BAB 3

ANALISIS DAN PERANCANGAN SISTEM

3.1 Analisis Sistem

Analisis merupakan penguraian dari suatu sistem informasi yang utuh ke dalam bagian-bagian komponennya dengan maksud untuk mengidentifikasikan dan mengevaluasi segala macam permasalahan, hambatan apa saja yang bisa terjadi dan kebutuhan-kebutuhan apa saja yang diharapkan dapat menjadi acuan untuk diusulkannya perbaikan-perbaikan.

3.1.1 Analisis Masalah

Pembelajaran dan pengenalan kebudayaan melalui permainan merupakan proses pembelajaran yang lebih menekankan pada penyajian bentuk permainan dengan muatan pentingnya mengenali dan mempelajari kebudayaan Indonesia.

Game yang diangkat dari kebudayaan Indonesia saat ini hanya beberapa saja yang sudah dibuat dalam media elektronik. Akan sangat menarik apabila permainan-permainan tradisional yang ada di Indonesia ini diimplementasikan menjadi permainan pada media elektronik seperti komputer. Seperti yang kita ketahui *game* merupakan aplikasi yang paling banyak digunakan dan dinikmati oleh pengguna media elektronik saat ini.

Dengan adanya cabang ilmu kecerdasan buatan, pada permainan ini membuat permainan ini dapat dimainkan dalam komputer, sehingga pemain dapat

terhibur tanpa harus mencari lawan bermain, karena komputer dapat bertindak sebagai pemain lawan, yang dapat berfikir untuk menjadi lawan main.

3.1.2 Analisis *Game* Sejenis

Analisis *game* sejenis dilakukan untuk memberikan gambaran system agar lebih mudah dalam pembangunan dan sebagai pembanding untuk kesempurnaan *game* yang akan dicapai. *Game* yang akan dibahas dan menjadi acuan disini adalah *game* 41 – empat satu.

3.1.2.1 *Game* 41 – empat satu

Empat satu merupakan salah satu jenis permainan kartu yang ada di Indonesia yang menggunakan kartu remi standar sebanyak 52 buah kartu dalam permainan. Kartu As bernilai 11, kartu bergambar (Raja, Ratu, Jack) dan sepuluh bernilai masing-masing 10, dan kartu dari angka 2 sampai 9 bernilai sama dengan angkanya. Tujuan dari permainan ini adalah mengumpulkan empat buah kartu dengan corak atau kembang yang sama dengan jumlah nilai semaksimal mungkin. Nilai 41 adalah nilai tertinggi yang bias dikumpulkan.



Gambar 3.1 Tampilan menu utama *game* 41 empat satu

A. *Gameplay*

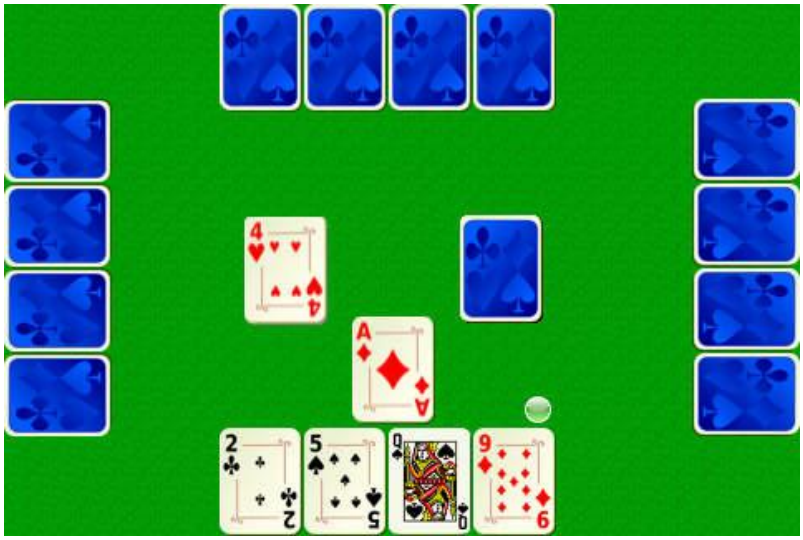
Pada awal permainan, masing-masing pemain akan dibagikan kartu sebanyak empat kartu. Setiap giliran pemain akan mengambil satu kartu boleh dari tumpukan tengah atau kartu buangan dari pemain sebelum gilirannya, dan membuang satu kartu. Pada giliran pertama, pemain yang pertama kali main hanya bisa mengambil kartu dari tumpukan tengah, karena belum ada buangan dari pemain sebelumya. Kartu yang bisa diambil dari buangan hanya kartu terakhir yang dibuang, atau kartu buangan yang paling atas.

Pemain boleh mengambil kartu buangan pemain sebelumnya kemudian membuang kartu yang sama. Seperti halnya jika pemain mengambil kartu dari tumpukan tengah, pemain boleh membuang kartu yang sama.

Jika pada awal permainan pemain yang mendapat giliran pertama mendapatkan 4 kartu bermotif sama, pemain bisa langsung menghentikan permainan. Permainan berhenti dan masing-masing pemain menunjukkan kartunya. Jika beruntung, pemain bisa mengandalkan 4 kartu yang sama tersebut untuk menang, karena pemain yang lain belum sempat mengganti kartu yang ada di tangan mereka.

Jika setelah menarik dan membuang kartu pemain mendapatkan 4 kartu yang bermotif sama, pemain tidak dapat menghentikan permainan, pemain harus menunggu sampai giliran selanjutnya.

Jika sudah tidak ada lagi kartu yang bisa diambil dari tumpukan tengah, dan pemain tidak mau mengambil kartu dari buangan sebelumnya, maka permainan berakhir dan kartu yang ditangan akan dihitung jumlahnya.



Gambar 3.2 Tampilan *game* 41 empat satu

B. *Scoring*

Saat permainan berakhir, semua pemain menunjukkan kartu masing-masing dan menghitung nilainya. Masing-masing pemain hanya bisa menghitung satu motif kartu untuk nilai positif, dan kartu lain yang bermotif berbeda akan bernilai negatif. Contoh :

- Sekop (raja) – sekop (3) – hati (9) – hati (8), akan bernilai 4, yaitu 9+8-10-3.
- Keriting (as) – wajik (2) – wajik (3) – wajik (5), akan bernilai 1, yaitu 11-2-3-5.

Nilai sempurna adalah 41, misalnya jika kartu ditangan pemain memiliki A-K-J-10 yang bermotif sama.

Pemain dengan nilai tertinggi adalah pemenang, dan yang memiliki nilai terendah kalah. Dan pemain yang menang akan pendapat giliran pertama pada permainan selanjutnya.

Jika nilai akhir pemain ada yang sama, maka kartu terendah akan dijadikan penentu menang atau kalah. Contoh:

- J–10–7–6 yang bermotif sama bernilai lebih tinggi dari A–K–7–5. Keduanya bernilai 33, tapi 6 lebih tinggi dari 5.
- J–10–7 hati dan 4 keriting bernilai lebih tinggi dari A–K–8 sekop dan 6 wajik. Keduanya bernilai 23, tapi 4 keriting lebih tinggi dari 6 wajik, karena dihitung bernilai negatif.

3.1.2.2 Aplikasi Permainan Capsah Banting

Capsah banting adalah salah satu permainan kartu yang cukup terkenal di kalangan remaja maupun orang dewasa di Indonesia. Permainan kartu ini diciptakan di daratan Cina sekitar tahun 1980, dan menjadi sangat terkenal di Shanghai, Hongkong, Taiwan, Filipina dan Singapura dan juga menyebar ke daerah barat. Permainan Capsah ini mempunyai beberapa nama, di China permainan ini dikenal dengan nama *Choi Dai Di* atau *Bu Bu Gao Sheng.* Di Amerika dan Australia dikenal dengan nama *Big Two, Big Deuce,* atau *Deuce.*

A. *Gameplay*

Permainan kartu Capsah Banting ini menggunakan kartu remi yang berjumlah 52 kartu. Urutan rangking-nya adalah 3,4,5,6,7,8,9,10,J,Q,K,2 (2 adalah nilai terbesar). Sedangkan urutan gambarnya dari kecil ke

besar adalah *diamond* ♦ (wajik), *club* ♣ (keriting), *heart* ♥ (hati), *spade* ♠ (sekop).

Kartu dibagikan kepada empat pemain secara acak. Pemain yang mendapatkan giliran pertama adalah pemain yang mempunyai kartu 3 *diamonds*. Kemudian pemain lain harus mengeluarkan kartu yang lebih besar. Pemain yang kartunya habis terlebih dahulu dialah yang menang.

B. *Scoring*

Urutan nilai kartu dari yang terkecil hingga yang terbesar adalah sebagai berikut :

- *One card hands / single* : kartu apapun yang berjumlah 1 buat
- *Two card hands / pair* : sepasang kartu yang memiliki rangking (angka) yang sama. Contoh : K ♥ - K ♦
- T*hree card hands / tris* : tiga buah kartu dengan rangkin yang sama. Contoh : 8♥ - 8♠ - 8♣
- *Five cards hands in order :*
    - *Straight* : kartu yang memiliki ranking berurutan.

      Contoh : K♥ - Q♠ - J♦ - 10♣ - 9♥
    - *Flush* : lima buah kartu dengan gambar yang sama.

      Contoh : 9♥ - 7♥ - 6♥ 5♥ - 3♥
    - *Full House* : 2 kartu dengan ranking yang sama dan 3 kartu lain dengan ranking yang sama pula.

      Contoh : 8♥ - 8♠ - 8♣ - K♥ - K♦

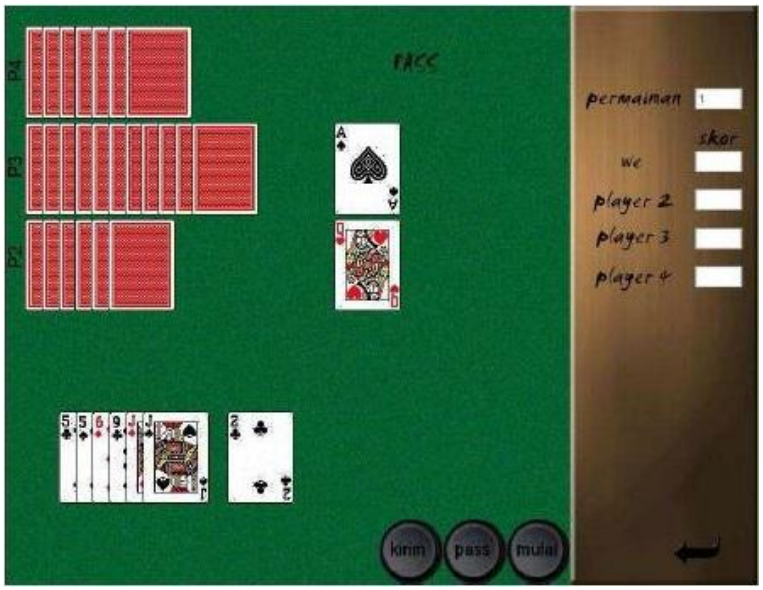
- *Four of a kind / four / boom* : 4 kartu dengan ranking yang sama

  Contoh : 9♦ - 9♣ - 9♥ - 9♠

- *Straight Flush* : 5 buah kartu yang rankingnya berurutan dan memiliki gambar yang sama.

  Contoh : A♠ - K♠ - Q♠ - J♠ - 10♠

Tujuan akhir dari *game* ini adalah memaksimalkan akumulasi skor sepanjang permainan. Seorang pemain dinyatakan menang jika kartu telah habis, dan pemain tersebut mendapatkan nilai positif (+) hasil akumulasi dari sisa kartu pemain yang kalah. Sebaliknya pemain yang kalah akan mendapatkan nilai negatif (-) dari jumlah kartu yang tersisa.



Gambar 3.3 Contoh permainan capsah banting

3.1.3 Analisis Pembangunan *Game* Kartu Tradisional Koa Minang

Koa Minang merupakan salah satu permainan kartu yang ada di Indonesia. Koa adalah kata yang diucapkan oleh pemain ketika pemain telah memenuhi saatu kondisi dimana kondisi ini merupakan kondisi disaat pemain satu langkah lagi memenangi permainan ini, karena itu lah permainan ini disebut Koa.

Permainan kartu koa ini menggunakan 180 kartu. Dimana terdapat 30 macam kartu yang berbeda yang diulang sebanyak enam kali. Setiap kartu memiliki dua bagian motif, yaitu bagian luar dan bagian dalam.

Kartu-kartu ini memiliki sembilan macam motif yang berbeda. Motif-motif tersebut bisa dilihat pada gambar bagian luar dari kartu. Delapan motif diantaranya masing-masing memiliki tiga macam motif dalam yang berbeda. Sedangkan satu motif lagi memiliki enam macam motif dalam yang berbeda.

3.1.3.1 *Gameplay*

Game ini dimainkan oleh empat orang, masing-masing pemain mendapatkan 11 kartu pada awal permainan. Sisa kartu diletakkan di tengah untuk diambil pemain pada setiap giliran. Pada setiap giliran, pemain akan mengambil satu kartu dan membuang satu kartu. Setiap kartu yang dibuang oleh pemain hanya dapat diambil oleh pemain yang berada pada giliran selanjutnya dari pemain yang membuang kartu tersebut. Pada gilirannya pemain dapat mengambil kartu dari tumpukan atau dari buangan pemain sebelumnya.

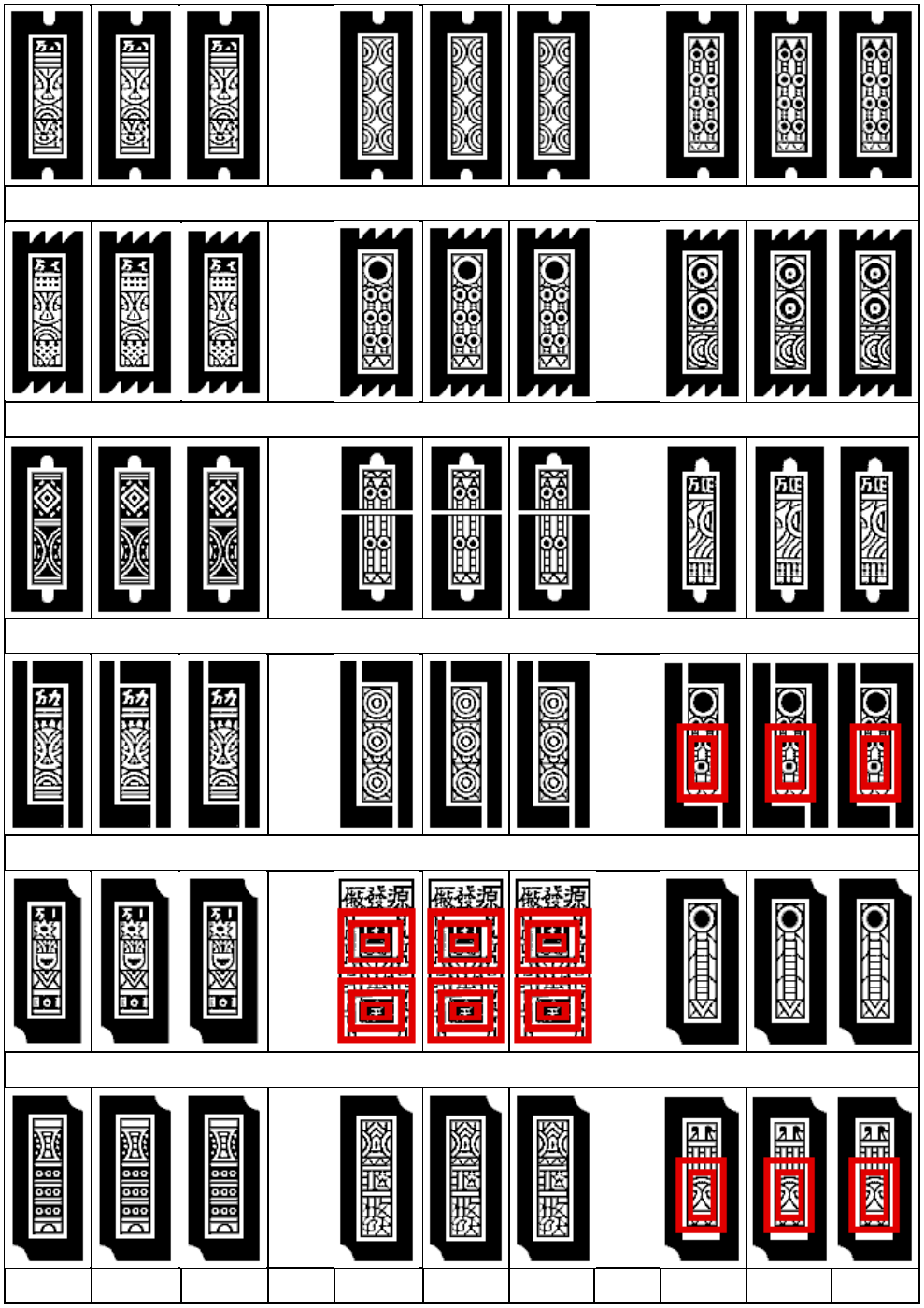
Untuk menyelesaikan *game* ini, pemain harus memenuhi kondisi yang disebut coki. Siapa yang duluan memenuhi kondisi coki, maka dialah yang menang. Berikut penjelasan dari kondisi coki yang harus dipenuhi oleh pemain :

Kartu yang berada di tangan pada dasarnya dikelompokkan menjadi empat bagian, yaitu Mato, Kaki, Kaki, dan Coki.

1. Mato

   Setiap pemain harus memiliki tiga buah kartu yang motif baik luar atau dalamnya sama persis. Tiga buah kartu ini sama motif ini dinamakan “mato”.

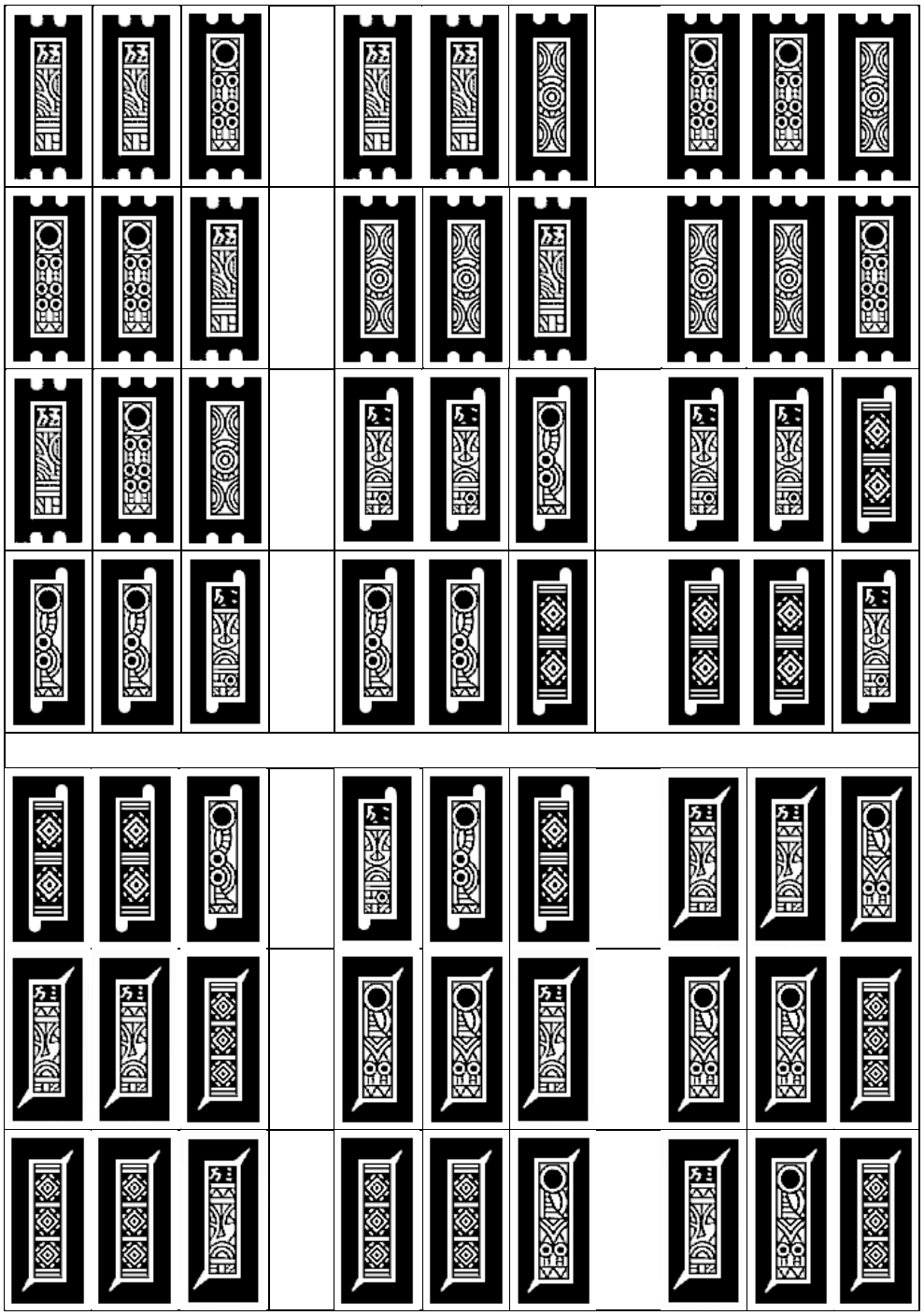


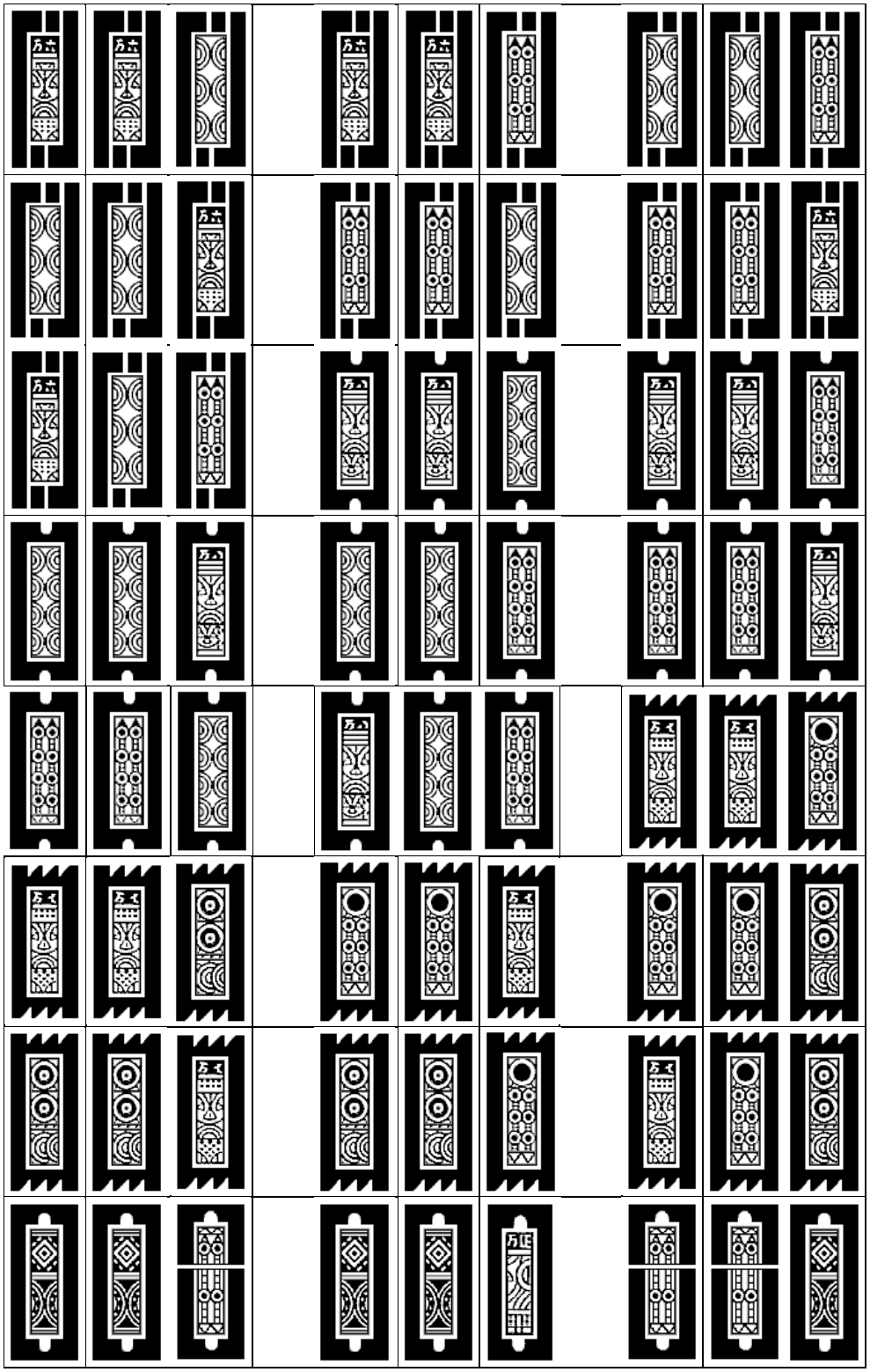


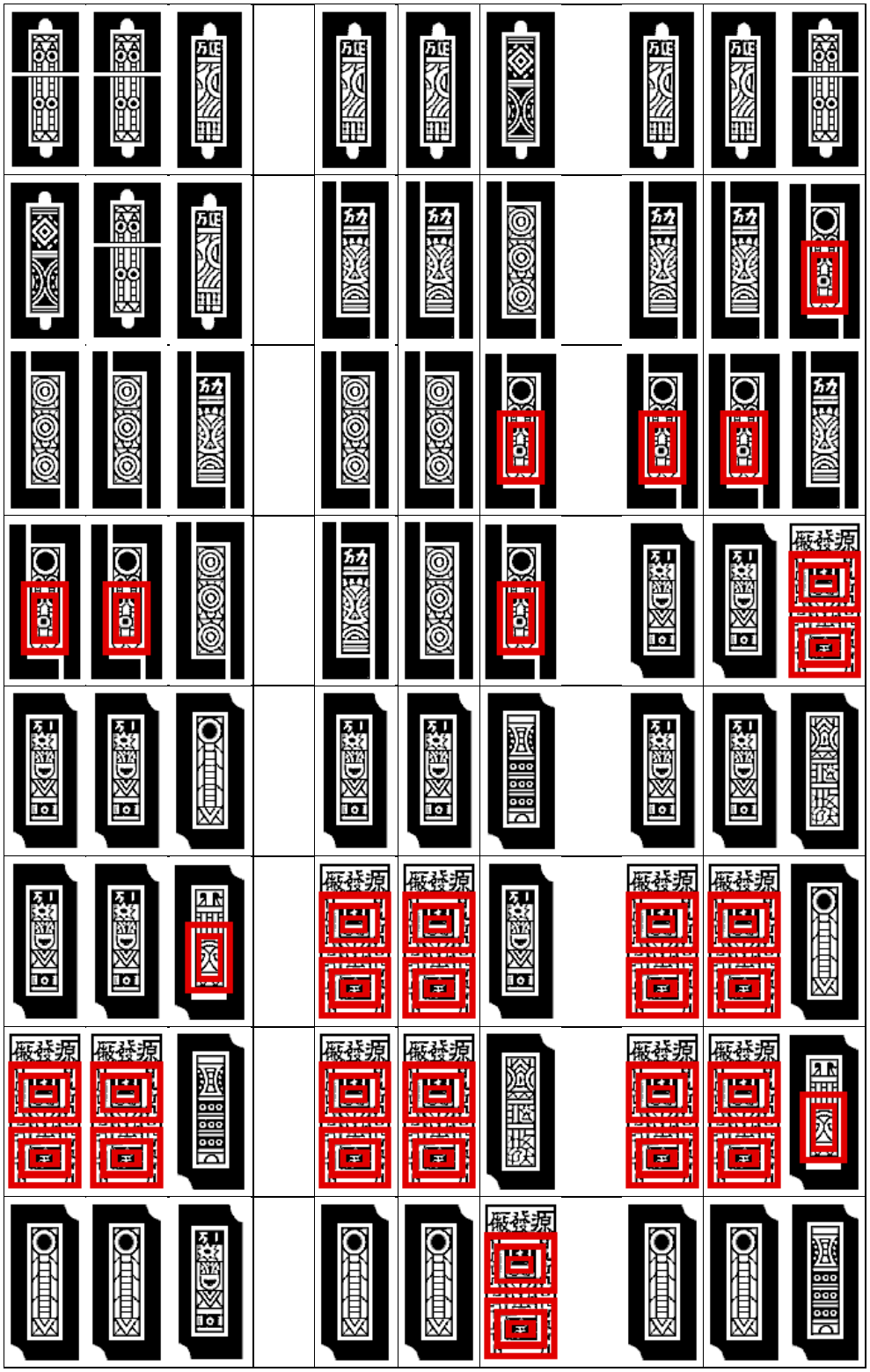
Gambar 3.4 Kombinasi-kombinasi mato

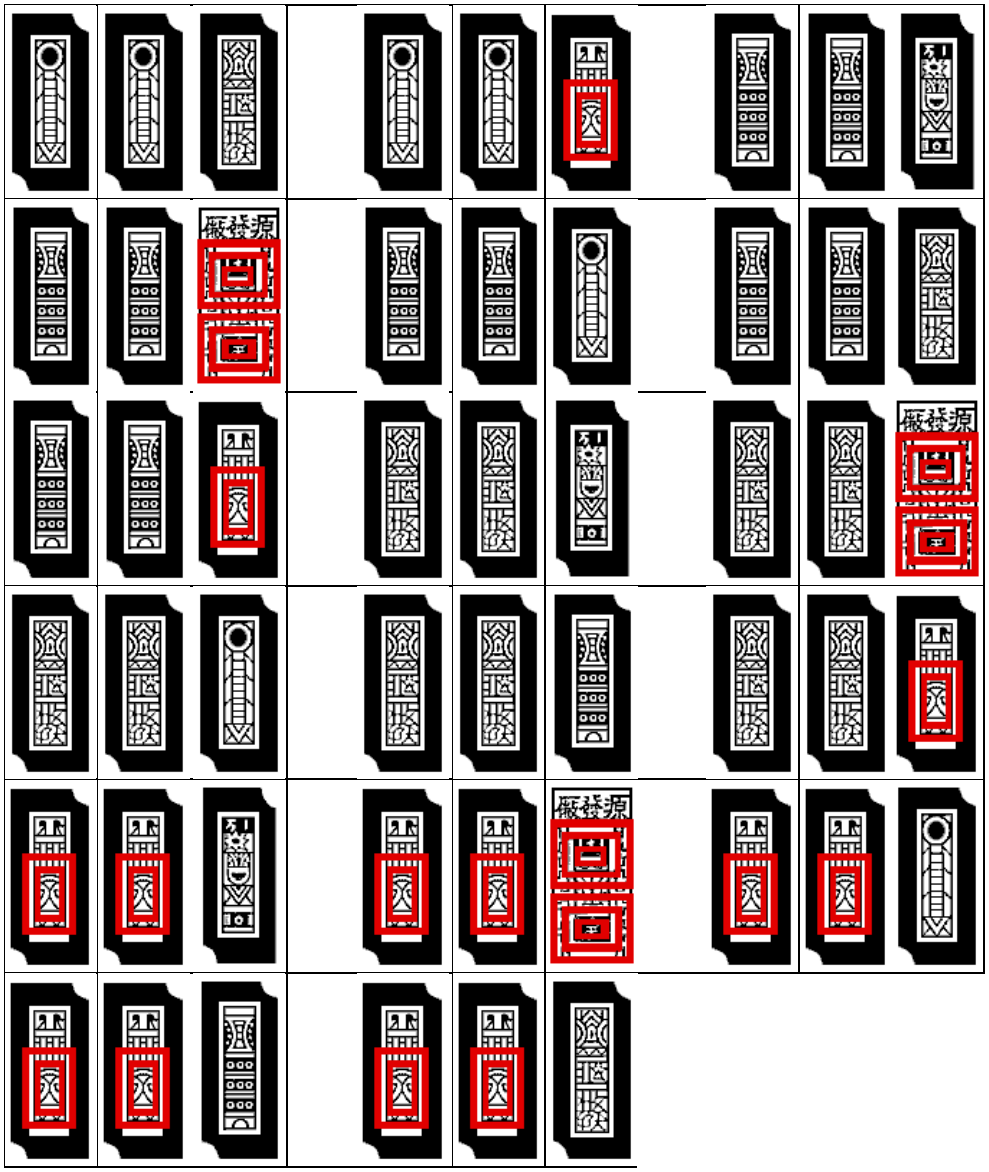
2. Kaki

Selanjutnya pemain harus memiliki dua bagian kartu yang disebut “kaki” yang masing-masing tiga kartunya harus memiliki motif luar yang sama, sedangkan untuk motif dalamnya boleh sama tapi boleh juga berbeda.





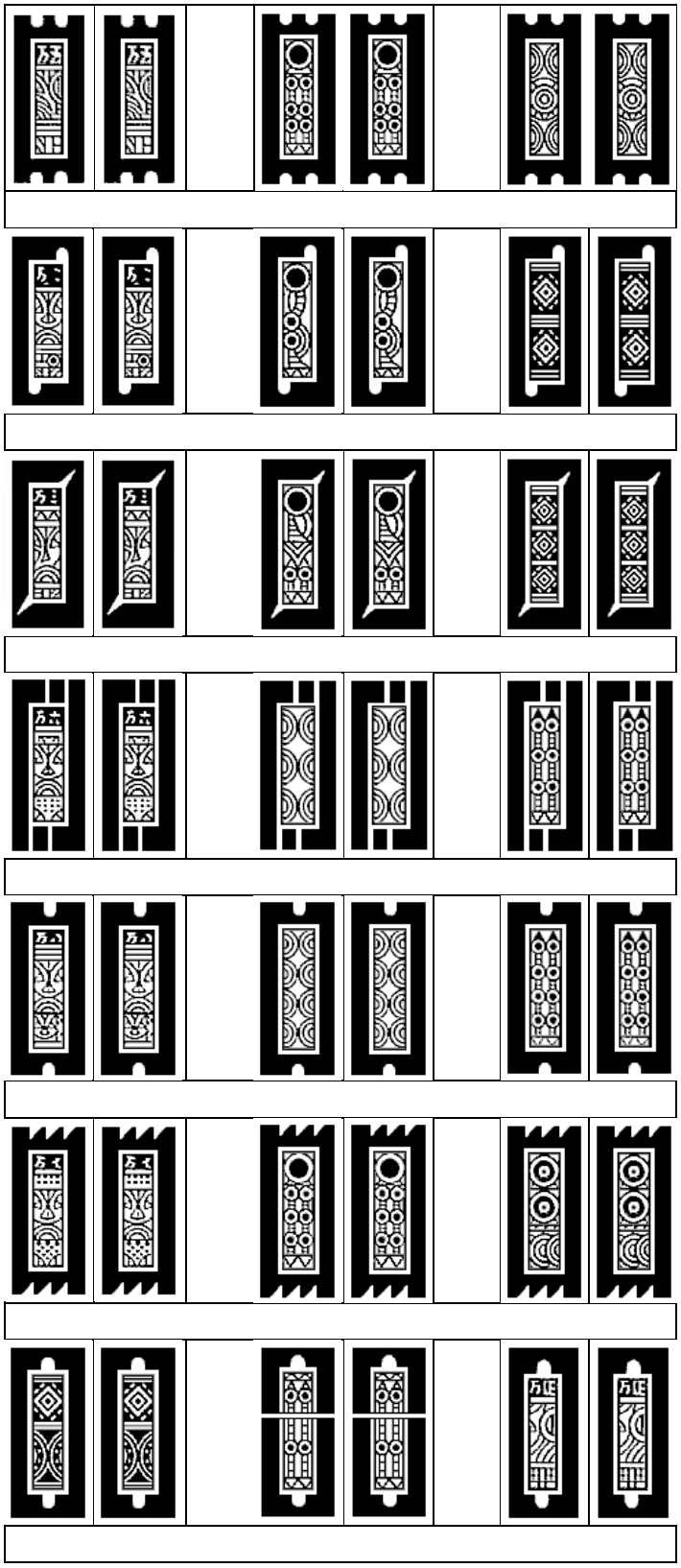


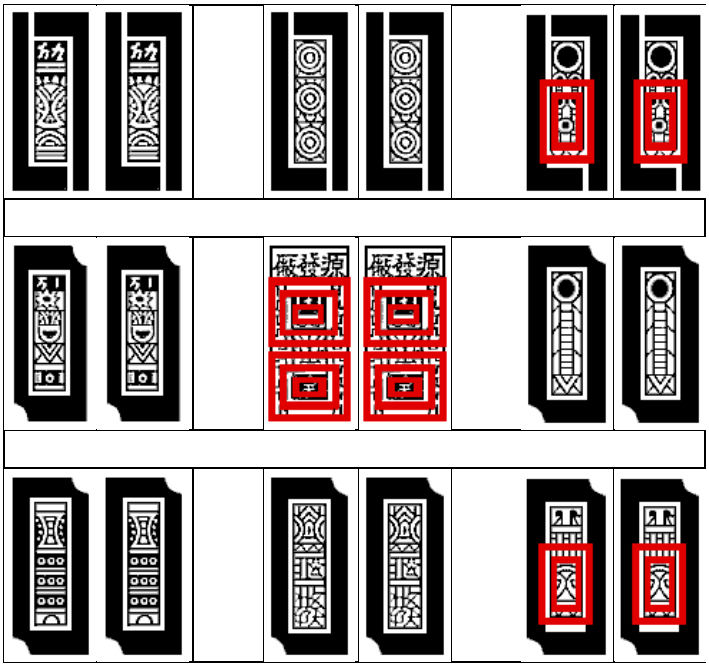


Gambar 3.5 Kombinasi-kombinasi kaki

3. Coki

Dan yang terakhir pemain harus memiliki dua kartu yang motif luar dan dalamnya harus sama persis. Dua kartu yang sama ini disebut “coki”.





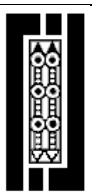
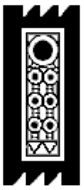
Gambar 3.6 Kombinasi-kombinasi coki

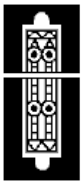
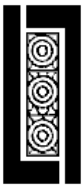
3.1.3.2 Analisis Masukan

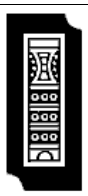
Tabel 3.1 Analisis Masukan

No	Array	Nama	Gambar
1	A1	Babi Aluih	
2	A2	Babi Gadang	
3	A3	Babi Pusek	

4	B4	Bengkok Aluih	
5	B5	Bengkok Gadang	
6	B6	Bengkok Itam	
7	C7	Jarum Aluih	
8	C8	Jarum Gadang	
9	C9	Jarum Wajik	
10	D10	Kapik Aluih	
11	D11	Kapik Anam	

12	D12	Kapik Manih	
13	E13	Pacah Aluih	
14	E14	Pacah Lapan	
15	E15	Pacah Manih	
16	F16	Sisiak Aluih	
17	F17	Sisiak Gadang	
18	F18	Sisiak Bendera	
19	G19	Suduang Itam	

20	G20	Suduang Pinggang	
21	G21	Suduang Putiah	
22	H22	Tali Aluih	
23	H23	Tali Bulek	
24	H24	Tali Merah	
25	I25	Hiu Aluih	
26	I26	Hiu Babak	
27	I27	Hiu Gadang	

28	I28	Hiu Itam	
29	I29	Hiu Kuciang	
30	I30	Hiu Merah	

3.1.3.3 Analisis Algoritma *Greedy*

Algortima yang digunakan dalam pembangun *game* tradisional Koa Minang ini adalah Algoritma *Greedy*. Berikut ini adalah penjelasan dari pengimplementasian algoritma *greedy* pada *game* tradisional Koa Minang.

Elemen-elemen yang digunakan dalam penerapan algoritma *greedy* antara lain :

1. Himpunan Kandidat = C

   Himpunan kandidat adalah kartu-kartu yang berada pada masing-masing NPC.

2. Himpunan Solusi = S

   Himpunan solusi yaitu kondisi-kondisi yang harus dipenuhi agar menang, yaitu : mato, kaki, kaki, dan coki.

3. Fungsi Seleksi

   Fungsi seleksi adalah proses pengambilan sebuah kandidat dari Himpunan Kandidat.

4. Fungsi Kelayakan

   Fungsi kelayakan untuk memeriksa sebuah kandidat berdasarkan array-nya. Dilakukan dengan cara mengecek apakah sudah memenuhi mato, kaki atau coki.

5. Fungsi Objektif

   Fungsi objektif adalah fungsi yang akan mencari kumpulan kartu yang memiliki motif luar yang sama dan motif dalam yang sama.

Algoritma *greedy* dapat dituliskan sebagai berikut :

1. Inisialisasi solusi dengan kondisi false.
2. Pilih sebuah kandidat dengan fungsi seleksi.
3. Kurangi C dengan kandidat yang telah terpilih.
4. Periksa apakah kandidat yang dipilih membentuk solusi atau tidak. Jika ya masukkan kandidat tersebut kedalam himpunan solusi. Jika tidak, buang kandidat tersebut dan tidak usah dihiraukan lagi.
5. Periksa apakah himpunan solusi sudah memberikan solusi yang lengkap. Jika ya, berhenti; jika tidak, ulangi lagi dari langkah (2).

Penjelasan algoritma *greedy* dapat dilihat pada contoh penyelesaian satu kali putaran NPC berikut :

1. Menentukan himpunan kandidat. Karena *greedy* pada aplikasi ini hanya digunakan untuk NPC, maka himpunan kandidat ditentukan sebagai array dari kartu yang ada di tangan NPC. Misalkan himpunan kandidat sebagai berikut :

   C = {A3,A1,A1,E13,E13,E13,I28,I29,I28,G20,G20}

2. Menentukan himpunan solusi, yaitu

   Mato = false
   Kaki1 = false
   Kaki2 = false
   Coki = false

3. Ambil tiga kandidat dari C.

   Kandidat yang diambil = A3,A1,A1

4. Kurangi C dengan kandidat yang diambil.

   C = {E13,E13,E13,I28,I29,I28,G20,G20}

5. Periksa kondisi yang ditentukan

6. Hasil dari langkah no lima (5)

   Mato = false

   Kaki1 = true

   Kaki2 = false

   Coki = false

7. Ulangi langkah no tiga (3), ambil tiga kandidat dari C.

   Kandidat yang diambil = E13,E13,E13

8. Lakukan langkah no empat (4), kurangi C dengan kandidat yang diambil.

C = {I28,I29,I28,G20,G20}

9. Lakukan langkah no lima (5)
10. Hasil dari langkah (9)

    Mato = true

    Kaki1 = true

    Kaki2 = false

    Coki = false

11. Ulangi langkah no tiga (3), ambil tiga kandidat dari C.

    Kandidat yang diambil = I28,I29,I28

12. Lakukan langkah no empat (4), kurangi C dengan kandidat yang diambil.

    C = {G20,G20}

13. Lakukan langkah no lima (5)
14. Hasil dari langkah (9)

    Mato = true

    Kaki1 = true

    Kaki2 = true

    Coki = false

15. Ulangi langkah no tiga (3), ambil tiga kandidat dari C.

    Kandidat yang diambil = G20,G20

16. Lakukan langkah no empat (4), kurangi C dengan kandidat yang diambil.

    C = { }

17. Lakukan langkah no lima (5)
18. Hasil dari langkah (9)

    Mato = true

    Kaki1 = true

    Kaki2 = true
19. Coki = true

3.1.4 Analisis Kebutuhan Non-Fungsional

Analisis kebutuhan non-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*.

3.1.4.1 Analisis Kebutuhan Perangkat Lunak

Analisis perangkat lunak digunakan untuk mengetahui spesifikasi perangkat lunak dalam membangun dan menjalankan aplikasi yang dibuat. Berikut merupakan spesifikasi perangkat lunak yang digunakan.

1. Sistem Operasi Windows 7
2. Adobe Flash Profesional CS5, sebagai perangkat lunak yang digunakan untuk membangun aplikasi
3. Adobe Photoshop

3.1.4.2 Analisis Kebutuhan Perangkat Keras

Analisis kebutuhan perangkat keras digunakan untuk mengetahui spesifikasi perangkat keras dalam membangun dan menjalankan aplikasi. Berikut merupakan spesifikasi perangkat keras yang digunakan.

1. Prosesor Inter Core i3 2.26 GHz
2. RAM DDR3 1GB
3. *Hardisk* 320 GB
4. *Mouse*
5. *Keyboard*

3.1.4.3 Analisis Kebutuhan *User*

Analisis kebutuhan *user* digunakan untuk mengetahui spesifikasi pengguna untuk memainkan aplikasi ini. Berikut spesifikasi pengguna yang dibutuhkan.

Tabel. 3.2 Kebutuhan Pengguna

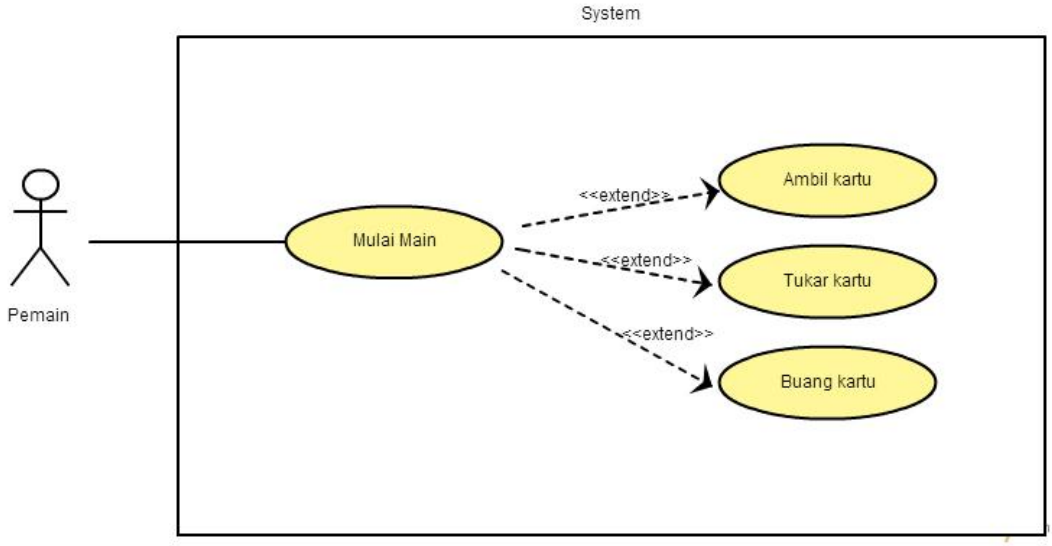
Karakteristik Fisik Pengguna	
Umur	11 tahun keatas
Gender	Laki-laki dan perempuan
Disabilities	Tidak buta warna serta tidak cacat fisik pada tangan
Pengetahuan dan Pengalaman Pengguna	
Computer Literacy	Memiliki pengalaman dalam mengoperasikan komputer
Task Experience	Pernah memainkan aplikasi *game* kartu yang lain

3.1.5 Analisis Kebutuhan Fungsional

Analisis kebutuhan fungsional menggambarkan proses kegiatan yang akan diterapkan dalam sebuah sistem dan menjelaskan kebutuhan yang diperlukan sistem agar dapat berjalan dengan baik sesuai dengan kebutuhan.

3.1.5.1 Diagram *Use Case*

Diagram *Use Case* merupakan konstruksi untuk mendeskripsikan hubungan hubungan yang terjadi antar aktor dengan aktivitas yang terdapat pada sistem. Sasaran pemodelan *use case* diantaranya adalah mendefinisikan kebutuhan fungsional dan operasional sistem dengan mendefinisikan scenario penggunaan sistem yang akan dibangun.



Gambar 3.7 Diagram *use case*

A. Definisi *Actor*

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

Tabel. 3.3 Definisi *actor*

No	*Actor*	Deskripsi
1	Pemain	Aktor utama yang mengakses aplikasi

B. Definisi *Use Case*

Use case berfungsi untuk mewakili apa yang system bisa lakukan.

Tabel 3.4 Definisi *use case*

No	*Use Case*	Deskripsi
1	Mulai main	Tampilan pertama saat aplikasi dijalankan
2	Ambil Kartu	Proses untuk mengambil kartu dari *deck*
3	Tukar Kartu	Proses untuk menukar kartu
4	Buang Kartu	Proses untuk membuang kartu

3.1.5.2 Skenario *Use Case*

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

1. Skenario *Use Case* Mulai Main

Skenario *use case* ini menjelaskan interaksi antara aktor, yaitu pemain, dengan *use case* Mulai Main yang dijelaskan pada table 3.5

Tabel 3.5 skenario *use case* mulai main

Identifikasi		
Nama	Mulai main	
Tujuan	Memulai permainan	
Aktor	Pemain	
Scenario Utama		
Konsisi Awal	Pemain berada pada menu utama	
Aksi Aktor		Reaksi Sistem
		a. Menampilkan menu utama
b. Memiilih menu mulai		
		c. Menampilkan mulai main
		d. Inisialisasi kartu
		e. Inisialisasi musuh
Kondisi akhir	Menampilkan permainan	

2. Skenario *Use Case* Ambil Kartu

Skenario *use case* ini menjelaskan interaksi antara aktor, yaitu pemain dengan *use case* ambil kartu yang dijelaskan pada table 3.6

Tabel 3.6 skenario *use case* ambil kartu

Identifikasi		
Nama	Ambil kartu	
Tujuan	Mengambil kartu dari *deck*	
Aktor	Pemain	
Scenario Utama		
Konsisi Awal	Pemain berada pada mulai main	
Aksi Aktor		Reaksi Sistem
1. pemain meng-klik *deck*		
		2. sistem memberikan satu kartu ke

	pemain
Kondisi akhir	Pemain mendapat satu kartu

3. Skenario *Use Case* Tukar Kartu

Skenario *use case* ini menjelaskan interaksi antara aktor, yaitu pemain dengan *use case* tukar kartu yang dijelaskan pada table 3.7

Tabel 3.7 skenario *use case* tukar kartu

Identifikasi	
Nama	Tukar kartu
Tujuan	Menukar posisi kartu yang ada di tangan
Aktor	Pemain
Scenario Utama	
Konsisi Awal	Pemain sudah mengambil kartu dari *deck*
Aksi Aktor	Reaksi Sistem
1. pemain meng-klik satu kartu yang ingin ditukar	
	2. array kartu disimpan dalam penyimpanan sementara (temp)
3. pemain meng-klik satu kartu lain yang akan ditukar	
	4. posisi kartu akan bertukar
Kondisi akhir	Posisi kartu bertukar

4. Skenario *Use Case* Buang Kartu

Skenario *use case* ini menjelaskan interaksi antara aktor, yaitu pemain dengan *use case* buang kartu yang dijelaskan pada table 3.8

Tabel 3.8 skenario *use case* buang kartu

Identifikasi		
Nama	Buang kartu	
Tujuan	Membuang kartu dari tangan	
Aktor	Pemain	
Scenario Utama		
Konsisi Awal	Pemain berada pada mulai main	
Aksi Aktor		Reaksi Sistem
1. pemain meng-klik satu kartu yang ingin dibuang		
		2. array kartu disimpan dalam penyimpanan sementara (temp)
3. pemain meng-klik tempat buang kartu		
		4. kartu dibuang
Kondisi akhir	Pemain membuang satu kartu	

3.1.5.3 *Activity* Diagram

Activity diagram merupakan bagian dari penggambaran sistem secara fungsional menjelaskan proses-proses logika atau fungsi yang terimplementasikan oleh kode program. *Activity* diagram memodelkan *event-event* yang terjadi di dalam suatu *use case* dan digunakan untuk pemodelan aspek dinamis dari sistem.

A. Diagram *Activity* Mulai Main

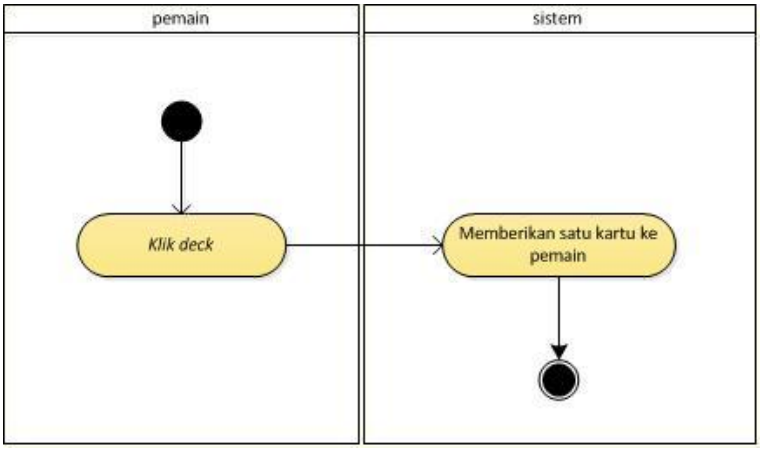
Proses ini terjadi saat permainan dijalankan. Sistem akan menampilkan arena permainan dimana pemain dapat langsung memainkan permainan ini. Berikut diagram *activity* mulai permainan.



Gambar 3.8 Diagram *activity* mulai main

B. Diagram *Activity* Ambil Kartu

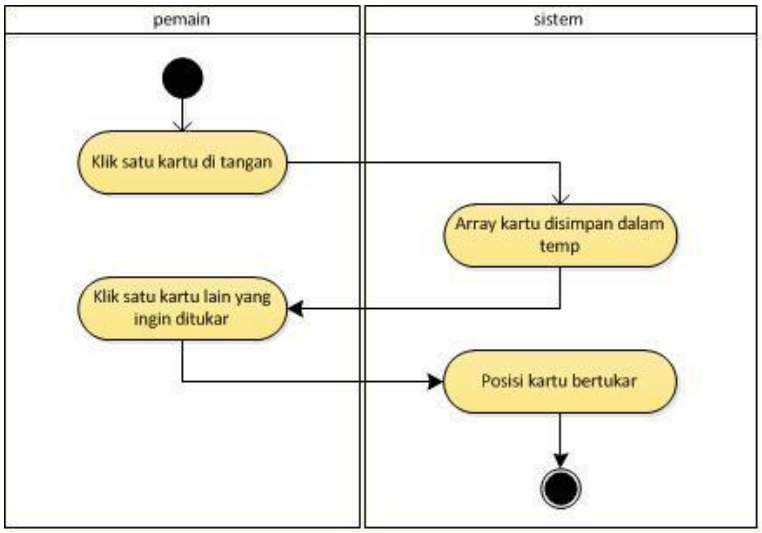
Menggambarkan rangkaian aliran dari aktivitas ambil kartu, digunakan untuk mengambil kartu.



Gambar 3.9 Diagram *activity* ambil kartu

C. Diagram *Activity* Tukar Kartu

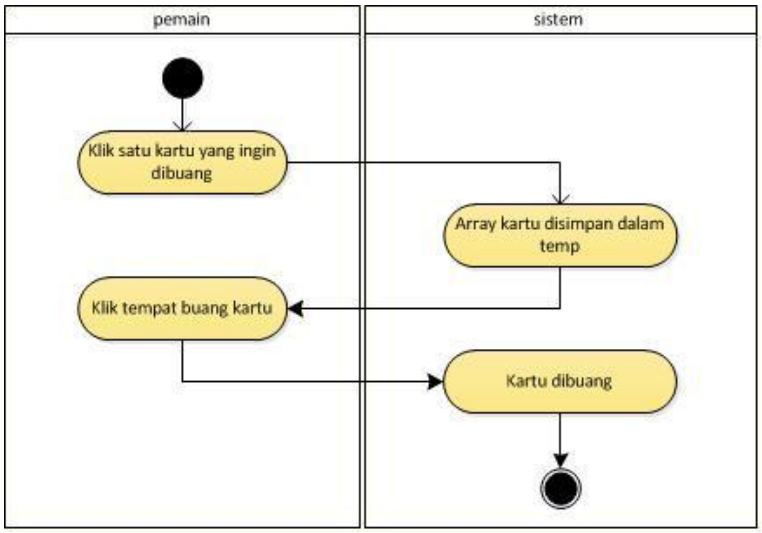
Menggambarkan rangkaian aliran dari aktivitas tukar kartu, digunakan untuk menukar kartu.



Gambar 3.10 Diagram *activity* tukar kartu

D. Diagram *Activity* Buang Kartu

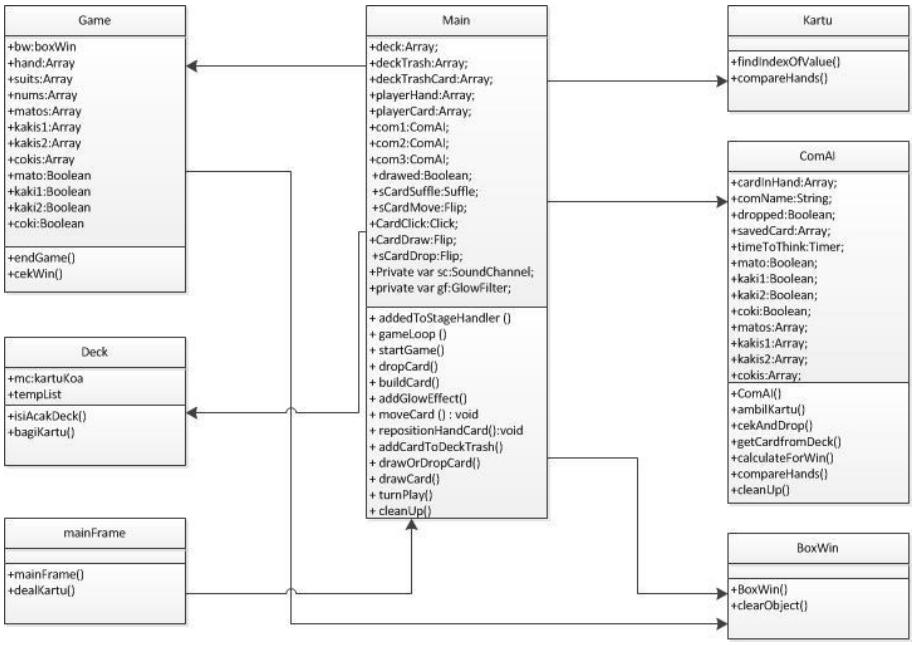
Menggambarkan rangkaian aliran dari aktivitas buang kartu, digunakan untuk membuang kartu.



Gambar 3.11 Diagram *activity* buang kartu

3.1.5.4 *Class Diagram*

Berikut ini adalah perancangan struktur sistem yang terdapat pada aplikasi Koa Minang, yang digambarkan dengan *class* diagram yang dapat dilihat pada Gambar 3.12.

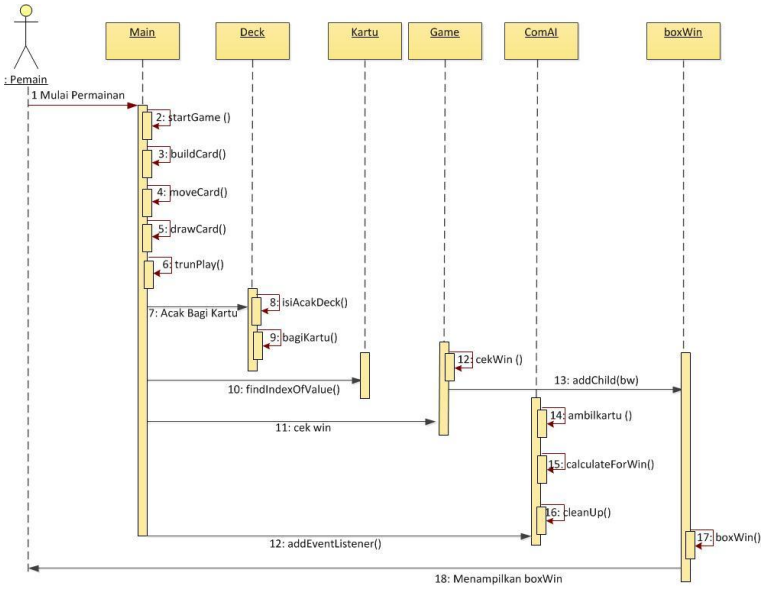


Gambar 3.12 Diagram class

3.1.5.5 Diagram *Sequence*

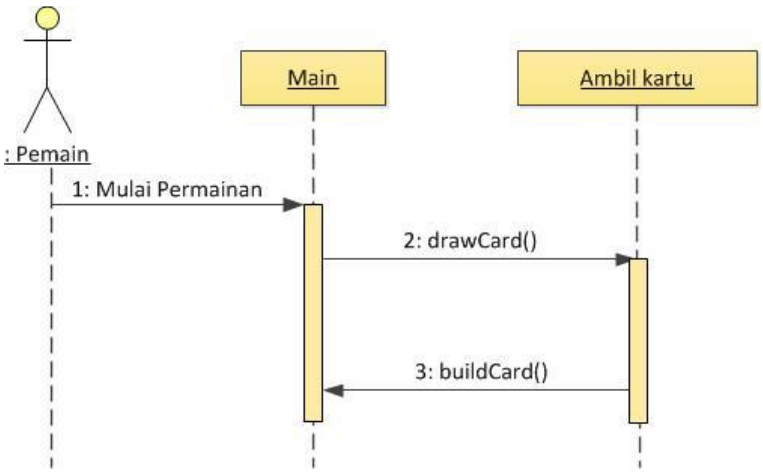
Berikut merupakan diagram *sequence* yang terdapat pada aplikasi berdasarkan *use case* yang ada.

A. Diagram Sequence Mulai Main



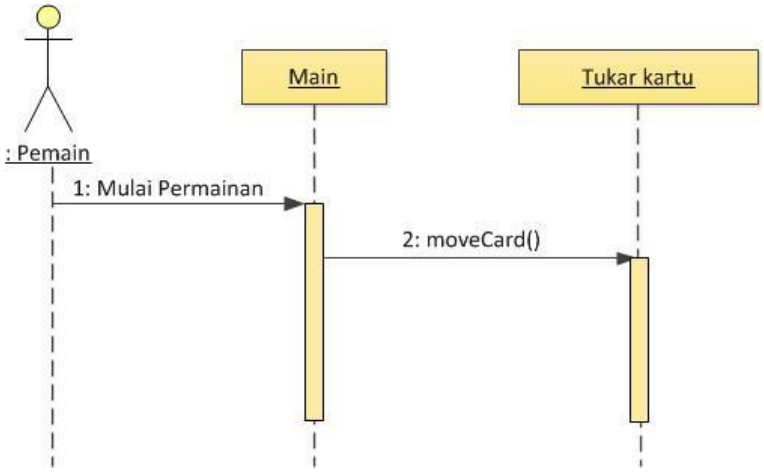
Gamber 3.13 Diagram sequence mulai main

B. Diagram Sequence Ambil Kartu



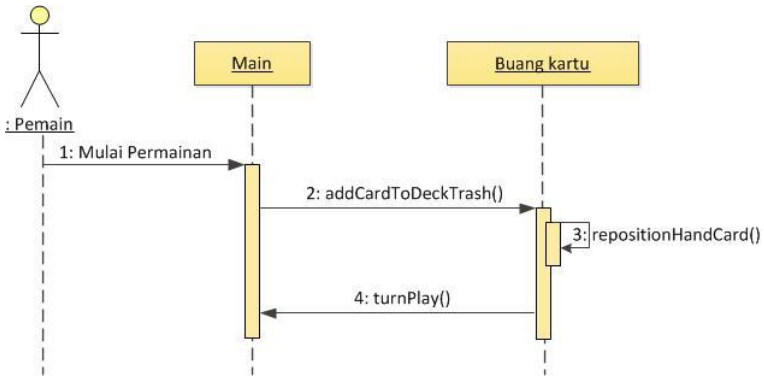
Gamber 3.14 Diagram sequence ambil kartu

C. Diagram Sequence Tukar Kartu



Gamber 3.15 Diagram sequence tukar kartu

D. Diagram Sequence Buang Kartu



Gamber 3.16 Diagram sequence buang kartu

3.2 Perancangan Sistem

Perancangan sistem dapat didefinisikan sebagai penggambaran, perencanaan, dan pembuatan sketsa atau pengaturan dari beberapa elemen yang terpisah kedalam satu kesatuan yang utuh dan berfungsi. Perangcangan system menentukan bagaimana suatu system akan menyelesaikan apa yang mesti diselesaikan.

3.2.1 Perangcangan Komponen Permainan

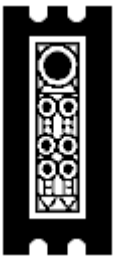
3.2.1.1 Kartu

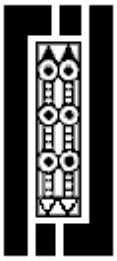
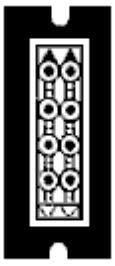
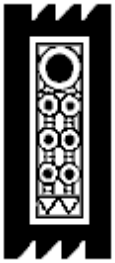
Berikut kartu-kartu yang digunakan dalam permainan ini beserta nama dan gambarnya.

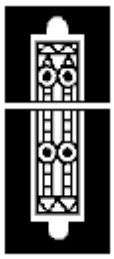
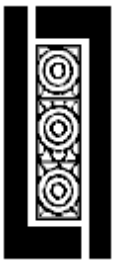
Tabel 3.9 Motif luar kartu koa

								
Babi	Bengkok	Jarum	Kapik	Pacah	Sisiak	Suduang	Tali	Hiu

Tabel 3.10 Nama-nama kartu koa

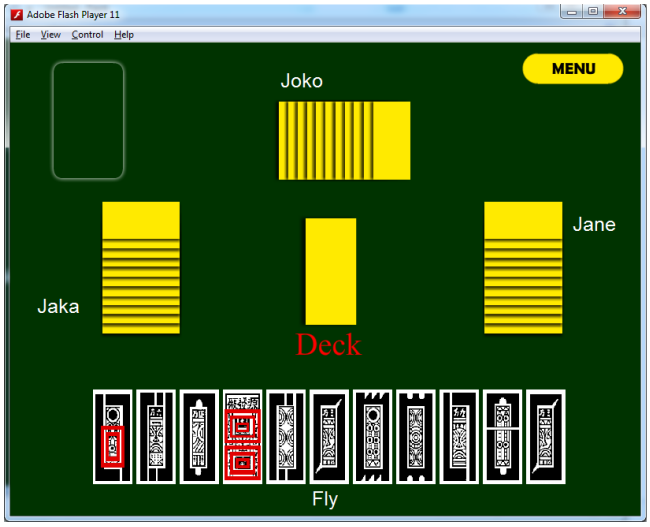
Babi		
		
Babi Aluih	Babi Gadang	Babi Pusek
Bengkok		
		
Bengkok Aluih	Bengkok Gadang	Bengkok Itam

Jarum		
		
Jarum Aluih	Jarum Gadang	Jarum Wajik
Kapik		
		
Kapik Aluih	Kapik Anam	Kapik Manih
Pacah		
		
Pacah Aluih	Pacah Lapan	Pacah Manih
Sisiak		
		
Sisiak Aluih	Sisiak Bendera	Sisiak Gadang

Suduang		
		
Suduang Putiah	Suduang Pinggang	Suduang Itam
Tali		
		
Tali Aluih	Tali Bulek	Tali Merah
Hiu		
		
Hiu Aluih	Hiu Babak	Hiu Gadang
		
Hiu Itam	Hiu Kuciang	Hiu Merah

3.2.1.2 *Storyboard*

Storyboard merupakan rangkaian gambar manual yang dibuat secara keseluruhan, sehingga menggambarkan suatu cerita. Setiap *scene* menggambarkan secara jelas komponen-komponen multimedia serta perilakunya. Penjelasan dapat menggunakan simpol maupun teks.



Gambar 3.17 *storyboard* permainan

Pada aplikasi Koa Minang ini para pemain dibagikan masing-masing 11 kartu. Setiap giliran pemain akan mengambil satu kartu dan membuang satu kartu. Pemain boleh mengambil kartu dari *deck* atau dari buangan pemain sebelum gilirannya. Permainan selesai ketika salah satu pemain memenuhi kondisi yang ditentukan.

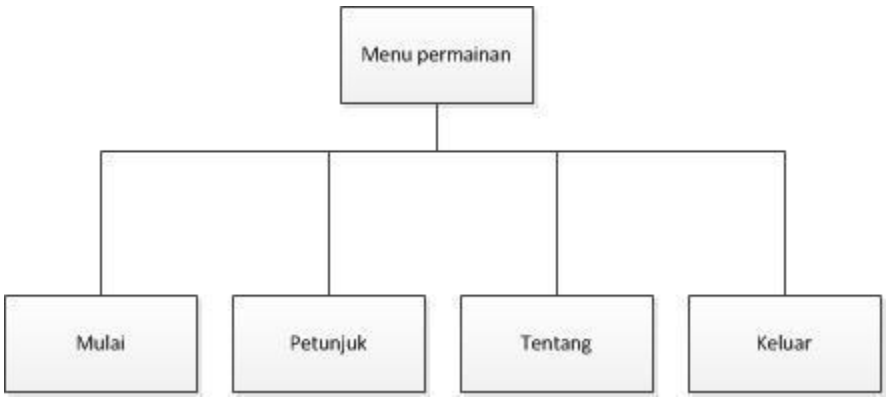
3.2.2 Perancangan Arsitektur

Setelah melakukan perancangan pada sistem yang akan dibangun, maka dilakukanlah perancangan arsitektur. Perancangan arsitektur yang telah dibuat

meliputi beberapa perancangan diantaranya struktur menu, perancangan antarmuka, perancangan pesan, dan jaringan semantik.

3.2.2.1 Perancangan Struktur Menu

Untuk mempermudah dalam pembuatan sistem, diperlukan perancangan struktur menu program yang akan dibangun. Perancangan struktur menu program ini membantu dalam merancang bagian-bagian dari sistem yang sebenarnya dan untuk mengetahui bagian mana yang nantinya akan diakses terlebih dahulu setelah program selesai dibuat. Perancangan struktur menu pada aplikasi permainan Koa Minang dapat dilihat pada gambar 3.18.



Gambar 3.18 Struktur menu *Koa Minang*

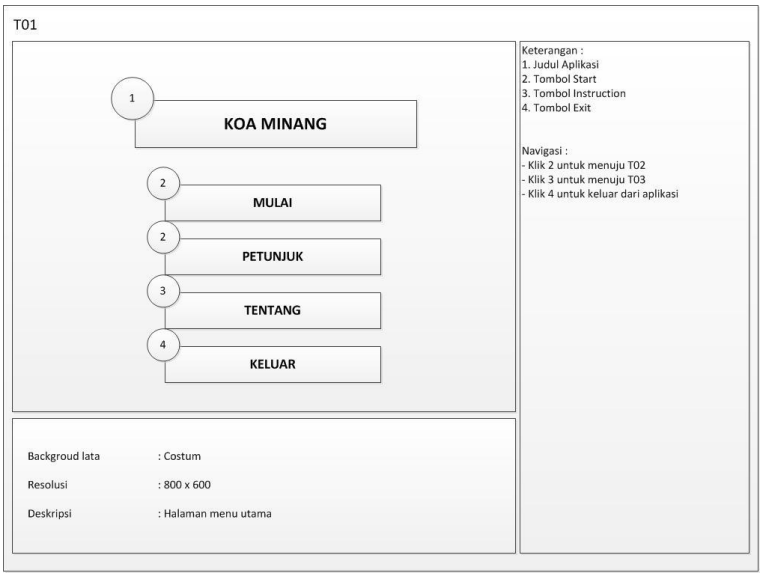
1. Mulai : merupakan menu untuk memulai permainan.
2. Petunjuk : merupakan menu untuk menampilkan informasi tentang cara memainkan aplikasi Koa Minang.
3. Tentang : merupakan menu untuk menampilkan informasi tentang kartu Koa.
4. Exit : merupakan menu untuk keluar dari aplikasi.

3.2.2.2 Perancangan Antar Muka

Antarmuka merupakan bentuk tampilan dari program yang tampil pada layar komputer yang bertujuan memberikan gambaran tentang aplikasi yang akan dibangun, sehingga akan memudahkan pengimplementasian aplikasi sesuai ukuran layar dan memudahkan dalam pembuatan aplikasi.

1. Perancangan Antarmuka Menu Utama

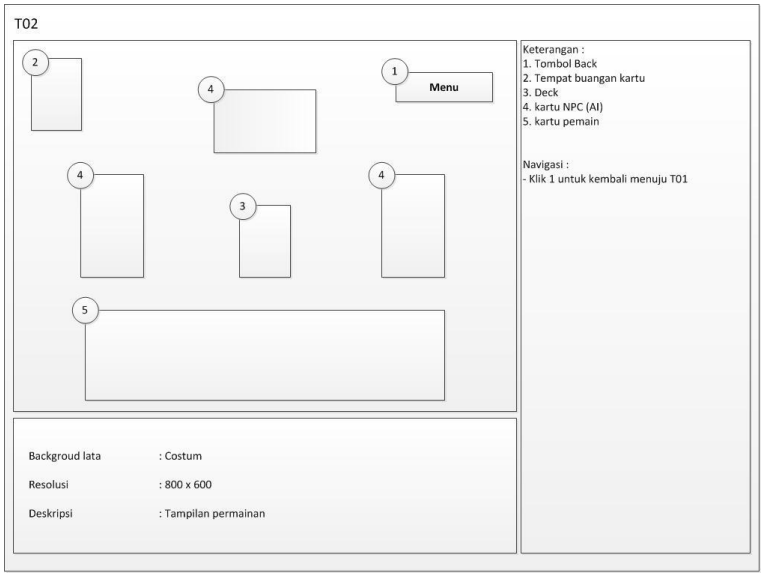
Pada menu utama terdapat tiga tombol pada layar yang akan dipilih oleh pemain sebelum bermain. Antarmuka menu utama dapat dilihat pada gambar 3.19.



Gambar 3.19 Perancangan antar muka menu utama

2. Perancangan Antarmuka Mulai Main

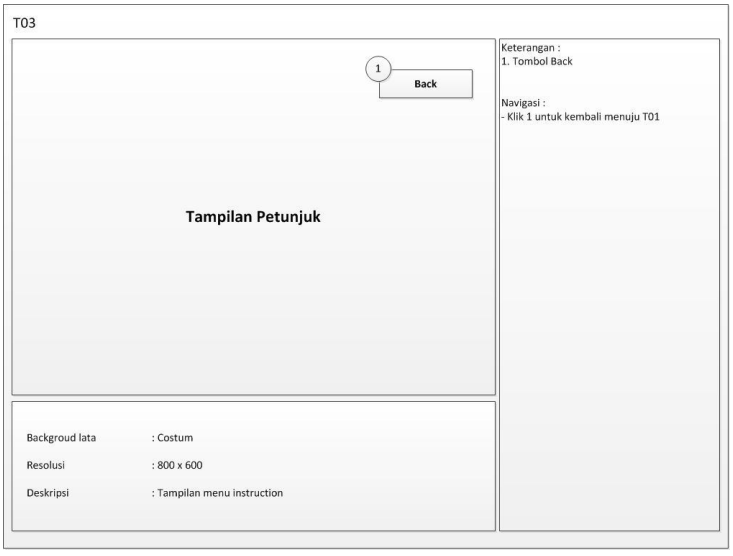
Masuk ke menu Start menampilkan area permainan dan tombol kembali ke menu utama. Antarmuka menu mulai main dapat dilihat pada gambar 3.20.



Gambar 3.20 Perancangan antar muka mulai main

3. Perancangan Antarmuka Petunjuk

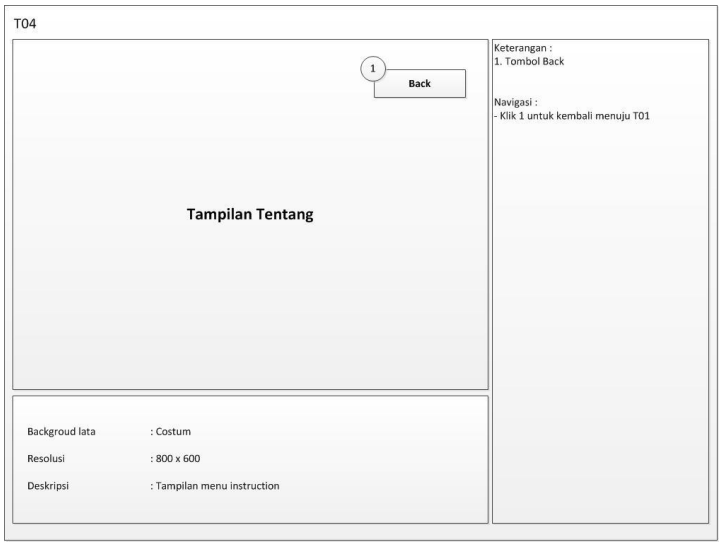
Masuk ke menu Petunjuk akan menampilkan petunjuk permainan yang dapat dilihat pada gambar 3.21.



Gambar 3.21 Perancangan antar muka petunjuk

4. Perancangan Antarmuka Tentang

Masuk ke menu Petunjuk akan menampilkan petunjuk permainan yang dapat dilihat pada gambar 3.22.



Gambar 3.22 Perancangan antar muka tampil tentang

3.2.2.3 Perancangan Pesan

Pesan yang dirancang dalam konteks aplikasi yang dikembangkan merupakan pesan peringatan yang keluar jika pengguna melakukan aktivitas tertentu yang akan memicu *event* tertentu.

1. Perancangan Pesan Peringatan Keluar Aplikasi



Gambar 3.23 Perancangan pesan peringatan keluar aplikasi

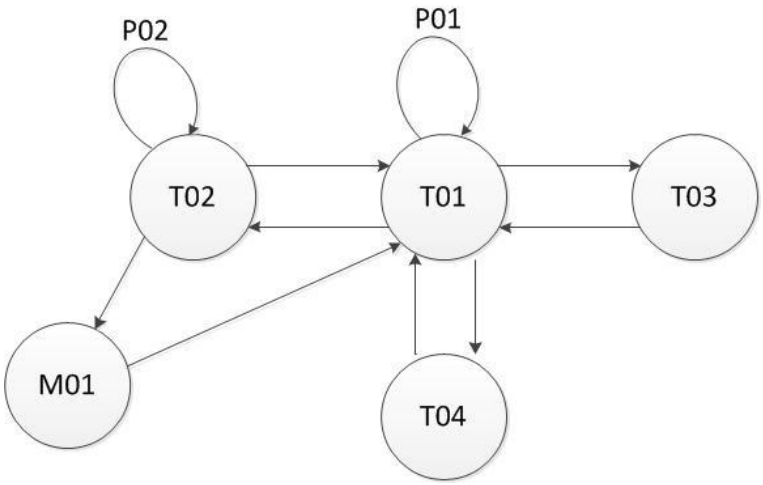
2. Perancangan Pesan Peringatan Kembali ke Menu



Gambar 3.24 Perancangan pesan peringatan keluar aplikasi

3.2.4 Jaringan Semantik

Jaringan semantik menggambarkan hubungan tampilan satu dengan tampilan lainnya. Jaringan semantic dapat dilihat pada gambar 3.25.



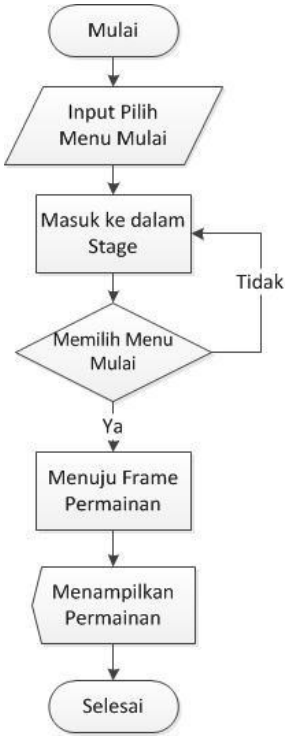
Gambar 3.25 Jaringan semantik *Koa Minang*

3.2.5 Perancangan *Method*

Perancangan *method* merupakan perangcangan yang berfungsi untuk mendeskripsikan *method-method* yang berada di dalam aplikasi. *Method* dapat dipanggil dengan menyertakan variable, baik hanya satu variable, banyak variable

atau bahkan tidak ada sama skali. Adapun *method-method* yang terdapat dalam *game* Koa Minang yang akan dibangun adalah sebagai berikut :

1. Perancangan *Method* Mulai Main



Gambar 3.26 Method mulai main