core duo 1.8 GHz

2. Memori 1 GB
3. *Space* kosong di *harddisk* 3 GB
4. VGA card 512 MB
5. Monitor 1366 x 768
6. Mouse dan keyboard
7. Speaker

2. Dari sisi pemain

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

1. Prosesor dengan kecepatan 1.6 Ghz
2. Memori 256 Mb
3. *Space* kosong di *harddisk* 40 MB
4. VGA card
5. Monitor
6. Mouse dan keyboard
7. Speaker

1.1.5.2 Analisis Kebutuhan Perangkat Lunak

Perangkat keras komputer tidak berarti tanpa perangkat lunak, begitu juga sebaliknya. Jadi perangkat lunak dan perangkat keras saling mendukung satu sama lain. Perangkat keras hanya berfungsi jika diberikan instruksi - intruksi kepadanya. Instruksi - instruksi inilah yang disebut dengan perangkat lunak.

1. Dari sisi pengembang

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

1. Sistem operasi Windows XP SP3, Vista SP1, Windows 7, Windows 8, Mac OSX 10.5.7, Mac OSX 10.6.3
2. Adobe Flash CS5 sebagai *Integrated Development Environment* (IDE)-nya
3. Aplikasi pengolahan grafis menggunakan Adobe Illustrator CS5 dan Adobe Fireworks CS5

2. Dari sisi pemain

Untuk menjalankan *game* edukasi teori musik ini dibutuhkan perangkat lunak, antara lain:

1. Sistem operasi Windows, Mac OS, atau Linux
2. Flash Player 10

1.1.5.3 Analisis Pengguna

Analisis pengguna merupakan analisis terhadap pengguna yang akan menggunakan aplikasi yang telah dibangun. Perangkat keras dan perangkat lunak yang ada tidak akan berguna apabila tidak ada pengguna yang mengoperasikannya. Oleh karena itu dibuatlah analisis pengguna yang berdasarkan pada user *profile*nya, yakni berdasarkan *user knowledge & experience, user job, task & needs, user physicological characteristic*, dan *user physical characteristic* seperti yang bisa dilihat pada tabel 3.3.

Table 3.3 Analisis Pengguna

KNOWLEDGE AND EXPERIENCE	
Computer Literacy	Rendah, minimal bisa menggunakan computer
Pengalaman Sistem	Sedang, minimal biasa menggunakan sistem operasi Windows
Pengalaman Aplikasi	Sedang, minimal pernah memainkan *game adventure* sejenis
Penggunaan Sistem Lain	Jarang
Edukasi	SD
Kemampuan Membaca	Bisa membaca dengan lancer
Kemampuan Mengetik	"Hunt and Peck" (10 WPM).
Bahasa / Kultur	Bahasa Indonesia
JOB / TASK / NEED	

Frekuensi Penggunaan	*Occasional*
Tingkat Kepentingan *Task*	Lemah
Interaksi Sosial	Tidak diperlukan
PSYCHOLOGICAL CHARACTERISTICS	
Attitude	Positif
Motivasi	Tinggi
Stress Level	*No Stress*
PHYSICAL CHARACTERISTICS	
Umur	6 – 13 tahun
Jenis Kelamin	Laki – laki dan Perempuan
Handedness	Kanan, Kidal, dan Keduanya
Disabilities	Tidak ada masalah penglihatan.

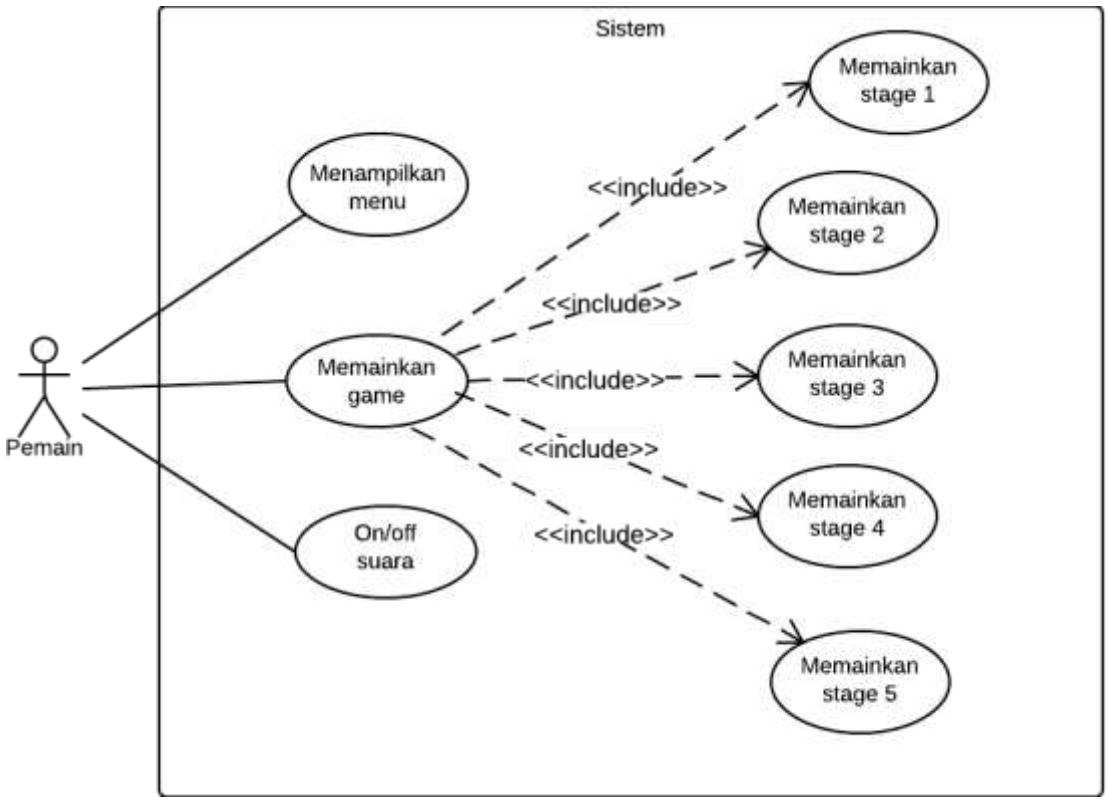
1.1.6 Analisis Kebutuhan Fungsional

Analisis kebutuhan fungsional menggambarkan proses kegiatan yang akan diterapkan dalam sebuah sistem dan menjelaskan kebutuhan yang diperlukan sistem agar sistem dapat berjalan dengan baik sesuai kebutuhan. Pada bagian ini akan dibahas tentang *use case diagram*, *activity diagram*, *class diagram*, dan *sequence diagram*.

1.1.6.1 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 3.6:



Gambar 3.6 Use Case Diagram

1. *Actor Definition*

Actor Definition berfungsi untuk menjelaskan *Actor* yang terdapat pada *Use Case Diagram*. *Actor Definition* diterangkan pada tabel 3.4.

Table 3.4 Actor Definition

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

2. *Use Case Definition*

Use Case Definition berfungsi untuk menjelaskan fungsi *Use Case* yang terdapat pada *Use Case Diagram*. *Use Case Definition* diterangkan pada tabel 3.5 di bawah ini.

Table 3.5 *Use Case Definition*

No	*Use Case*	Deskripsi
1	Memainkan *game*	Proses untuk memainkan *game*
2	Menampilkan menu *about*	Proses untuk menampilkan menu *about*
3	On/off suara	Proses untuk menghidupkan atau mematikan suara
4	Memainkan *stage* 1	Proses untuk melakukan permainan pada *stage* 1
5	Memainkan *stage* 2	Proses untuk melakukan permainan pada *stage* 2
6	Memainkan *stage* 3	Proses untuk melakukan permainan pada *stage* 3
7	Memainkan *stage* 4	Proses untuk melakukan permainan pada *stage* 4
8	Memainkan *stage* 5	Proses untuk melakukan permainan pada *stage* 5

3. *Use Case Descriptions*

Sebuah diagram yang menunjukkan *use case* dan aktor mungkin menjadi titik awal yang baik, tetapi tidak memberikan detail yang cukup untuk desainer sistem untuk benar-benar memahami persis bagaimana kebutuhan sistem akan dipenuhi. Cara terbaik untuk mengungkapkan informasi penting ini adalah dalam bentuk penjelasan berbasis teks.

a. *Use Case Description* Memainkan *Game*

Use Case Description dari *Use Case* Memainkan *Game* dijelaskan pada tabel 3.6 berikut :

Table 3.6 *Use Case Description* Memainkan *Game*

Nama Use Case	Memainkan *Game*	
Tujuan	Untuk memainkan permainan utama	
Kondisi Awal	Pemain berada di menu utama	
Kondisi Akhir	Permainan selesai dilakukan	
Pemicu	Pemain memilih menu mainkan	
Aliran Utama	Step	Aksi
	1. Include::Memainkan	Sistem menampilkan *stage* 1

	Stage 1	
	2. Include:: Memainkan *Stage* 2	Sistem menampilkan *stage* 2
	3. Include:: Memainkan *Stage* 3	Sistem menampilkan *stage* 3
	4. Include:: Memainkan *Stage* 4	Sistem menampilkan *stage* 4
	5. Include:: Memainkan *Stage* 5	Sistem menampilkan *stage* 5

b. *Use Case Description* Menampilkan Menu *About*

Use Case Description dari *Use Case* Menampilkan Menu *About* dijelaskan pada tabel 3.7 berikut :

Table 3.7 *Use Case Description* Menampilkan Menu *About*

Nama Use Case	Menampilkan Menu About	
Tujuan	Untuk menampilkan menu *about*	
Kondisi Awal	Pemain berada di menu utama	
Kondisi Akhir	Sistem menampilkan menu *about*	
Pemicu	Pemain memilih menu *about*	
Aliran Utama	Step	Aksi
	1.	Pemain memilih menu *about*
	2.	Sistem menampilkan menu *about*

***c. Use Case Description On/Off* Suara**

Use Case Description dari *Use Case* Memainkan *on/off* suara dijelaskan pada tabel 3.8 berikut :

Table 3.8 *Use Case Description* On/Off Suara

Nama Use Case	*On/off* suara

Tujuan	Untuk menghidupkan atau mematikan suara	
Kondisi Awal	Pemain berada di menu utama	
Kondisi Akhir	Sistem menghidupkan atau mematikan suara	
Pemicu	Pemain memilih *on/off* suara	
Aliran Utama	Step	Aksi
	1.	Pemain memilih menu *on/off* suara
	2.	Sistem mengidupkan atau mematikan suara
Ekstensi	Step	Aksi Cabang
	1.1	Suara ada
	1.2	Sistem mematikan atau menghidupkan suara

d. *Use Case Description* Memainkan *Stage* 1

Use Case Description dari *Use Case* Memainkan *Stage* 1 dijelaskan pada tabel 3.9 berikut :

Table 3.9 *Use Case Description* Memainkan *Stage* 1

Nama Use Case	Memainkan *Stage* 1	
Tujuan	Untuk melakukan permainan berupa *stage* 1	
Kondisi Awal	Pemain sudah berada di menu *game*	
Kondisi Akhir	Menampilkan *stage* 1	
Pemicu	Pemain menekan tombol *stage* 1	
Aliran Utama	Step	Aksi
	1.	Sistem men-stop suara di menu sebelumnya
	2.	Sistem menampilkan tampilan *stage* 1
	3.	Sistem menampilkan animasi petunjuk *stage*

e. *Use Case Description* Memainkan *Stage* 2

Use Case Description dari *Use Case* Memainkan *Stage* 2 dijelaskan pada tabel 3.10 berikut :

Table 3.10 *Use Case Description* Memainkan Stage 2

Nama Use Case	Memainkan *Stage* 2	
Tujuan	Untuk melakukan permainan berupa *stage* 2	
Kondisi Awal	Pemain sudah menyelesaikan *stage* 1	
Kondisi Akhir	Menampilkan *Stage* 2	
Pemicu	Pemain menekan tombol *stage* 2	
Aliran Utama	Step	Aksi
	1.	Sistem inisialisasi skor dan waktu
	2.	Sistem menampilkan *stage* 2
	3.	Sistem memulai waktu permainan
	4.	Pemain memilih gambar
	5.	Sistem mengecek jawaban
	6.	Sistem menampilkan hasil permainan
Ekstensi	Step	Aksi Cabang
	4.1	Jawaban salah
	4.2	Kurangi skor
	4.3	Jawaban benar
	4.4	Tambahkan skor

f. *Use Case Description* Memainkan *Stage* 3

Use Case Description dari *Use Case* Memainkan *Stage* 3 dijelaskan pada tabel 3.11 berikut :

Table 3.11 *Use Case Description* Memainkan *Stage* 3

Nama Use Case	Memainkan *Stage* 3	
Tujuan	Untuk melakukan permainan berupa *stage* 3	
Kondisi Awal	Pemain sudah menyelesaikan *stage* 2	
Kondisi Akhir	Menampilkan *stage* 3	
Pemicu	Pemain menekan tombol *stage* 3	
Aliran Utama	Step	Aksi

	1.	Sistem inisialisasi skor dan waktu
	2.	Sistem menampilkan *stage* 3
	3.	Sistem memulai waktu permainan
	4.	Pemain memilih gambar
	5.	Sistem mengecek jawaban
	6.	Sistem menampilkan hasil permainan
Ekstensi	Step	Aksi Cabang
	4.1	Jawaban salah
	4.2	Kurangi skor
	4.3	Jawaban benar
	4.4	Tambahkan skor

g. *Use Case Description* Memainkan *Stage* 4

Use Case Description dari *Use Case* Memainkan *Stage* 4 dijelaskan pada tabel 3.12 berikut :

Table 3.12 *Use Case Description* Memainkan Stage 4

Nama Use Case	Memainkan *Stage* 4	
Tujuan	Untuk melakukan permainan berupa *stage* 4	
Kondisi Awal	Pemain sudah menyelesaikan *stage* 3	
Kondisi Akhir	Menampilkan *stage* 4	
Pemicu	Pemain menekan tombol *stage* 4	
Aliran Utama	Step	Aksi
	1.	Sistem inisialisasi skor dan waktu
	2.	Sistem menampilkan *stage* 4
	3.	Sistem memulai waktu permainan
	4.	Pemain mencocokkan gambar
	5.	Sistem mengecek jawaban
	6.	Sistem menampilkan hasil permainan

Ekstensi	Step	Aksi Cabang
	4.1	Gambar sesuai, tambah skor
	4.2	Gambar tidak sesuai, kurangi skor

h. *Use Case Description* Memainkan *Stage* 5

Use Case Description dari *Use Case* Memainkan *Stage* 5 dijelaskan pada tabel 3.13 berikut :

Table 3.13 *Use Case Description* Memainkan *Stage* 5

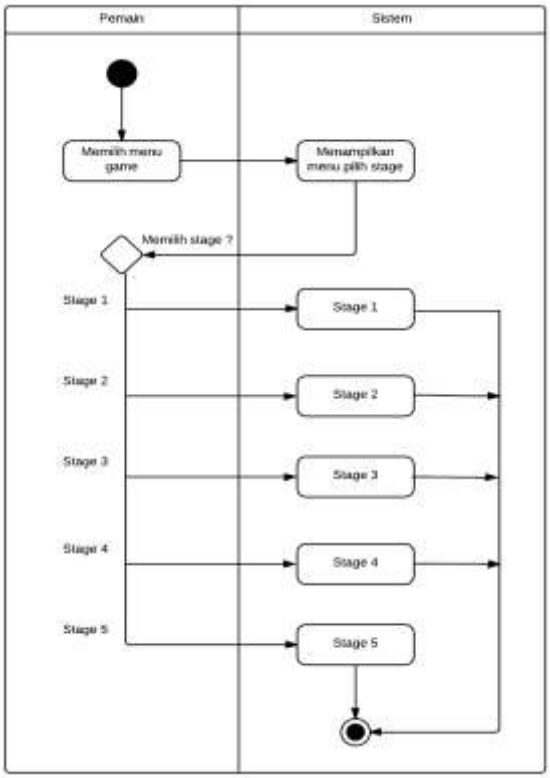
Nama Use Case	Memainkan *Stage* 5	
Tujuan	Untuk melakukan permainan berupa *stage* 5	
Kondisi Awal	Pemain sudah menyelesaikan *stage* 4	
Kondisi Akhir	Menampilkan *stage* 5	
Pemicu	Pemain menekan tombol *stage* 5	
Aliran Utama	Step	Aksi
	1.	Sistem inisialisasi skor dan waktu
	2.	Sistem menampilkan *stage* 5
	3.	Sistem memulai waktu permainan
	4.	Pemain menyusun puzzle
	5.	Sistem mengecek jawaban
	6.	Sistem menampilkan hasil permainan

1.1.6.2 Activity Diagram

Activity Diagram menunjukkan aktivitas oleh manusia atau objek dalam cara yang mirip dengan *Flow Chart*.

1. Activity Diagram* Memainkan *Game

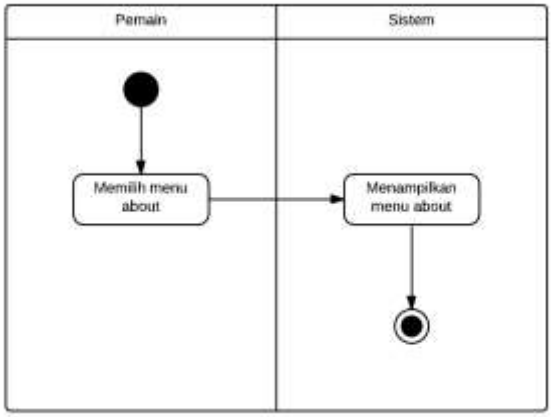
Berikut ini adalah diagram yang menunjukan alur aksi pada aktivitas Memainkan *Game* yang dapat dilihat pada gambar 3.7.



Gambar 3.7 Activity Diagram Memainkan Game

2. *Activity Diagram* Menampilkan Menu *About*

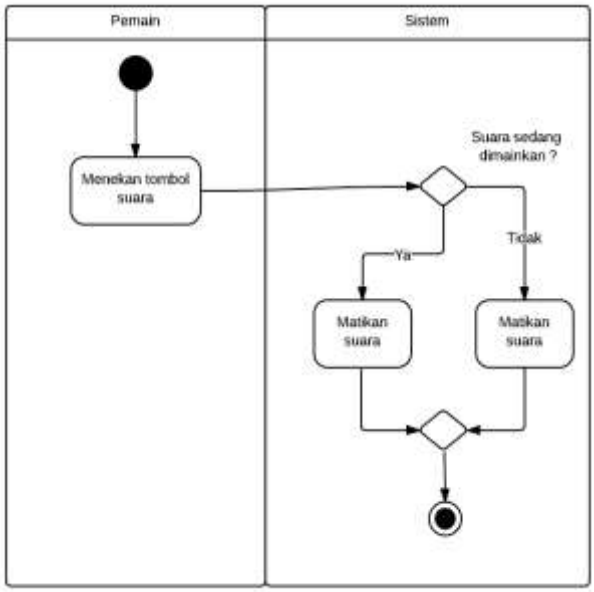
Berikut ini adalah diagram yang menunjukan alur aksi pada aktivitas Menampilkan Menu *About* yang dapat dilihat pada gambar 3.8.



Gambar 3.8 *Activity Diagram* Menampilkan Menu *About*

3. *Activity Diagram On/Off* Suara

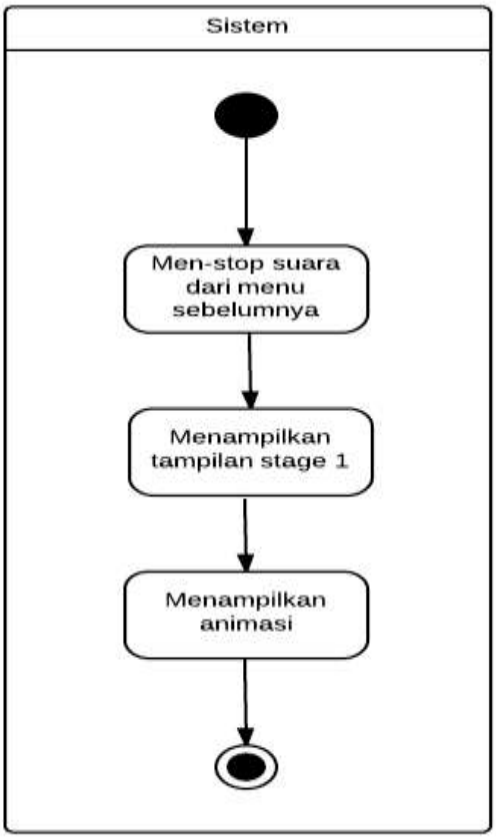
Berikut ini adalah diagram yang menunjukan alur aksi pada aktivitas Memainkan *Mini Game* yang dapat dilihat pada gambar 3.9.



Gambar 3.9 Activity Diagram On/Off Suara

4. *Activity Diagram* Memainkan *Stage* 1

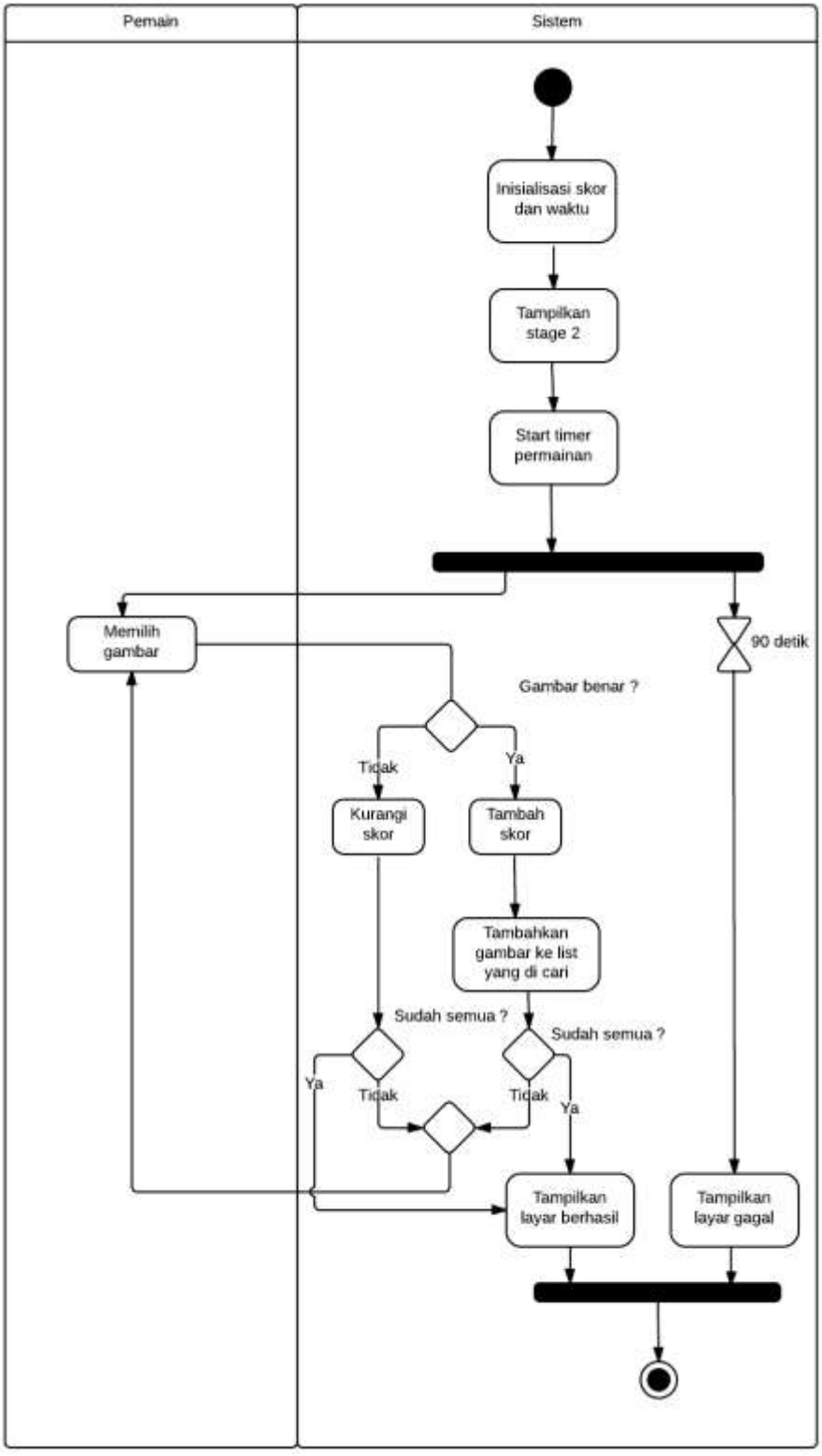
Berikut ini adalah diagram yang menunjukan alur aksi pada aktivitas Memainkan *Stage* 1 yang dapat dilihat pada gambar 3.10.



Gambar 3.10 Activity Diagram Memainkan Stage 1

5. *Activity Diagram* Memainkan *Stage* 2

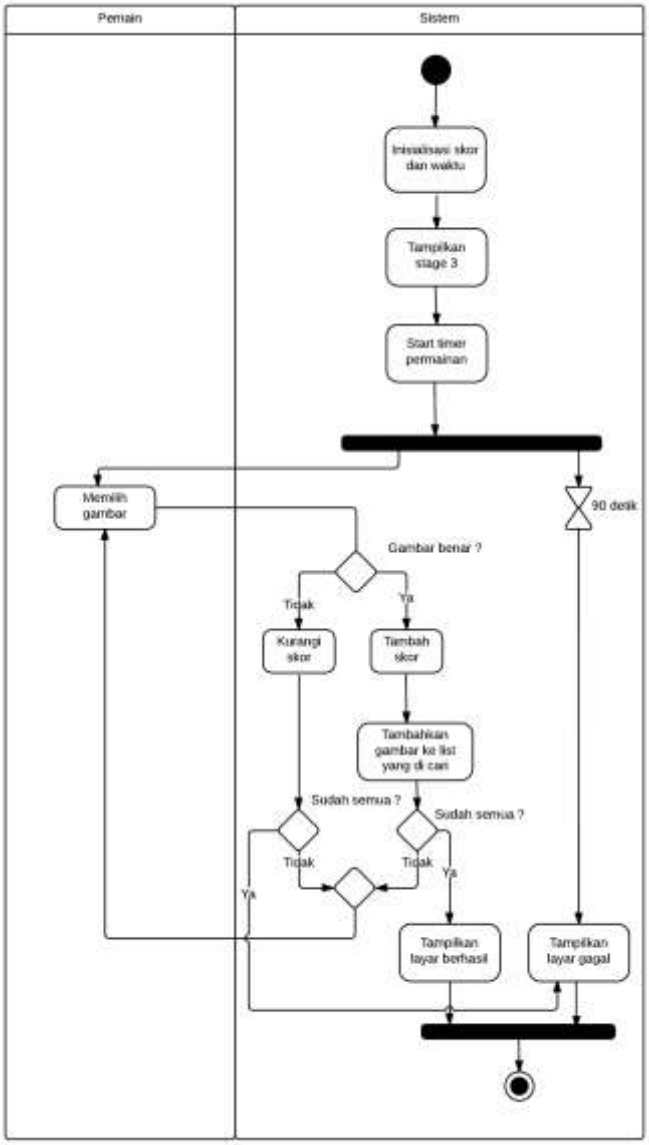
Berikut ini adalah diagram yang menunjukan alur aksi pada aktivitas Memainkan *Stage* 2 yang dapat dilihat pada gambar 3.11.



Gambar 3.11 Activity Diagram Memainkan *Stage* 2

6. *Activity Diagram* Memainkan *Stage* 3

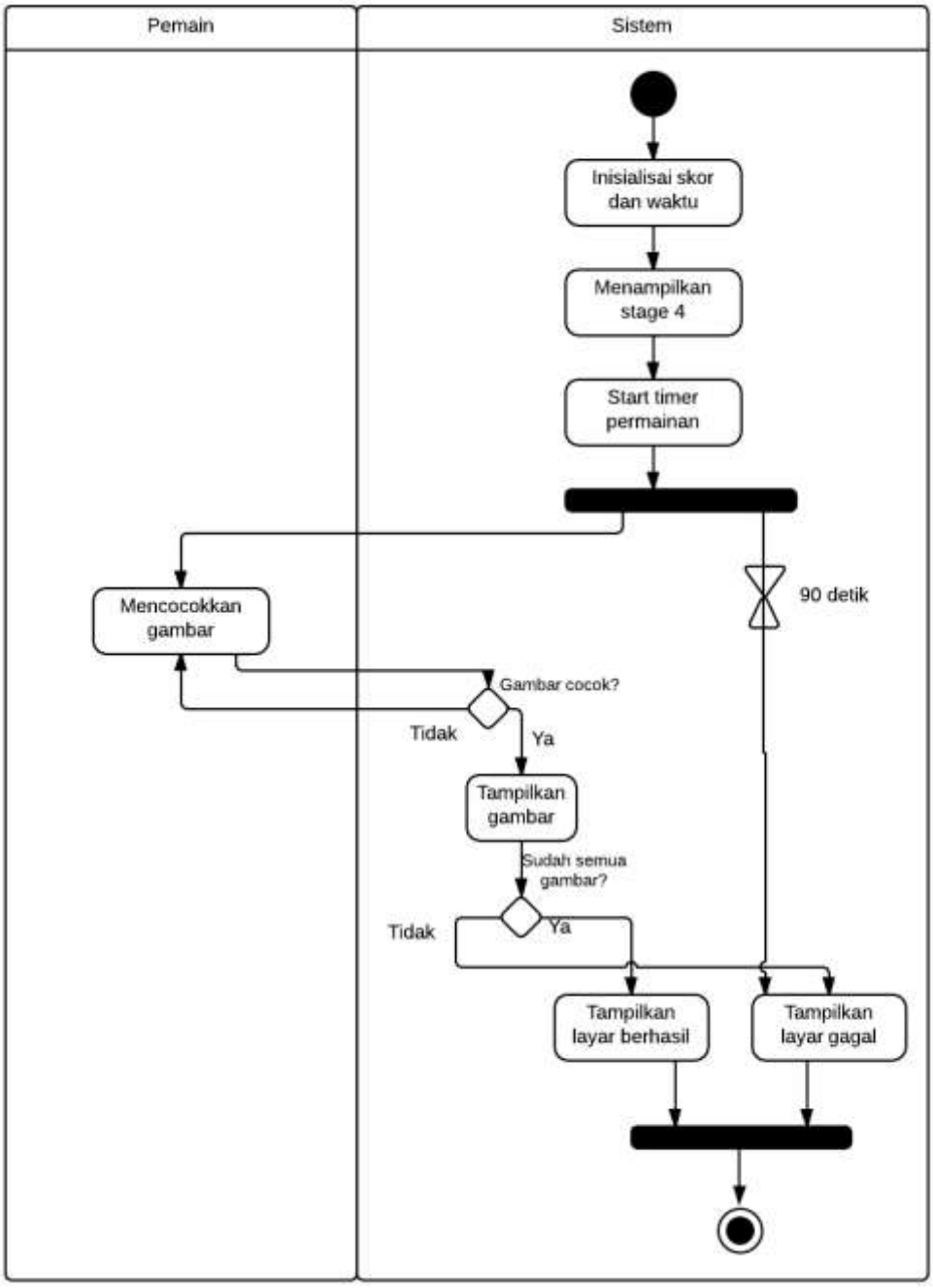
Berikut ini adalah diagram yang menunjukan alur aksi pada aktivitas Memainkan *Stage* 3 yang dapat dilihat pada gambar 3.12.



Gambar 3.12 *Activity Diagram* Memainkan *Stage* 3

7. *Activity Diagram* Memainkan *Stage* 4

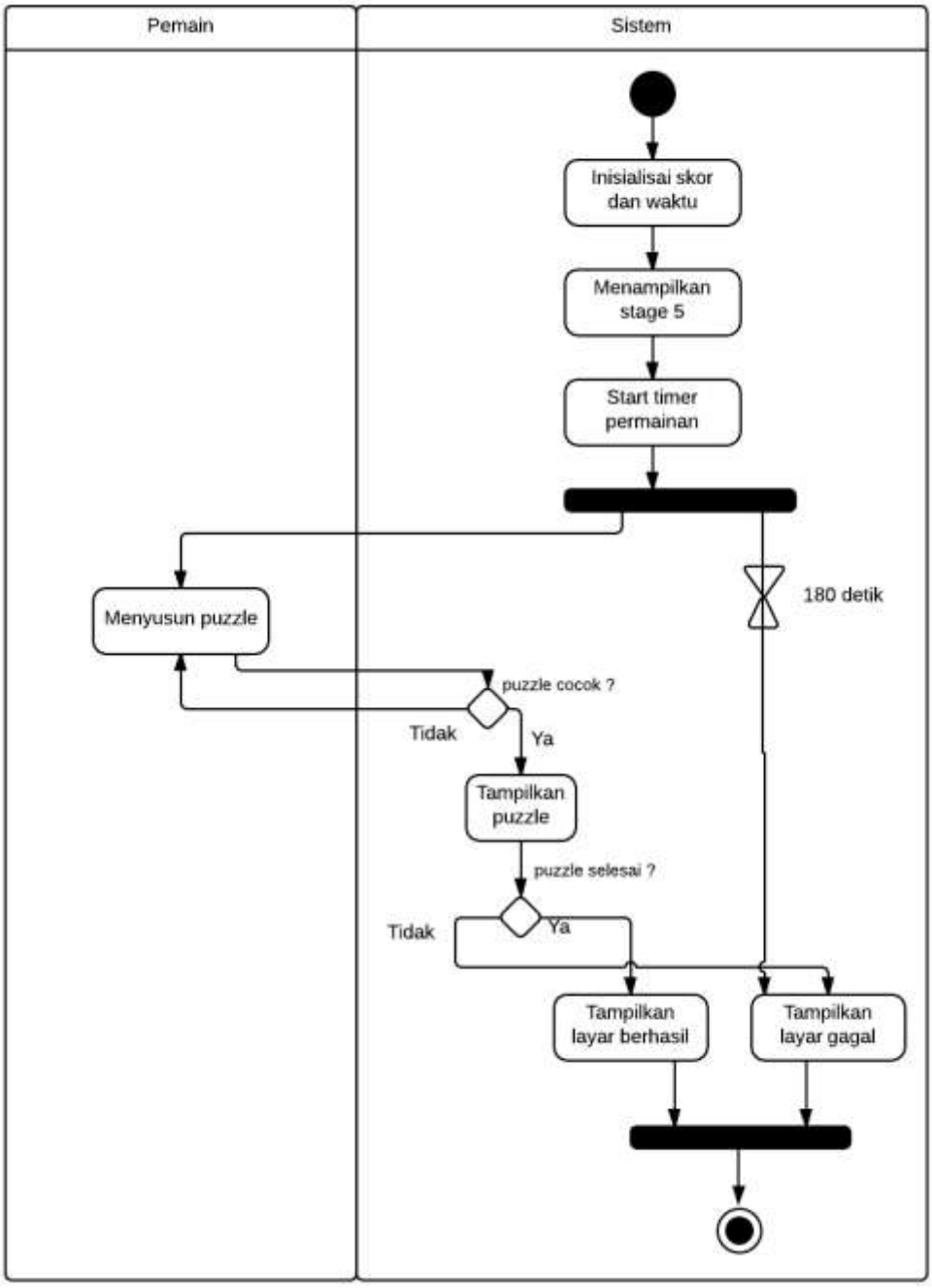
Berikut ini adalah diagram yang menunjukan alur aksi pada aktivitas Memainkan *Stage* 4 yang dapat dilihat pada gambar 3.13.



Gambar 3.13 Activity Diagram Memainkan Stage

8. ***Activity Diagram*** Memainkan *Stage* 5

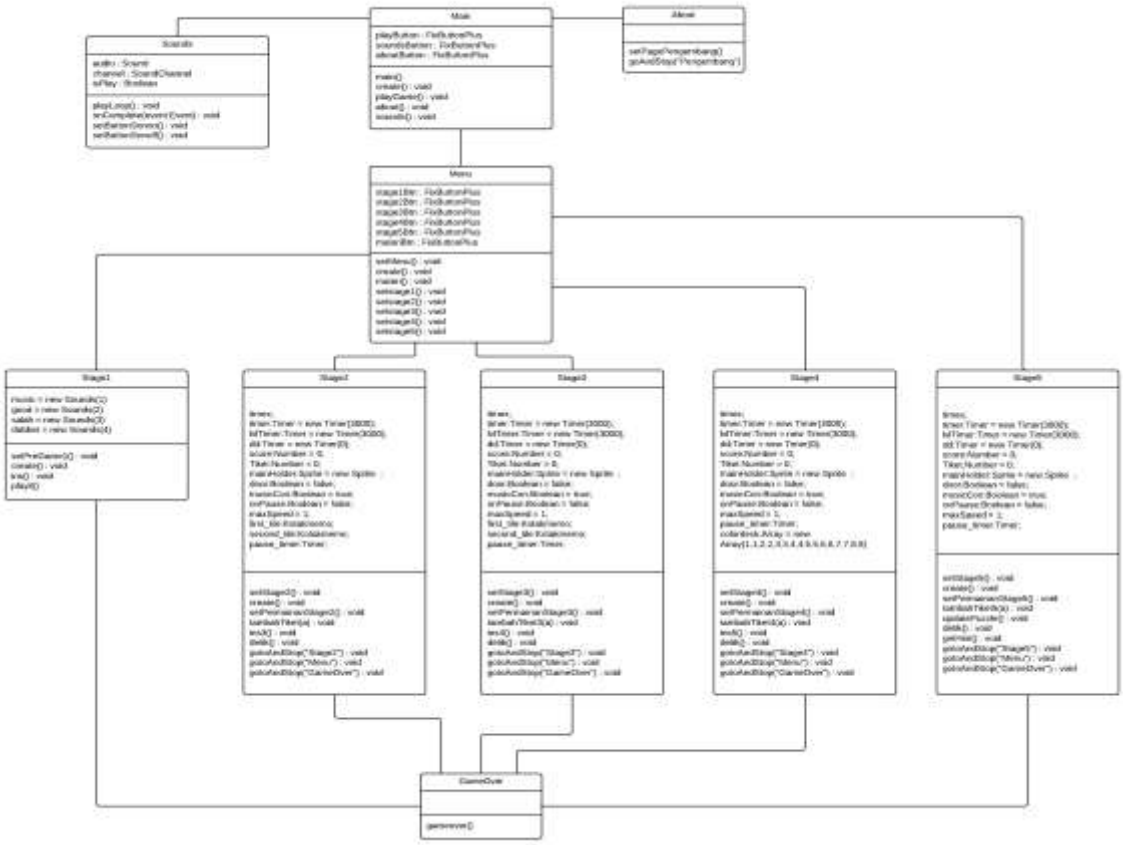
Berikut ini adalah diagram yang menunjukan alur aksi pada aktivitas Memainkan *Stage* 5 yang dapat dilihat pada gambar 3.14.



Gambar 3.14 Activity Diagram Memainkan Stage 5

1.1.6.3 Class Diagram

Class Diagram menunjukkan kelas dan bagaimana mereka dapat dipasang bersama-sama. Perancangan *class diagram* pada *game* ini dapat dilihat pada gambar 3.15 berikut



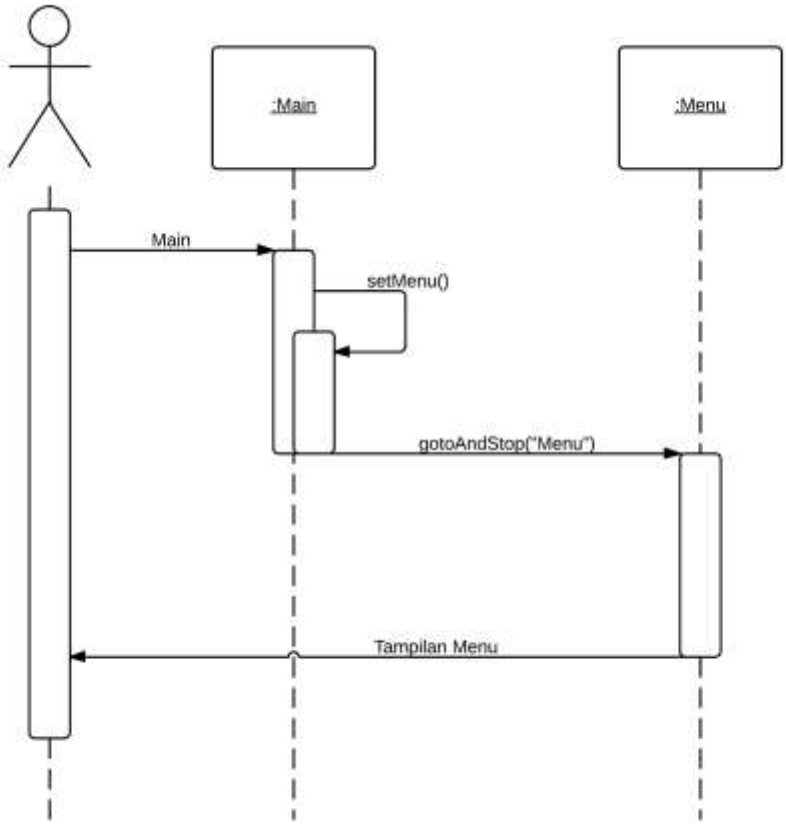
Gambar 3.15 Class Diagram

1.1.6.4 Sequence Diagram

Sequence diagram menggambarkan tindakan objek pada *use case* dengan mendeskripsikan waktu hidup objek dan *message* yang dikirimkan dan diterima antar objek.

1. *Sequence Diagram* Memainkan *Game*

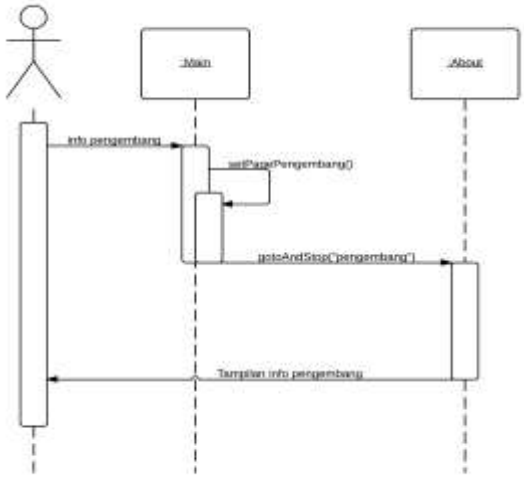
Diagram ini menjelaskan interaksi antar partisipan di dalam aktivitas Memainkan *Game* dalam urutan waktu. Untuk lebih jelasnya lihat gambar 3.16.



Gambar 3.16 *Sequence Diagram* Memainkan *Game*

2. *Sequence Diagram* Menampilkan Menu *About*

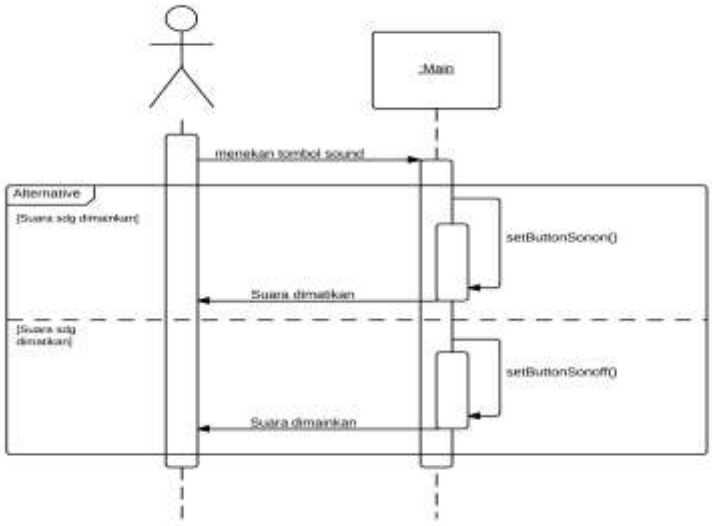
Diagram ini menjelaskan interaksi antar partisipan di dalam aktivitas Menampilkan Menu *About* dalam urutan waktu. Untuk lebih jelasnya lihat gambar 3.17.



Gambar 3.17 *Sequence Diagram* Menampilkan Menu *About*

3. *Sequence Diagram On/Off* Suara

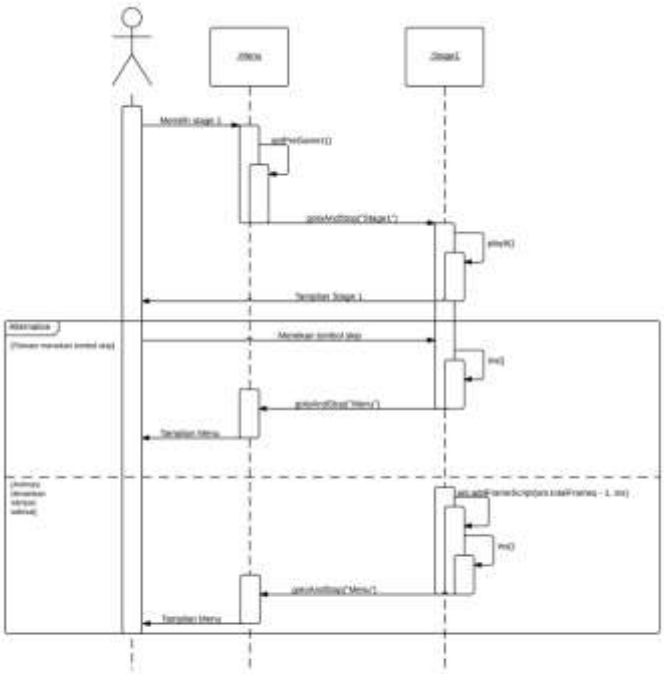
Diagram ini menjelaskan interaksi antar partisipan di dalam aktivitas *On/Off* Suara dalam urutan waktu. Untuk lebih jelasnya lihat gambar 3.18.



Gambar 3.18 Sequence Diagram On/Off Suara

4. *Sequence Diagram* Memainkan *Stage* 1

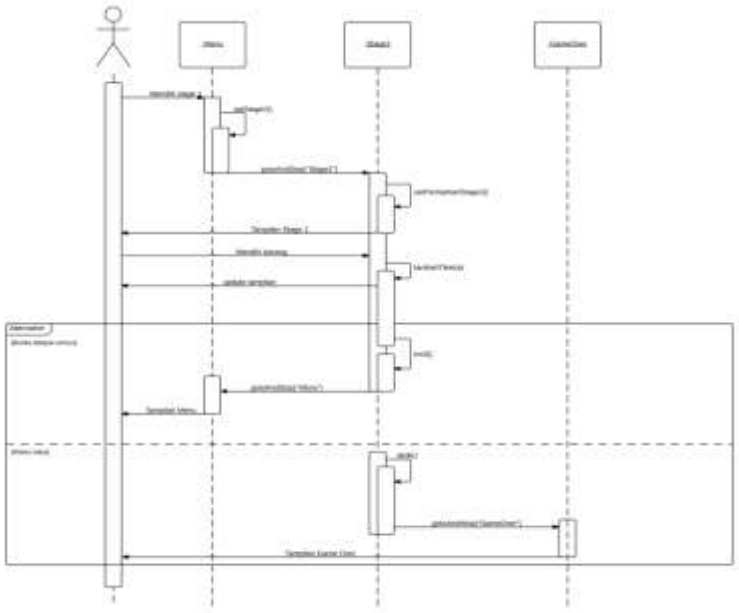
Diagram ini menjelaskan interaksi antar partisipan di dalam aktivitas Memainkan *Stage* 1 dalam urutan waktu. Untuk lebih jelasnya lihat gambar 3.19.



Gambar 3.19 *Sequence Diagram* Memainkan *Stage* 1

5. Sequence Diagram Memainkan *Stage* 2

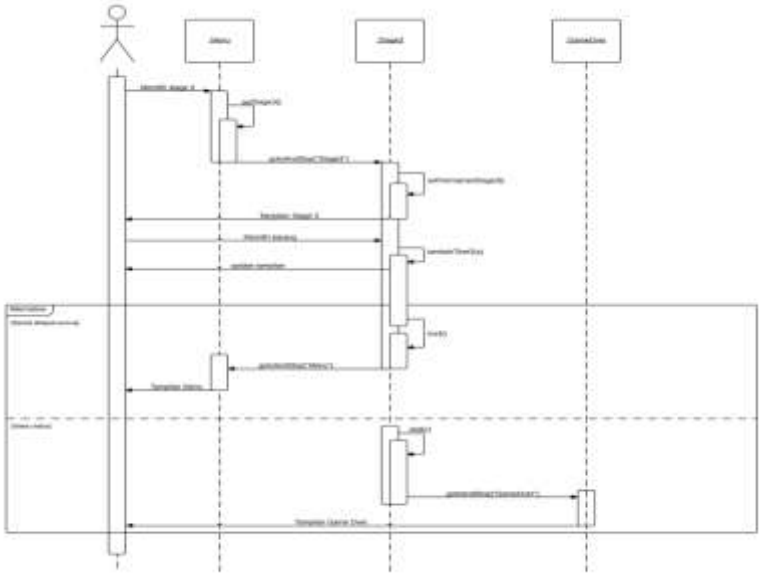
Diagram ini menjelaskan interaksi antar partisipan di dalam aktivitas Memainkan *Stage* 2 dalam urutan waktu. Untuk lebih jelasnya lihat gambar 3.20.



Gambar 3.20 Sequence Diagram Memainkan *Stage* 2

6. *Sequence Diagram* Memainkan *Stage* 3

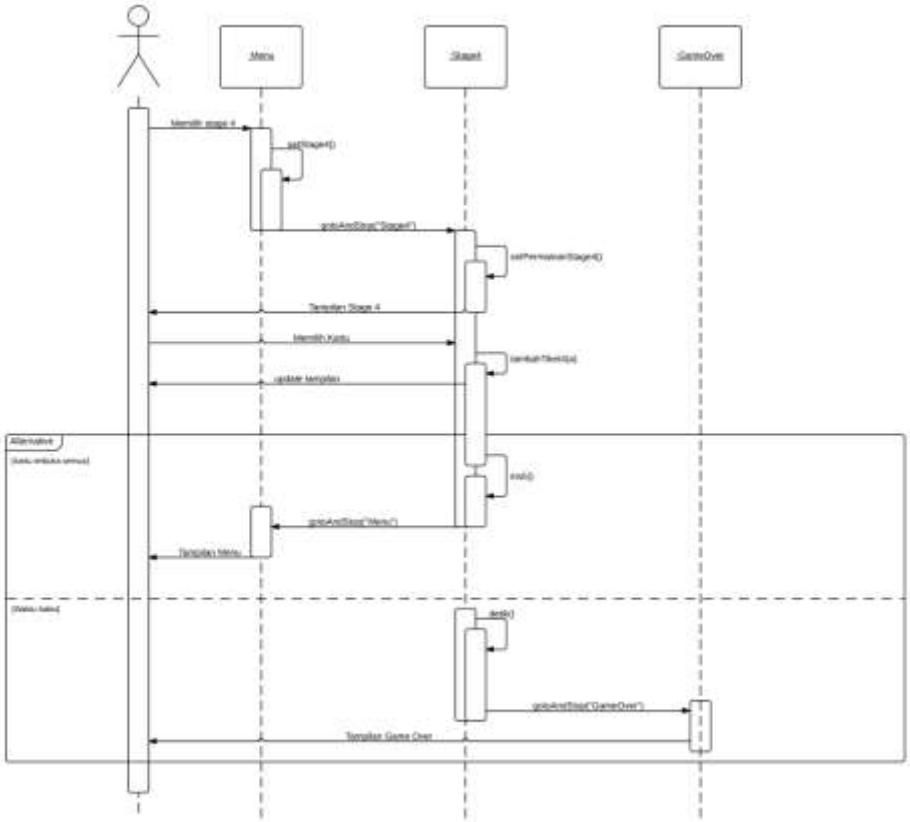
Diagram ini menjelaskan interaksi antar partisipan di dalam aktivitas Memainkan *Stage* 3 dalam urutan waktu. Untuk lebih jelasnya lihat gambar 3.21.



Gambar 3.21 Sequence Diagram Memainkan Stage 3

7. *Sequence Diagram* Memainkan *Stage* 4

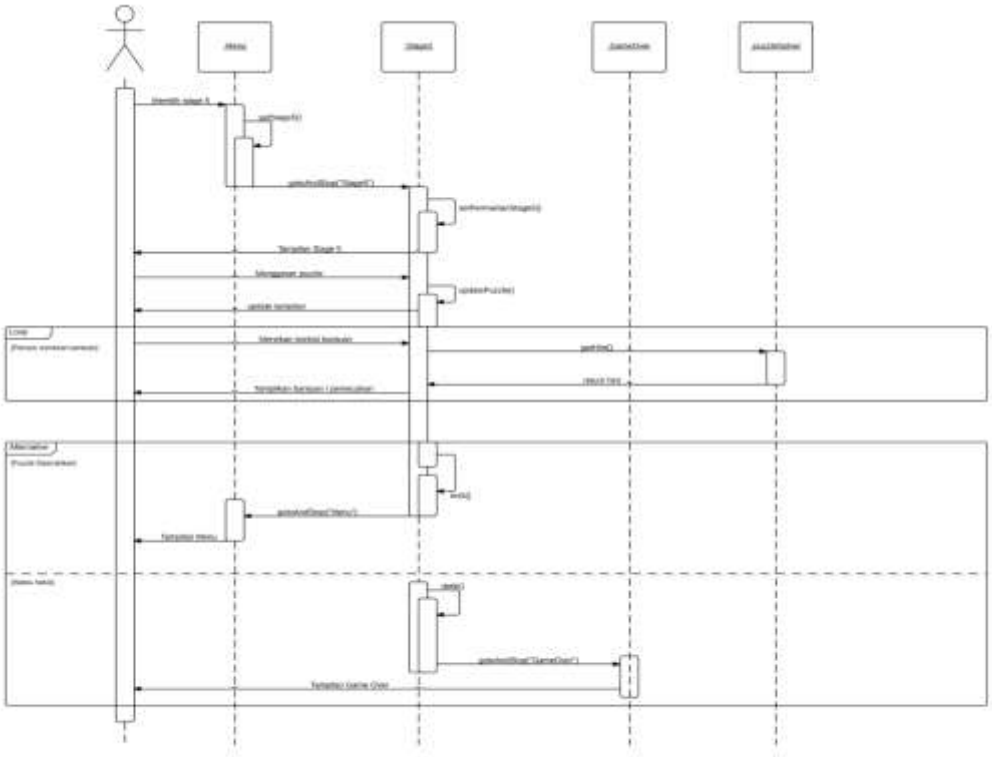
Diagram ini menjelaskan interaksi antar partisipan di dalam aktivitas Memainkan *Stage* 4 dalam urutan waktu. Untuk lebih jelasnya lihat gambar 3.22.



Gambar 3.22 Sequence Diagram Memainkan *Stage* 4

8. *Sequence Diagram* Memainkan *Stage* 5

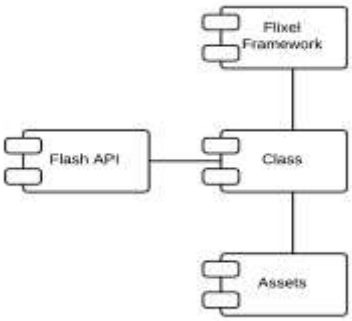
Diagram ini menjelaskan interaksi antar partisipan di dalam aktivitas Memainkan *Stage* 5 dalam urutan waktu. Untuk lebih jelasnya lihat gambar 3.23.



Gambar 3.23 *Sequence Diagram* Memainkan *Stage* 5

1.1.6.5 Component Diagram

Component Diagram menggambarkan struktur dan hubungan antar komponen piranti lunak termasuk ketergantungan (*dependency*). Komponen piranti lunak yang yang dimaksud adalah modul yang berisi *source code* atau *binary code*, yang ada *library* atau *excutetable* yang muncul pada *compile time, link time* ataupun pada *runtime*. *Component Diagram* pada *game* ini bisa dilihat pada gambar 3.24 berikut :



Gambar 3.24 Component Diagram

1.2 Perancangan Sistem

Di dalam bagian ini akan dijelaskan perancangan *storyboard*, antarmuka, perancangan pesan dan jaringan semantik dari *game* bandung *city adventure*.

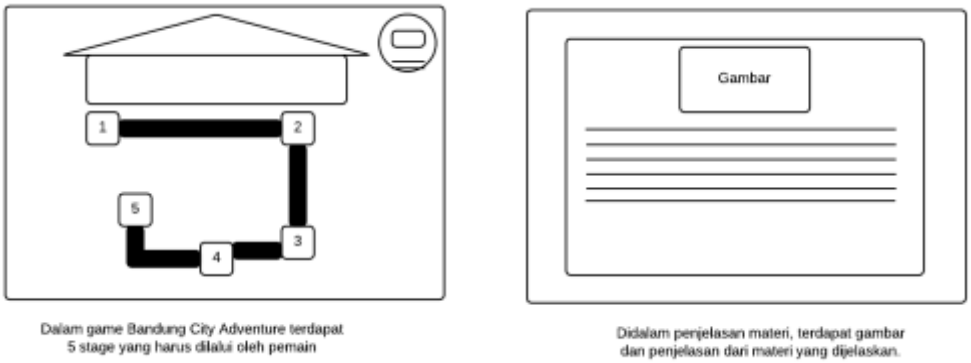
1.2.1 Storyboard

Storyboard adalah sketsa gambar yang disusun berurutan sesuai dengan naskah, dengan *storyboard* 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.

Pada bagian ini akan dijelaskan *storyboard game* dan pemberian materi.

1. *Storyboard game* dan pemberian materi

Berikut ini adalah *storyboard* dari *gameplay game* dan pemberian materi yang dapat dilihat pada gambar 3.25 :



Gambar 3.25 *Storyboard Game* dan Pemberian Materi

2. *Storyboard stage 1*

Berikut ini adalah *storyboard stage* 1 yang dapat dilihat pada gambar 3.26 :

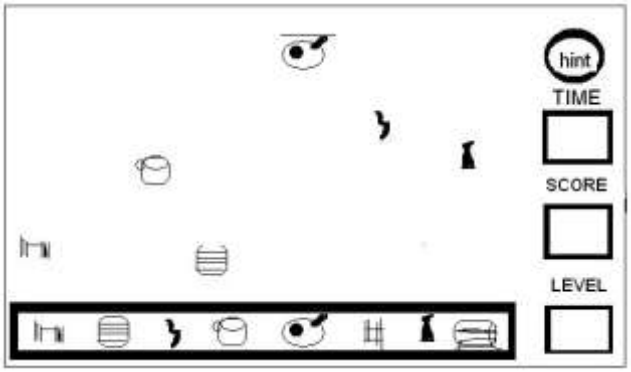


Gambar 3.26 *Storyboad Stage* 1

Pada stage 1 yang berlokasi di stasiun bandung akan diterangkan sekilas tentang sejarah kota bandung, dengan bantuan suara pembicara dan teks penjelasan.

3. *Storyboard stage 2*

Berikut ini adalah *storyboard stage* 1 yang dapat dilihat pada gambar 3.27 :

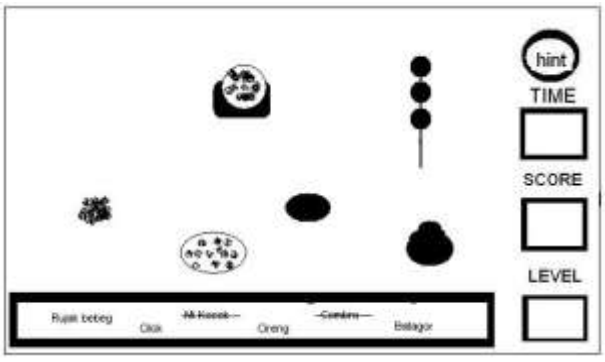


Gambar 3.27 *Storyboad Stage* 2

Pada *stage* 2 ini terdapat permainan mencari gambar alat tradisional khas bandung yang siluet nya terdapat dalam kotak barang yang dicari, pemain harus menemukan barang yang ada di daftar tersebut agar dapat menyelesaikan permainan.

4. *Storyboard stage 3*

Berikut ini adalah *storyboard stage* 1 yang dapat dilihat pada gambar 3.28 :



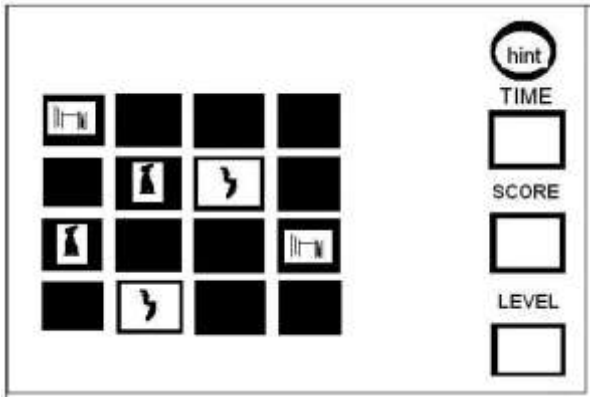
Gambar 3.28 *Storyboad Stage* 3

Pada *stage* 3 ini terdapat permainan mencari gambar kuliner tradisional khas bandung yang daftar makanan nya terdapat dalam kotak barang yang dicari, pemain harus memilih nama makanan terlebih dahulu agar bias menemukan makanan yang dicari,

pemain harus menemukan makanan yang ada di daftar tersebut agar dapat menyelesaikan permainan.

5. *Storyboard stage 4*

Berikut ini adalah *storyboard stage* 1 yang dapat dilihat pada gambar 3.29 :

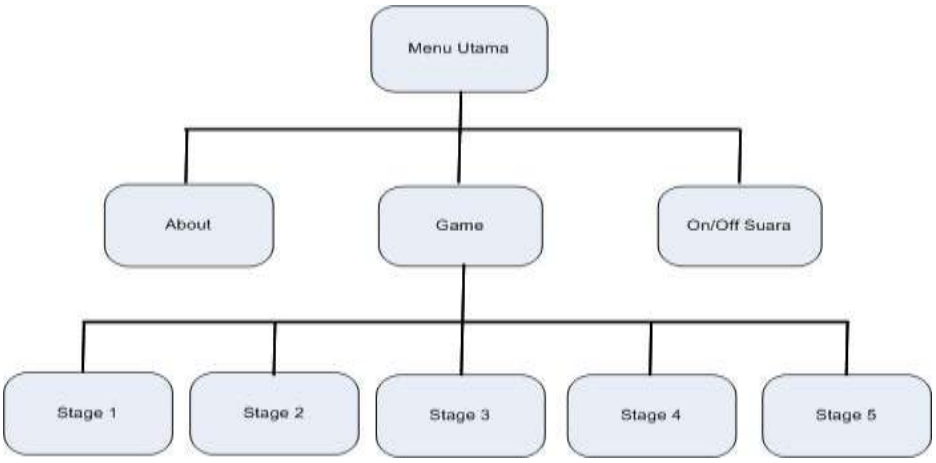


Gambar 3.29 *Storyboad Stage* 4

Pada *game stage* 4 ini terdapat permainan *memory game*, yaitu mencocokkan dua buah gambar yang sama, apabila pemain berhasil mencocokkan kedua gambar tersebut, maka skor akan bertambah, apabila salah maka skor akan berkurang. Pemain harus menyelesaikan semua gambar tersebut agar dapat menyelesaikan permainan.

1.2.2 Perancangan Struktur Menu

Perancangan struktur menu pada *game* edukasi *adventure* Bandung *City Adventure* pada gambar 3.30 berikut :



Gambar 3.30 Perancangan struktur menu

1.2.3 Perancangan Antarmuka

Perancangan antarmuka 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 antarmuka menu utama, perancangan antarmuka menu *about*, perancangan antarmuka pilih *stage*, perancangan antarmuka *stage* 1, perancangan antarmuka *stage* 2, perancangan antarmuka *stage* 2, perancangan antarmuka *stage* 3, perancangan antarmuka *stage* 4, dan perancangan antarmuka *stage* 5.

1.2.3.1 Perancangan Antarmuka Menu Utama

Menu utama merupakan menu yang akan muncul pertama kali pada saat pengguna membuka *game*. Gambar 3.31 menunjukan perancangan antarmuka Menu Utama.



Gambar 3.31 Perancangan Antarmuka Menu Utama

1.2.3.2 Perancangan Antarmuka *About*

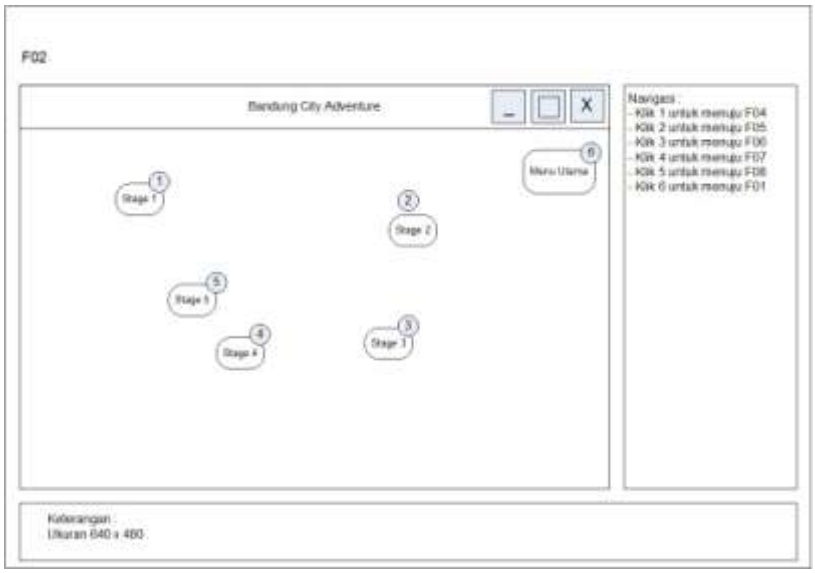
Menu *about* merupakan menu yang menampilkan profil penulis. Gambar 3.32 menunjukan perancangan antarmuka Menu *About*.



Gambar 3.32 Perancangan Antarmuka *About*

1.2.3.3 Perancangan Antarmuka Menu Pilih *Stage*

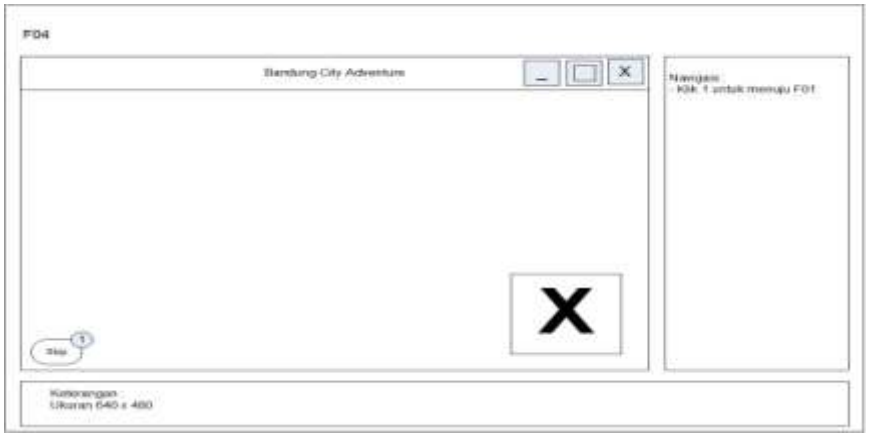
Menu *Mini Game* merupakan menu yang menampilkan pilihan *stage* yang akan dimainkan pengguna. Gambar 3.33 menunjukan perancangan antarmuka Menu Pilih *Stage*.



Gambar 3.33 Perancangan Antarmuka Menu Pilih *Stage*

1.2.3.4 Perancangan Antarmuka *Stage* 1

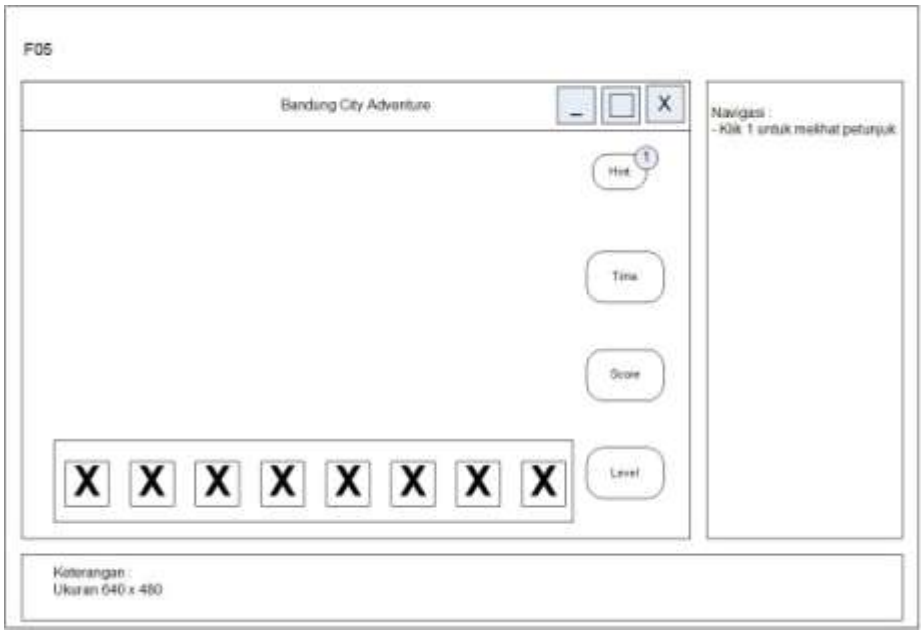
Pilih Level merupakan tampilan yang menampilkan *gameplay stage* 1 yang akan dimainkan pengguna. Gambar 3.34 menunjukan perancangan antarmuka *Stage* 1.



Gambar 3.34 Perancangan Antarmuka *Stage* 1

1.2.3.5 Perancangan Antarmuka *Stage* 2

Antarmuka *Stage* 2 merupakan tampilan yang menampilkan *gameplay stage* 2. Gambar 3.35 menunjukan perancangan antarmuka *Stage* 2.



Gambar 3.35 Perancangan Antarmuka *Stage* 2

1.2.3.6 Perancangan Antarmuka *Stage* 3

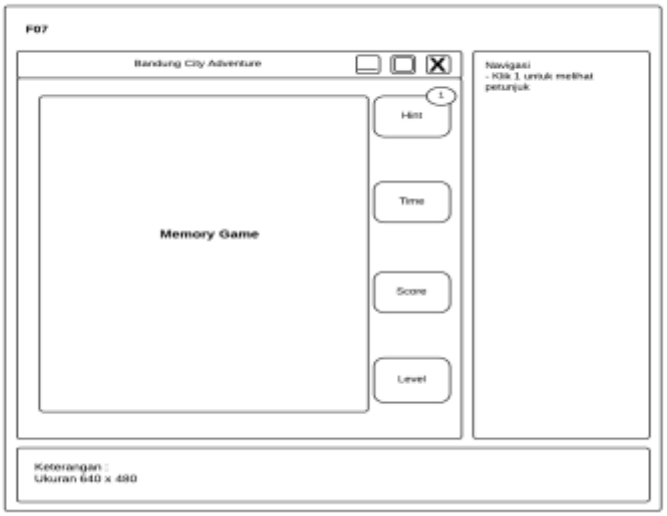
Antarmuka *Stage* 3 merupakan tampilan yang menampilkan *gameplay stage* 3. Gambar 3.36 menunjukan perancangan antarmuka *Stage* 3.



Gambar 3.36 Perancangan Antarmuka *Stage* 3

1.2.3.7 Perancangan Antarmuka *Stage* 4

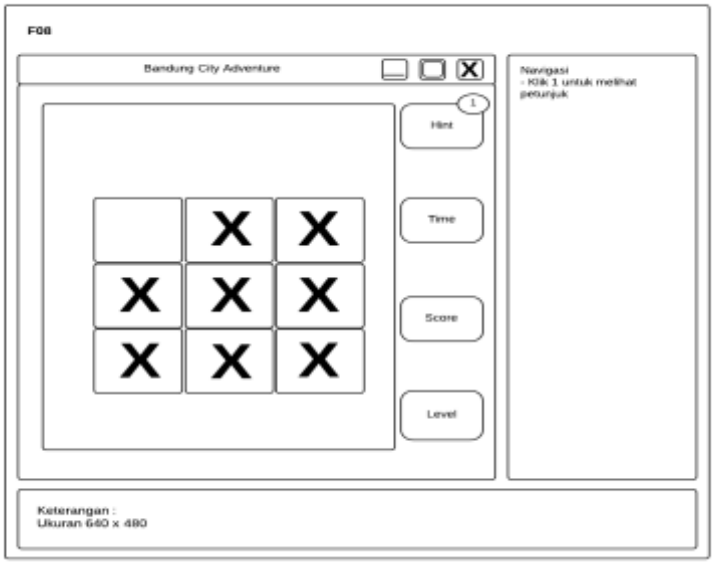
Antarmuka *Stage* 4 merupakan tampilan yang menampilkan *gameplay stage* 4 Gambar 3.37 menunjukan perancangan antarmuka *Stage* 4



Gambar 3.37 Perancangan Antarmuka *Stage* 4

1.2.3.8 Perancangan Antarmuka *Stage* 5

Antarmuka *Stage* 5 merupakan tampilan yang menampilkan *gameplay stage* 5 Gambar 3.38 menunjukan perancangan antarmuka *Stage* 5.



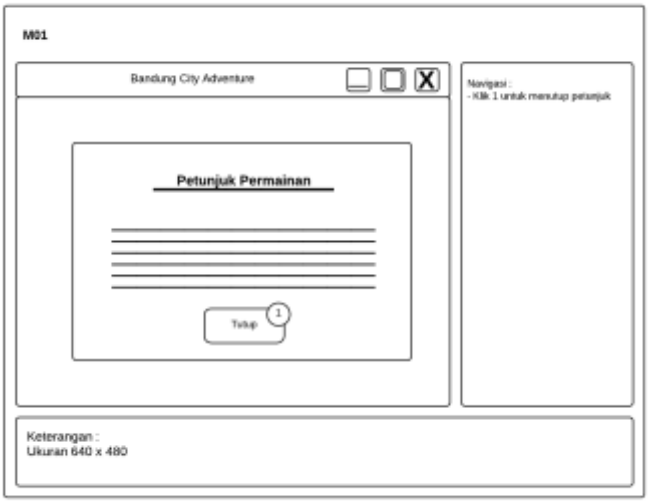
Gambar 3.38 Perancangan Antarmuka *Stage* 5

1.2.4 Perancangan Pesan

Pada bagian ini akan dijelaskan perancangan pesan yang terdapat pada *game* edukasi bandung *city adventure* yaitu perancangan pesan petunjuk, perancangan pesan misi gagal, perancangan pesan misi berhasil, dan perancangan pesan *stage* selesai.

1.2.4.1 Perancangan Pesan Petunjuk *Game*

Pesan Petunjuk *Game* merupakan pesan yang muncul ketika pengguna menampilkan petunjuk *game*. Untuk lebih jelasnya dapat dilihat pada gambar 3.39.



Gambar 3.39 Perancangan Pesan Petunjuk *Game*

1.2.4.2 Perancangan Pesan Misi Gagal

Pesan Misi Gagal merupakan pesan yang mucul ketika pemain gagal menyelesaikan misi . Untuk lebih jelasnya dapat dilihat pada gambar 3.40.



Gambar 3.40 Perancangan Pesan Misi Gagal

1.2.4.3 Perancangan Pesan Misi Berhasil

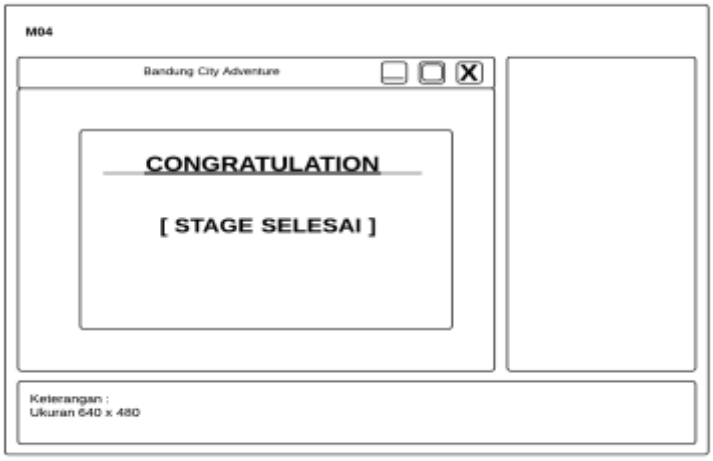
Pesan Misi Berhasil merupakan pesan yang mucul ketika pemain berhasil menyelesaikan misi. Untuk lebih jelasnya dapat dilihat pada gambar 3.41.



Gambar 3.41 Perancangan Pesan Misi Berhasil

1.2.4.4 Perancangan Pesan *Stage* Selesai

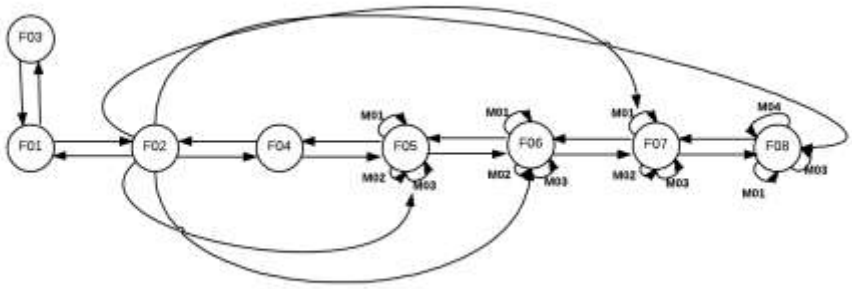
Pesan *Stage* Selesai merupakan pesan yang muncul ketika pengguna selesai semua *stage*. Untuk lebih jelasnya dapat dilihat pada gambar 3.42.



Gambar 3.42 Perancangan Pesan *Stage* Selesai

1.2.5 Jaringan Semantik

Untuk mengetahui keterhubungan dari setiap tampilan yang dibuat dapat menggunakan jaringan semantik. Gambar 3.43 berikut adalah jaringan semantik pada *game* edukasi *adventure* teori musik.



Gambar 3.43 Jaringan Semantik