

## **BAB III**

### **ANALISA DAN PERANCANGAN**

#### **III.1. Analisis Masalah**

Strategi di belakang Minimax algoritma adalah komputer berasumsi bahwa kedua pemain akan main sejauh kemampuan mereka. Maka, jika lawan mempunyai pilihan suatu gerak tidak baik atau suatu yang baik gerak, komputer akan mempunyai lawan memilih yang baik gerak. sungguh mencoba untuk memilih gerak yang mengakibatkan nilai titik yang paling rendah. Konsep relatif nyata dan sederhana ini menjadi rahasia di belakang minimax pohon. Jika komputer di program untuk mencari maksimal, maka rangkaian atau gerakan yang terbaik akan selalu ditemukan yang mengakibatkan nilai titik yang paling tinggi. Demikian juga dengan minimal, maka rangkaian atau gerakan yang terbaik akan selalu ditemukan yang mengakibatkan nilai titik yang paling rendah.

##### **III.1.1. Analisis Komponen Permainan**

Beberapa komponen dalam permainan *checkers* :

###### 1. Permainan

Permainan ini dimainkan oleh dua orang. Berperan sebagai lawan satu sama lain.

###### 2. Papan

Papan yang digunakan dalam permainan ini mirip dengan papan catur. Terdiri dari 8 x 8 kotak-kotak kecil warna terang dan gelap. Permukaan yang dimainkan terdiri dari 32 kotak hanya gelap. Sebuah konsekuensi ini

adalah bahwa, dari perspektif masing-masing *player*, kiri dan kanan tikungan mendorong strategi yang berbeda.

### 3. Koin

Koin yang digunakan berbentuk silinder datar seperti koin pada umumnya. Koin yang digunakan pada permainan ini berwarna terang dan gelap. Pada awalnya, warna terang direpresentasikan dengan warna putih dan warna gelap direpresentasikan dengan warna merah. Tapi pada perkembangannya, banyak warna yang digunakan untuk menggantikan warna-warna tersebut.

### 4. Posisi awal

Masing-masing pemain mempunyai 12 koin sewarna yang diletakkan pada 3 baris pertama pada bidang yang terdekat dengan pemain dan diletakkan pada bidang yang berwarna gelap. Pemain yang memainkan koin berwarna gelap, memulai langkah awal terlebih dahulu.

### 5. Cara bergerak

Koin bergerak secara diagonal mengikuti warna bidang yang gelap, satu di tiap langkahnya. Cara lain adalah dengan melangkahi satu buah koin lawan. Keadaan itu mungkin dapat dilakukan jika pada diagonal setelah koin lawan, merupakan bidang kosong. Jika langkah kedua itu terjadi, koin lawan yang dilangkahi mati, dan harus keluar dari bidang permainan. Koin dengan pangkat “biasa” hanya dapat bergerak maju. Namun, koin dengan pangkat “raja”, dapat bergerak maju maupun mundur.

#### 6. Raja

Raja merupakan sebutan untuk koin yang mendapat perlakuan istimewa. Koin ini adalah koin yang bertahan dan berhasil masuk ke ujung baris bidang lawan. Koin ini didefinisikan sebagai tumpukan dari dua koin biasa. Keistimewaan koin ini adalah dapat bergerak baik maju maupun mundur.

#### 7. Permainan selesai

Permainan selesai ditandai oleh habisnya koin lawan pada bidang permainan atau koin sudah tidak dapat bergerak kemanapun.

### **III.2. Penerapan Metode Algoritma Minmax**

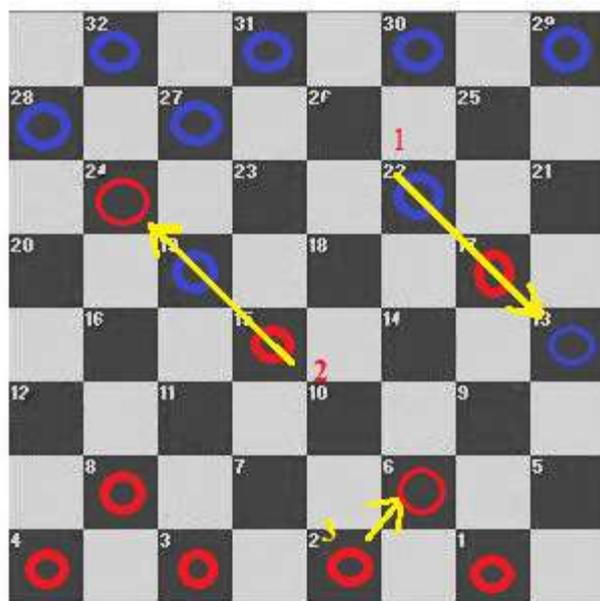
Diasumsikan bahwa permainan memiliki dua pemain, disebut Maks dan Min, dan yang Maks bergerak pertama. Max ingin memaksimalkan hasil yang bisa dicapai. Min, di lain pihak, ingin meminimalkan hasil yang diperoleh dapat Max. Permainan teori-nilai yang permainan ketika adalah hasil baik dan MinMax bermain dengan baik. Nilai ini dapat ditentukan, dalam teori, oleh semuakemungkinan memperluas secara rekursif dari continuations permainan's mulai posisi semua jalan menuju ke akhir permainan (posisi terminal).

Aturan yang digunakan untuk menyebarkan hasil dikenal (dari pandangan Max) dari terminal kembali ke posisi awal. Apabila itu adalah Max's memindahkan, dia selalu memilih beraktivitas yang mengarah ke nilai maksimum. Ketika itu Min's memindahkan, dia akan mencoba untuk meminimalkan Max's keuntungan dengan memilih selalu berpindah tempat dengan nilai minimum.

Berasumsi bahwa mungkin hasil dari permainan adalah nilai bulat di rentang  $-9 - 9$  (dimana angka positif menunjukkan hasil yang menguntungkan untuk Maks). Itu posisi awal adalah untuk mencari kedalaman 2 lapis (satu lapis merujuk dibuat untuk memindahkan satu per satu *player*). Setiap *node* di pohon mewakili posisi permainan dan ujung-ujungnya antara *node* mewakili bergerak. Square *node* menunjukkan posisi dimana Maks telah berpindah tempat, dan circled *node* Min adalah di mana telah beraktivitas.

Algoritma Minimax yang meluaskan pohon secara rekursif dalam ke-kiri-kanan. Pencarian mulai dalam posisi A. Pertama yang bergerak terkemuka a1 adalah diperluas untuk permainan posisi B, kemudian bergerak menuju ke posisi b1, yang merupakan posisi terminal (dalam hal ini contoh sederhana) dan memiliki hasil +5. Algoritma yang sekarang backtracks, kembali +5 atas ke tingkat sebelumnya (node B), di mana bergerak B2 dan b3 yang diperluas dengan gaya yang sama; mereka mendapatkan nilai +9 dan +8, masing-masing. Pada B Min pemain yang memiliki pilihan yang pindah ke memilih, dan dia memilih memindahkan b1 karena mengurangi Max's keuntungan. Dengan demikian +5 (nilai pindah dari b1) adalah kembali kembali ke A. Dengan demikian, memindahkan a1 menjamin Maks nilai minimal +5. Max, namun harus terus menjelajahi tersisa dua bergerak, karena mereka dapat mengakibatkan sebuah hasil yang lebih menguntungkan. Dalam contoh ini yang tidak kasus, dan pencarian nilai yang akan ditentukan +5.

Adapun proses langkah permainan checkers ialah secara diagonal atau menyilang dari kiri maupun kekanan dalam tetap hanya di areal papan berwarna gelap. Proses langkah permainan akan di tunjukan oleh gambar III.1 berikut ini :



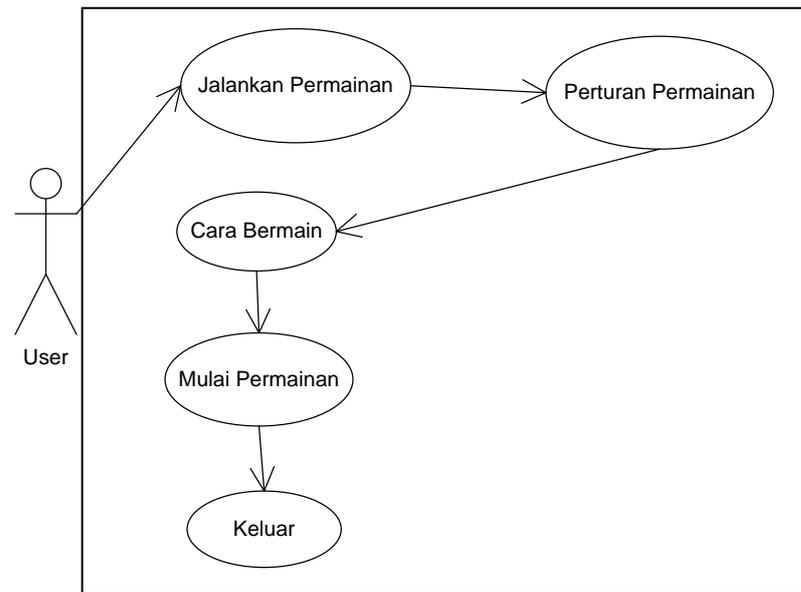
**Gambar III.1. Proses Langkah Permainan Checkers**

Pada angka no 1 (satu) menjelaskan koin biru memakan koin merah, sehingga koin biru berpindah ke papan no 13. Kemudian pada papan 2 (dua) sama halnya seperti papan biru hanya saja disini papan merah yang memakan koin biru, Sedangkan pada no 3 (tiga) adalah proses langkah koin.

### III.3. Desain Sistem

#### III.3.1. Use Case Diagram

Adapun rancangan *use case* diagram yang dibangun pada aplikasi *game* permainan checkers sebagai berikut.



**Gambar III.2. Use Case Diagram Permainan Checkers**

### III.3.2. Storyboard

*Storyboard* merupakan gambaran skenario yang dibuat secara bertahap yakni setiap *Layer* pada aplikasi. *Storyboard* ini yang nantinya menerangkan susunan atau urutan hasil aplikasi yang dibuat. *Storyboard* berupa pengembangan dari setiap bagian yang akan disampaikan dalam aplikasi, yaitu sebagai berikut :

1. Layer 1 : Judul, digunakan untuk menampilkan informasi Judul aplikasi serta terdapat dua buah menu *Play* dan *Exit*.
2. Layer 2 : Menu Permainan, digunakan untuk menampilkan permainan checkers pada aplikasi saat *user* memilih *Play*.

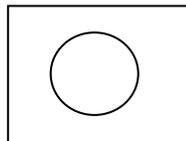
Berikut adalah *story board* dari perancangan aplikasi permainan checkers dengan metode minimax berdasarkan komponen permainan.

### 1. Pemain

Permainan ini dimainkan oleh dua orang. Berperan sebagai lawan satu sama lain. Di dalam perancangan ini pemain dilambangkan dengan koin. Masing-masing pemain mempunyai 8 koin sewarna yang diletakkan pada 2 baris pertama pada bidang yang terdekat dengan pemain dan diletakkan pada bidang yang berwarna gelap. Pemain yang memainkan koin berwarna gelap, memulai langkah awal terlebih dahulu.

### 2. Koin

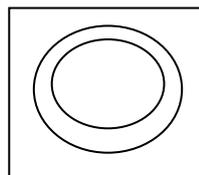
Koin yang digunakan berbentuk silinder datar seperti koin pada umumnya. Koin yang digunakan pada permainan ini berwarna terang dan gelap. Pada awalnya, warna terang direpresentasikan dengan warna putih dan warna gelap direpresentasikan dengan warna hitam. Di dalam perancangan permainan koin yg dibuat diwakili dengan warna, warna tersebut adalah merah dan biru.



**Gambar III.3. Koin Yang di Letakan di Papan Checkers**

Mewakili koin yang dimainkan oleh komputer/koin Pemain bila bermain dalam modus 2 pemain. Koin bergerak secara diagonal mengikuti warna bidang yang gelap, satu di tiap langkahnya. Cara lain adalah dengan melangkahi satu buah koin lawan. Keadaan itu mungkin dapat dilakukan jika pada diagonal setelah koin lawan, merupakan bidang kosong. Jika

langkah kedua itu terjadi, koin lawan yang dilangkahi mati, dan harus keluar dari bidang permainan. Koin dengan pangkat “biasa” hanya dapat bergerak maju. Namun, koin dengan pangkat “raja”, dapat bergerak maju maupun mundur.

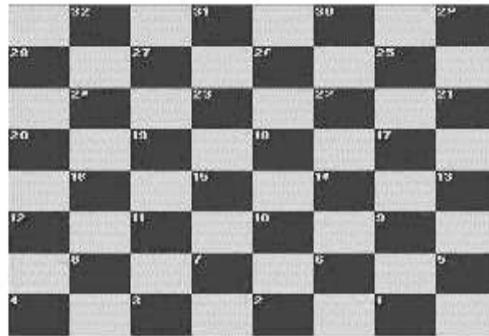


**Gambar III.4. Koin dengan pangkat raja**

Raja merupakan sebutan untuk koin yang mendapat perlakuan istimewa. Koin ini adalah koin yang bertahan dan berhasil masuk ke ujung baris bidang lawan. Koin ini didefinisikan sebagai tumpukan dari dua koin biasa. Keistimewaan koin ini adalah dapat bergerak baik maju maupun mundur.

### 3. Papan

Papan yang digunakan dalam permainan ini mirip dengan papan catur. Terdiri dari 8 x 8 kotak-kotak kecil warna terang dan gelap. Permukaan yang dimainkan terdiri dari 32 kotak hanya gelap. Sebuah konsekuensi ini adalah bahwa, dari perspektif masing-masing player, kiri dan kanan tikungan mendorong strategi yang berbeda. Dalam perancangan permainan ini papan diwakili dengan gambar kotak berwarna hitam dan putih yang disusun sehingga membentuk arena permainan.



**Gambar III.5. Papan checkers**

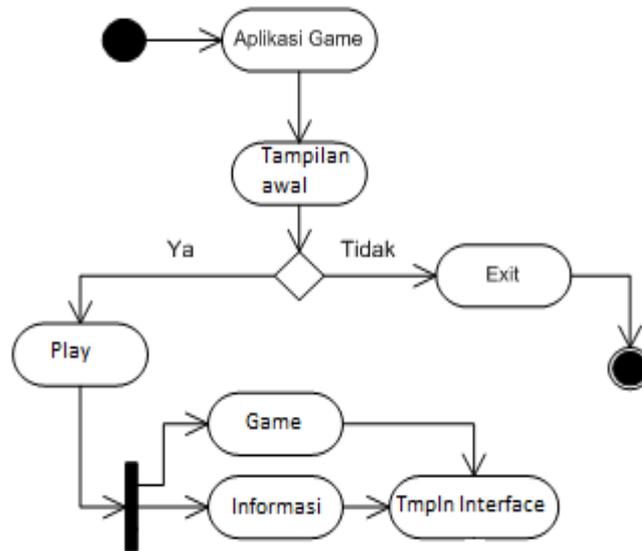
Permainan selesai ditandai oleh habisnya koin lawan pada bidang permainan atau koin sudah tidak dapat bergerak kemana pun.

### **III.3.3. Activity Diagram**

*Activity diagram* adalah representasi grafis dari seluruh tahapan alur kerja. Diagram ini mengandung aktivitas, pilihan tindakan. Pada pemodelan UML, diagram ini dapat digunakan untuk menjelaskan alur kerja sistem.

#### **1. Activity Diagram Menu Game**

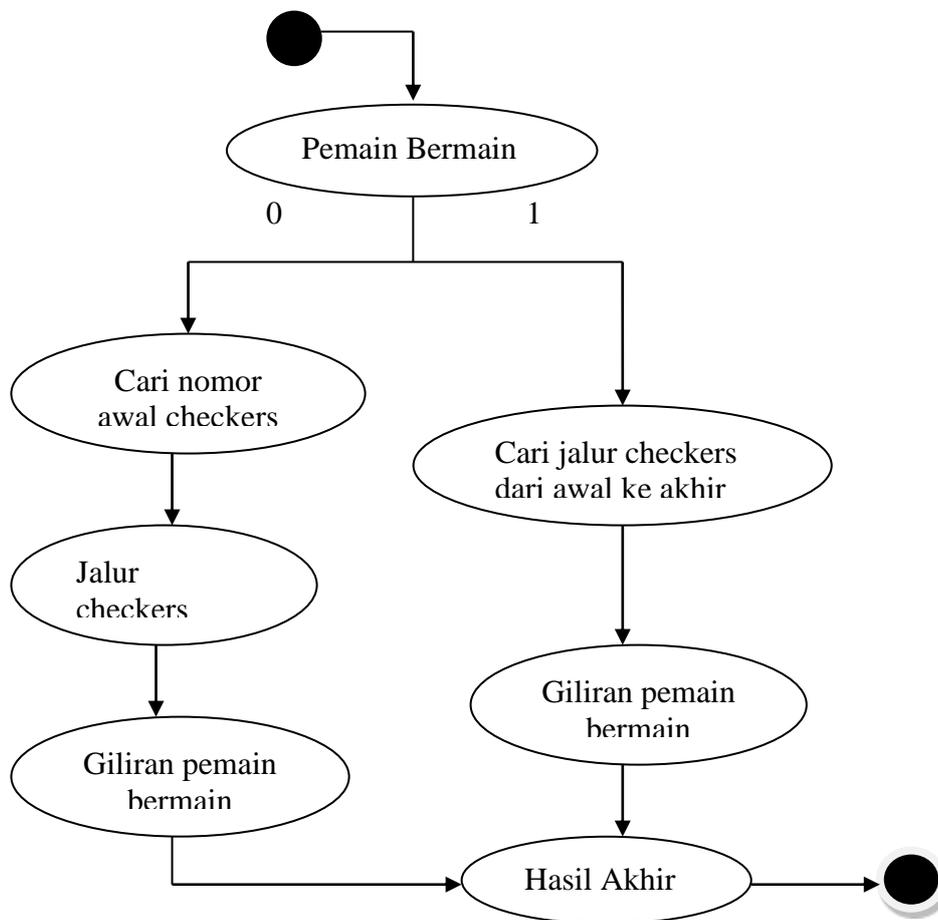
Berikut ini merupakan *activity diagram menu game* pada aplikasi permainan checkers dengan metode atau algoritma minmax yang dirancang.



**Gambar III.6 Activity Diagram Menu Permainan Checkers**

## 2. Activity Diagram Permainan Checkers

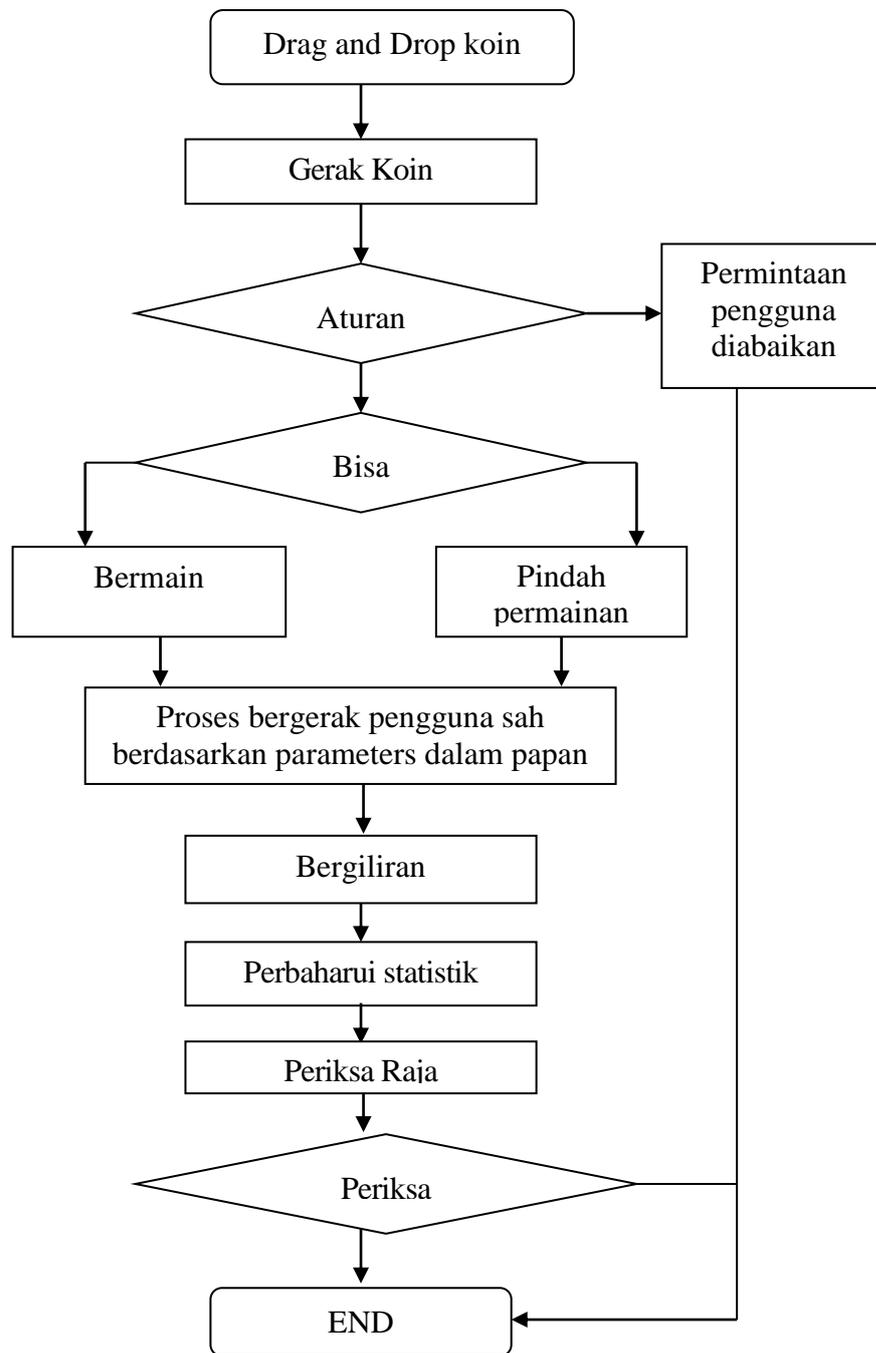
Berikut ini merupakan *activity diagram* pada aplikasi permainan checkers yang dirancang :



**Gambar III.7. Activity Diagram Aplikasi Permainan Checkers**

#### III.3.4. Flowchart Permainan Checkers

Berikut ini merupakan *flowchart permainan* pada aplikasi permainan checkers yang dirancang :



**Gambar III.8. Flowchart Aplikasi Permainan Checkers**

#### III.4. Desain User Interface

Antarmuka (*Interface*) merupakan mekanisme komunikasi antara pengguna (*user*) dengan sistem. Antarmuka (*Interface*) dapat menerima informasi

dari pengguna (*user*) dan memberikan informasi kepada pengguna (*user*) untuk membantu mengarahkan alur penelusuran masalah sampai ditemukan suatu solusi.

### 1. Rancang Tampilan Form Tampilan

Gambar dibawah merupakan rancang tampilan *Form* tampilan awal yang dirancang.

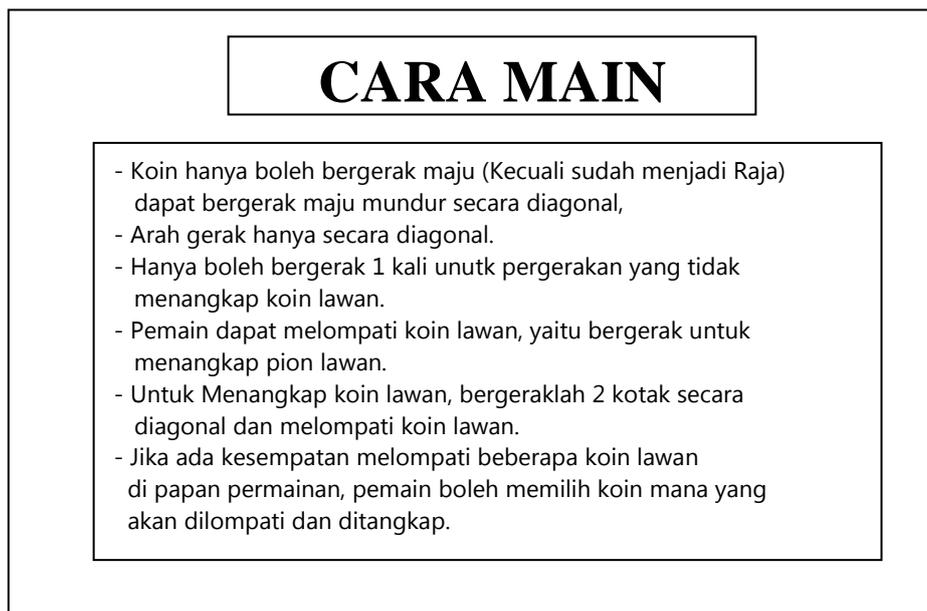


**Gambar III.9. Tampilan Halaman Awal Aplikasi Checkers**

Gambar diatas memperlihatkan judul aplikasi 4 buah tombol yaitu mulai permainan, cara main, peraturan permainan, mulai game yang digunakan untuk masuk ke dalam permainan, *Exit* digunakan untuk keluar dari aplikasi permainan.

### 2. Rancang Tampilan Form Cara Permainan Checkers

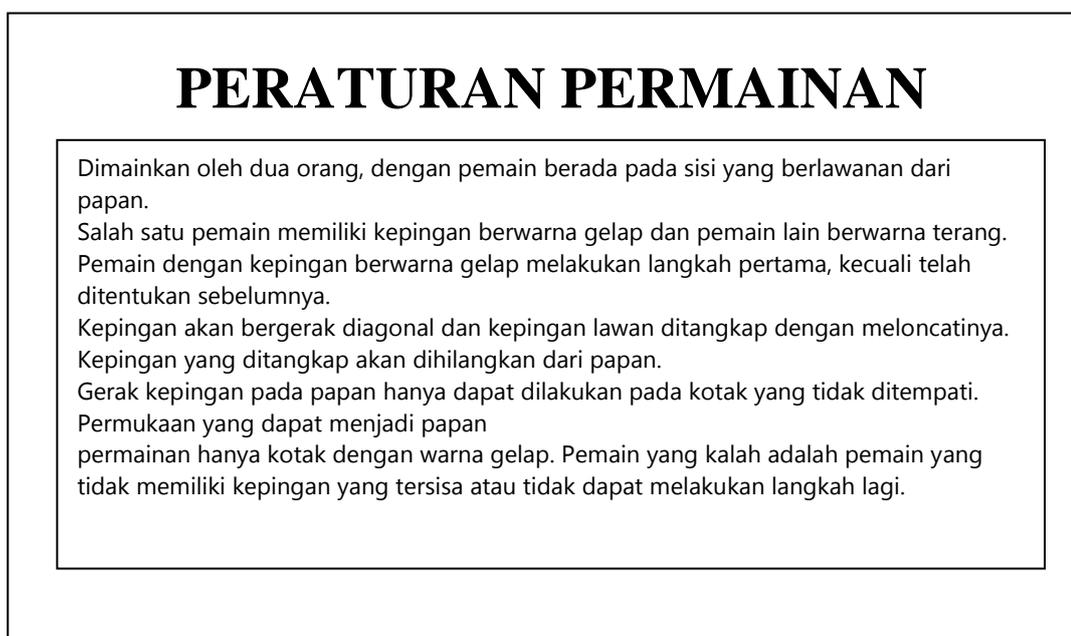
Gambar dibawah merupakan rancang tampilan cara main permainan checkers.



**Gambar III.10. Rancang Tampilan Cara Main Permainan Checkers**

### 3. Rancang Tampilan Form Peraturan Permainan Checkers

Gambar dibawah merupakan rancang tampilan peraturan permainan chackers.



**Gambar III.11. Rancang Tampilan Peraturan Permainan Checkers**

