BAB IV

PERANCANGAN

4.1 Perancangan Sistem

Perancangan adalah suatu bagian dari metodelogi pengembangan pembangunan suatu perangkat lunak yang dilakukan setelah tahapan untuk memberikan gambaran secara terperinci. Berdasarkan uraian diatas perancangan sistem merupakan tahapan dari siklus pengembangan sistem yang didefinisikan dari kebutuhan-kebutuhan fungsional dan persiapan untuk rancang bangun implementasi yang menggambarkan bagaimana suatu sistem dibentuk, yang dapat berupa penggambaran, perancangan, dan pembuatan sketsa atau pengaturan dari beberapa elemen yang terpisah kedalam satu kesatuan yang utuh dan berfungsi, juga menyangkut konfigurasi dari komponen-komponen perangkat keras dan perangkat lunak dari suatu sistem.

4.1.1 Tujuan Perancangan Sistem

Adapun tujuan dari perancangan *game puzzle* angka ini adalah untuk membuat *game puzzle* angka pada perangkat lunak yang tersedia yaitu netbeans, dan untuk mengimplementasikan algorima *Depth-First Search* yang dapat membantu dalam pemecahan solusi pada *game puzzle* angka dalam struktur ruang pohon.

Sehingga pembuatan *game puzzle* angka ini diharapkan mampu manjadi game edukasi yang bermanfaat bagi penggunanya.

4.1.2. Gambaran Umum Sistem

pembuatan *game puzzle* angka yang akan dibuat ini akan mampu menampilkan urutan angka-angka dari 1 hingga 15, yang kemudian angka-angka tersebut nantinya akan teracak secara system menjadi urutan-urutan yang tidak beraturan. Nantinya pengguna akan menyusun angka-angka yang tidak beraturan itu menjadi angka-angka yang tersusun kembali dari 1 hingga 15, atau hingga tujuan dari *game* ini ditemukan yaitu angka tersusun, jika pengguna merasa tidak menemukan jalan keluar atau tidak meneukan tujuan maka jika ingin mengakhiri permainan dapat lagsung mengklik tombol exit diatas kanan *game*. Sehingga dapat disimpulkan bahwa *game puzzle* angka ini mempunyai tujuan menyusun angka-angka yang teracak menjadi tersusun dengan benar.

4.2 Implementasi Algoritma *Depth-Frist Search*

Fungsi utama *game puzzle* angka sebagai *game* edukasi adalah untuk membatu para anak-anak dalam mengasah otak dan sebagai fasilitator dalam belajar di bidang elektronik, pembuatan *game puzzle* angka inipun tidak terlepas dari peran algoritma sebagai salah satu bentuk pemecahan masalah dari pengacakan angka tersebut agar menemukan tujuan atau *goal* dari suatu permainan dalam bentuk pohon/*tree solution*.

Adapun algoritma yang dapat di implementasikan dalam *game puzzle* angka ini dari beberapa macam algoritma yang ada penulis menggunakan algoritma *Depth-First Search* karena algoritma ini di anggap mampu di implementasikan dalam *game puzzle* angka ini.

Implementasi pertama dari algoritma *Depth-First Search* adalah masing-masing node yang ada dalam *puzzle* angka yang telah diacak semua memiliki kesempatan untuk dicari jalan keluar dengan pencarian secara vertical sehingga menemukan tujuan.

Implementasi yang kedua adalah jika hasil dari pencarian tujuan dengan menggunakan algoritma *Depth-First Search* tersebut mendapatkan hasil maka permainan *game puzzle* telah sukses, melalui proses aplikasi netbeans.

Setelah dilakukan implementasi pada *geme puzzle* dengan menggunakan algoritma tersebut yaitu *Depth-First Search* yang di terapkan pada program netbeans ini maka tujuan dari *game puzzle* angka ini akan ditemukan dan proses pencarian angka yang tersusun dengan baik telah sukses.

4.2.1. Pengembangan Cara Kerja Umum Algoritma *Depth frist search*

Pada keadaan awal puzzle angka, algoritma *Depth-First Search* akan bekerja dengan mengembangkan 1 kemugkinan arah simpul yang dapat dibangun pada suatu akar pohon menjadi keadaan baru. Kemudian selanjutnya, algoritma *Depth-First Search* akan mengembangkan lebih maju pencarian ke simpul berikutnya.

Bila tidak ada simpul yang dapat dikembangkan pada suatu jalur, maka metode *Depth-First Search* akan kembali ke jalur sebelumnya dan mengambil arah simpul yang lain dengan arah secara vertikal/horizontal yakni atas dan bawah. Keadaan ini berlanjut hingga simpul terakhir dikembangkan. Secara ringkas, algoritma proses *Depth-First Search* dapat dijabarkan sebagai berikut:

1. Masukkan simpul ujung (akar) ke dalam tumpukan
2. Ambil simpul dari tumpukan teratas, lalu cek apakah simpul merupakan solusi
3. Jika simpul merupakan solusi, pencarian selesai dan hasil dikembalikan.
4. Jika simpul bukan solusi, masukkan seluruh simpul yang bertetangga dengan simpul tersebut (simpul anak) ke dalam tumpukan
5. Jika tumpukan kosong dan setiap simpul sudah dicek, pencarian selesai dan mengembalikan hasil solusi tidak ditemukan
6. Ulangi pencarian dari langkah kedua.

Pencarian dengan metode ini dilakukan dari simpul awal secara mendalam hingga yang paling akhir atau sampai ditemukannya solusi.

Sebuah Game puzzle angka yang terdiri atas 4 buah kolom dan 4 baris di nomori 1-15.

1	10	14	15
5	11	13	7
8	2		4
6	9	3	12

Initial State

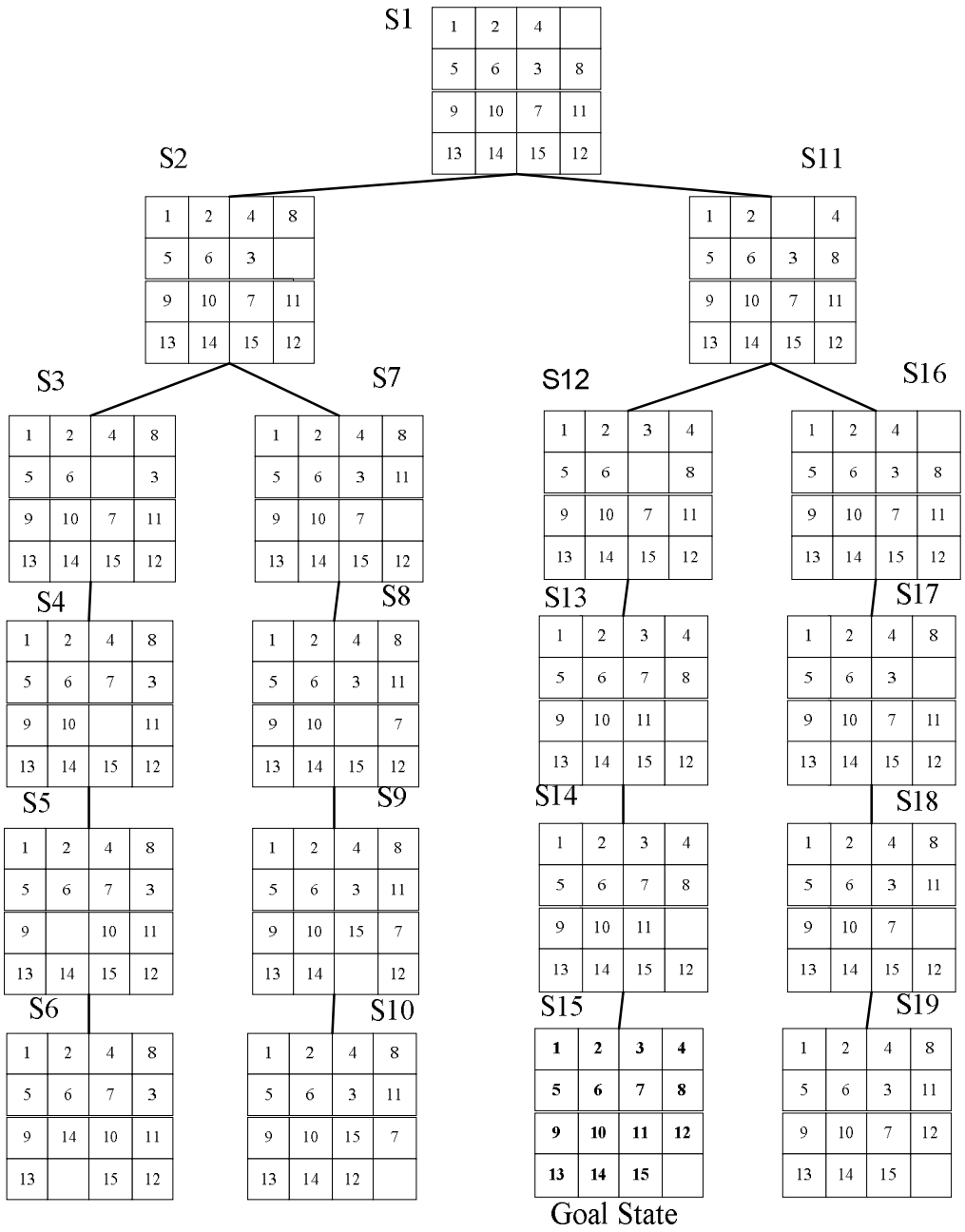
1	2	3	4
5	6	7	8
9	10	11	12
13	14	15	

Goal State

Gambar 4.1 Contoh Goal state dan Initial state

Pada gambar 4.1 di atas yang menggambarkan keadaan *game puzzle* angka yang teracak dari angka 1-15 yang disebut dengan intial state, dan pada gambar disamping nya merupakan goal state atau tujuan dari *game puzzle* agar angka tersusun secara tersusun dari angka 1-15.

Pada gambar di bawah akan dibahas mengenai struktur ruang pohon yang ditelurusi oleh algoritma *Depth-Frist Search.* Dari mulai angka yang teracak sampai dengan menemukan solusi atau tujuan dari angka yang teracak dalam bentuk struktur pohon.



Gambar 4.2 Pohon ruang status yang dibangkitkan dengan algoritma *DFS*

Penyelesaian untuk masalah permainan game puzzle angka dengan ruang status yang di bangkitkan dengan algoritma *Depth-Frist Search* pada gambar 4.2. adalah :

1. Simpul pertama (S1) dengan jumlah suksesor 2 yaitu simpul s1, dan s11 Artinya pada simpul pertama terdapat 2 kemungkinan perpindahan suatu angka yang dapat bergerak.
2. Simpul kedua (s2) merupakan simpul yang pertama ditelusuri dengan perpindahan kotak kosong ke atas. Dan pada simpul kedua terdapat 2 kemungkinan perpindahan/suksesor dari angka yang telah tersusun pada simpul ke pertama jika pada simpul pertama tidak menemukan solusi.
3. Simpul ketiga (s3) merupakan suksesor atau anak simpul dari simpul kedua, pada simpul ini dilihat perpindahan node dan jika belum ditemukan solusi maka pencarian akan berlanjut pada simpul ke empat (s4)
4. Simpul keempat (s4) merupakan suksesor atau anak simpul dari simpul ketiga, pada simpul ini dilihat perpindahan node dan jika belum ditemukan solusi maka pencarian akan berlanjut pada simpul ke lima (s5)
5. Simpul kelima (s5) merupakan suksesor atau anak simpul dari simpul keempat, pada simpul ini dilihat perpindahan node dan jika belum ditemukan solusi maka pencarian akan berlanjut pada simpul ke enam (s6)
6. Simpul keenam (s6) merupakan suksesor atau anak simpul dari simpul kelima, pada simpul ini dilihat perpindahan node dan jika belum

ditemukan solusi maka pencarian akan berlanjut pada simpul ke tujuh (s7) pada anak simpul berikutnya.

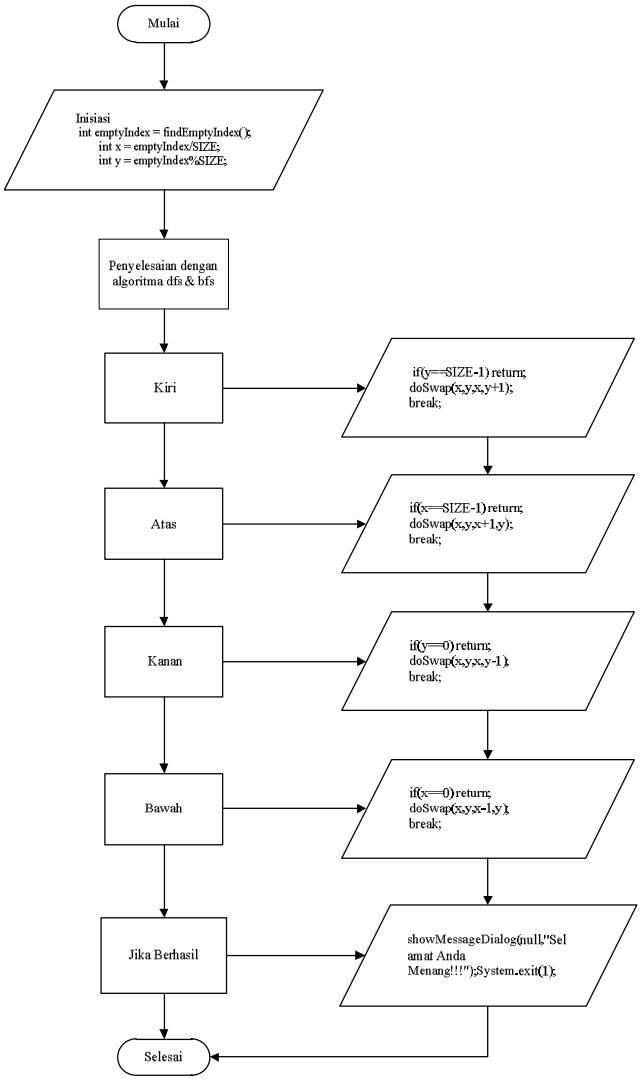
7. Simpul ke tujuh (s7) merupakan anak simpul dari simpul ke dua (s2) jika pada simpul ini masih belum ditemukan solusi maka pencarian dilanjutkan pada simpul ke delapan (s8).
8. Simpul kedelapan (s8) merupakan suksesor atau anak simpul dari simpul ketujuh, pada simpul ini dilihat perpindahan node dan jika belum ditemukan solusi maka pencarian akan berlanjut pada simpul ke sembilan (s9)
9. Simpul kesembilan (s9) merupakan suksesor atau anak simpul dari simpul kedelapan, pada simpul ini dilihat perpindahan node dan jika belum ditemukan solusi maka pencarian akan berlanjut pada simpul ke sepuluh (s10)
10. Dan begitu seterusnya sampai dengan menemukan solusi, pada struktur pohon diatas solusi/goal ditemukan pada simpul ke lima belas (s15).

4.3 Perancangan Prosedur

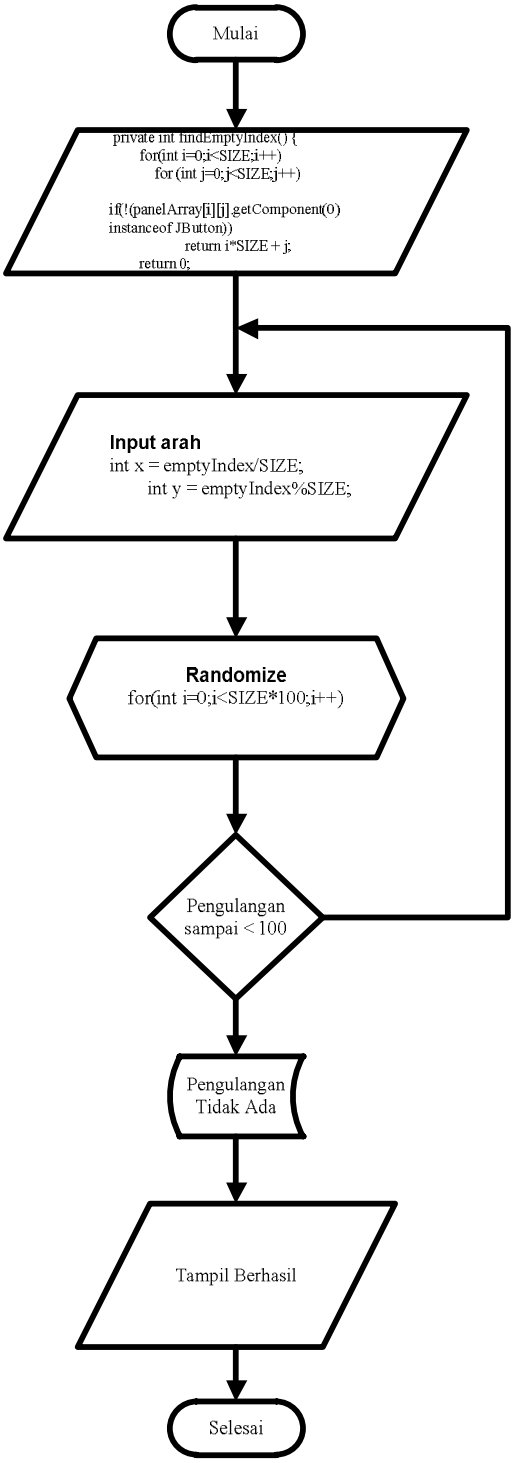
Proses perancangan ini merupakan tahap awal dari perancangan *game puzzle* angka yang dilakukan sebagai pemecahan masalah yang ada pada proses *game puzzle* angka ketika dimainkan.

4.3.1 Flow Chart

Flowchart adalah penggambaran secara grafik dari langkah-langkah dan urutan-urutan prosedur dari suatu program. Flowchart menolong analis dan programmer untuk memecahkan masalah kedalam segmen-segmen yang lebih kecil dan menolong dalam menganalisis alternatif-alternatif lain dalam pengoperasian.



Gambar 4.3 Flowchart Algoritma Untuk Mencari Solusi



Gambar 4.4 Flowchart Acak Angka

4.4 Perancangan Program

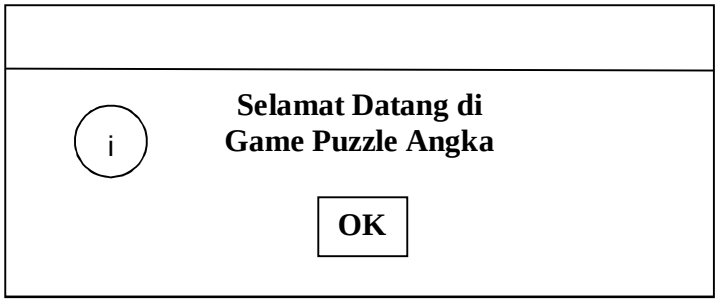
Program dapat didefinisikan sebagai kumpulan dari intruksi-intruksi atau perintah-perintah terperinci yang sudah disiapkan oleh komputer sehingga dapat melakukan fungsi sesuai dengan yang telah ditentukan. Tujuan dari pembuatan *game puzzle* angka ini adalah untuk mengetahui solusi yang dapat memecahkan tujuan dari *game puzzle* angka yang telah diacak dengan menggunakan metode yang telah ditentukan.

4.4.1 Perancangan Antar Muka

Perancangan antar muka atau yang biasa disebut dengan istilah interface merupakan bentuk dari tampilan perangkat lunak dengan user atau pengguna perangakat lunak tersebut. Perancangan ini dibuat sedemikian rupa untuk memperoleh hasil yang optimal dengan berbagai aturan-aturan tertentu yang berlaku pada *game puzzle* angka ini agar dapat menghasilkan antar muka antara perangkat lunak dan pemakai yang baik dan benar.

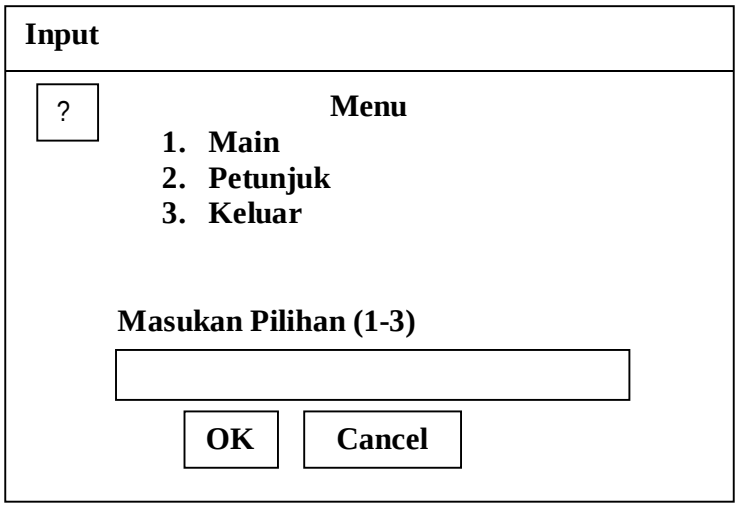
Karena *game puzzle* angka ini dibuat praktis untuk anak usia 6-8 tahunan maka perancangan antar muka yang digunakan tidak dibuat serumit *game puzzle* pada umumnya. perancangan yang digunakan untuk pengguna *game* hanya menekan tombol keyboard menggunakan tombol panah atas, bawah, kanan, dan kiri sebagai tombol untuk mengatur gerak kotak angka/mengeser-geser angka untuk menyusun kembali.

Dibawah ini merupakan perancangan antar muka yang penulis desain pada *game puzzle* angka pada gambar 4.4.1



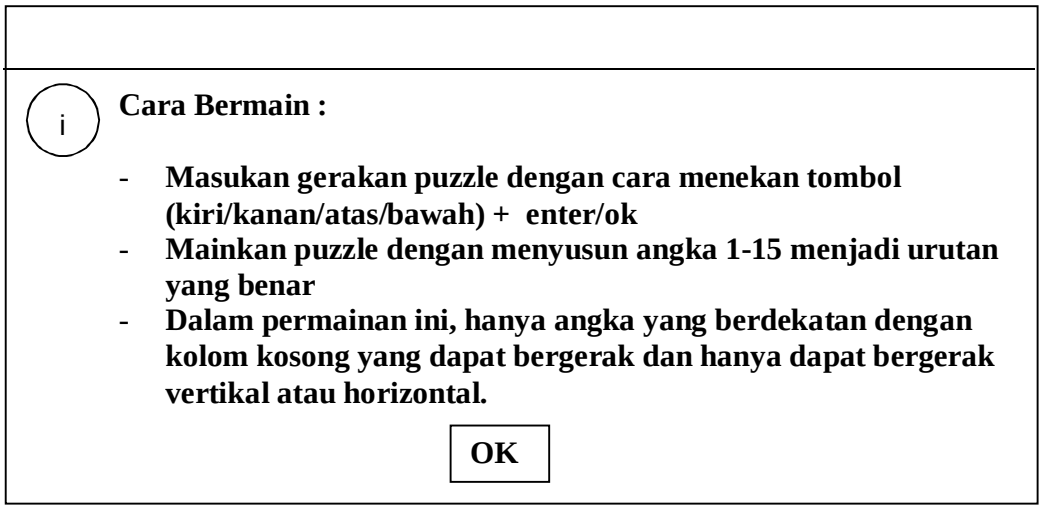
Gambar 4.5 Tampilan Awal *Game Puzzle*

Pada gambar di atas terlihat tampilan awal permainan *game puzzle* angka yang dirancang oleh penulis pada *game puzzle* angka, pada tampilan seperti di atas pun di implementasikan pada program netbeans agar perancangan tersebut sesuai dengan apa yang telah dirancangan sebelumnya.



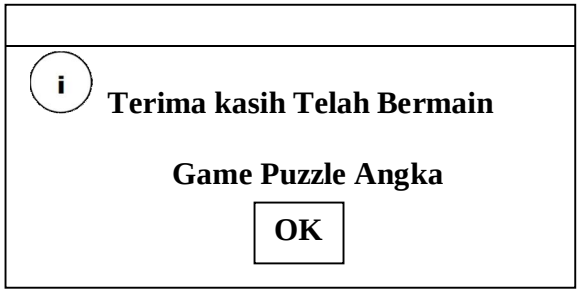
Gambar 4.6 Tampilan Menu

Setelah tampilan awal muncul dan pengguna menggunakan tombol enter maka selanjutnya tampilan menu yang akan muncul. Disini digunakan untuk memberikan petunjuk bagi pengguna ataupun dapat langsung memainkan dengan menekan tombol angka 1-3 pada kolom yang telah disediakan lalu pilih ok.



Gambar 4.7 Tampilan Petunjuk

Pada gambar di atas merupakan gambar tampilan petunjuk yang dapat memberikan informasi cara penggunaan *game* bagi pengguna *game puzzle* angka ini sehingga pengguna tidak merasa kesulitan dalam mencari tombol apa yang diperlukkan untuk memainkan *game puzzle* angka tersebut.

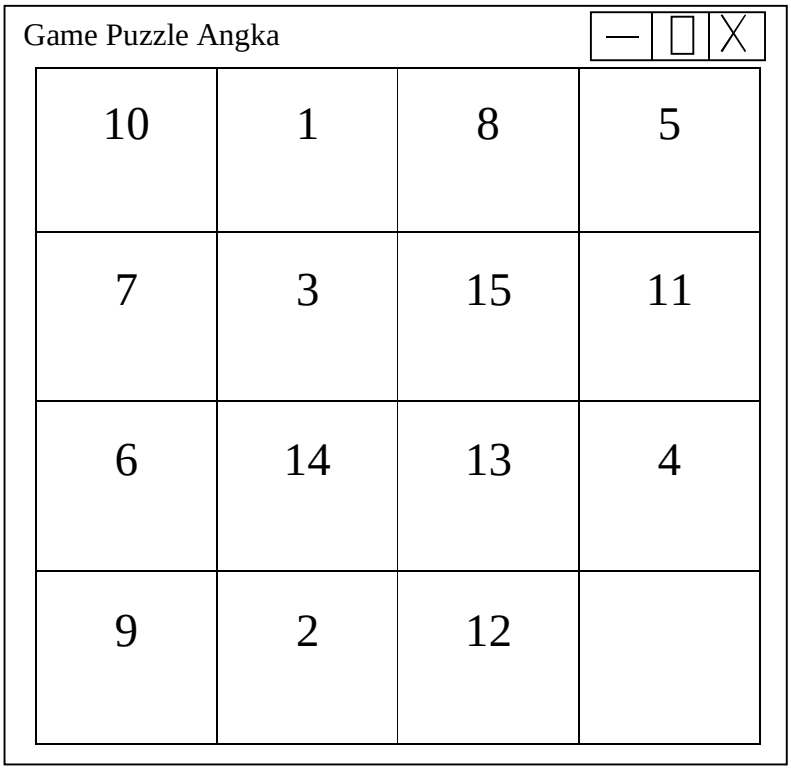


Gambar 4.8 Tampilan Keluar

Gambar diatas merupakan tampilan jika pengguna menekan tombol 3 yang berati keluar dari game puzzle angka tersebut dan tidak memainkan *game puzzle* angka.

4.4.2 Perancangan Game Mengacak dan Menyusun Angka

Setelah penulis melakukan perancangan untuk tampilan antar muka maka selanjutnya penulis melakukan perancangan untuk tampilan *game* pada saat pengguna/user melakukan pengacakan pada angka *puzzle* dan menyusunnya hingga berurutan.



Gambar 4.9 Tampilan Awal Permainan

Setelah teracak seperti gambar di atas maka pengguna/user tinggal menekan tombol tombol panah atas, bawah, kanan, dan kiri sebagai tombol untuk mengatur gerak kotak angka/mengeser-geser angka untuk menyusun kembali sebagai tombol utama dalam penggerakan dalam pemindahan kotak-kotak *puzzle* angka. Dengan keterangan sebagai berikut :

Panah kiri = gerak ke kiri, panah bawah = gerak ke bawah, panah kanan = gerak ke kanan, panah atas = gerak ke atas.