

PERANCANGAN APLIKASI UNTUK MENYELESAIKAN RUBIK DENGAN METODE LAYER BY LAYER

Tranius Eko¹, Riyadi J. Iskandar², Antonius³

^{1,2,3}Teknik Informatika, STMIK Widya Dharma, Pontianak

e-mail: ¹traniuseko@gmail.com, ²riyadijiskandar@gmail.com, ³antoniusok@yahoo.com

Abstract

Rubik's Cube has so many combinations of positions that may occur, so it is difficult to solve with random rotation without using the exact algorithm. Along with the development of technology, computers have been able to do some work as well as that done by humans, no exception to resolve complex problems though. Therefore, the authors intend to design an application to guide beginners in solving Rubik's cube with layer by layer method. The preparation of this study using a Causal Relationship Research Design as study design, while the data collection methods used by the author is the study of literature and data analysis techniques using diagrams of the Unified Modelling Language (UML), then the system is designed using Microsoft Visual C # 2010 and the database using Microsoft Office Access 2010. This research resulted in an application that can guide beginners in solving the Rubik's cube with layer by layer method in accordance with the original purpose of this study. The conclusion from this study is that the resulting application can help users in solving the Rubik's cube with layer by layer method and Rubik's cube that can be solved is 3x3x3 Rubik's cube. In addition, the application also gives instruction about notation and algorithms used in solving Rubik's cube itself.

Keywords: Application, Rubik, Layer by Layer Method

Abstrak

Kubus rubik mempunyai begitu banyak kombinasi posisi yang mungkin terjadi sehingga sangat sulit untuk menyelesaikannya dengan perputaran acak tanpa menggunakan metode dengan algoritma yang pasti. Seiring dengan berkembangannya teknologi, komputer telah mampu melakukan beberapa pekerjaan seperti dan sebaik yang dilakukan oleh manusia, tak terkecuali untuk menyelesaikan permasalahan rumit sekalipun. Oleh sebab itu penulis bermaksud untuk merancang suatu aplikasi yang dapat menuntun para pemula dalam menyelesaikan kubus rubik dengan metode layer by layer. Penyusunan penelitian ini menggunakan Desain Penelitian Hubungan Kausal sebagai rancangan penelitian, sedangkan metode pengumpulan data yang digunakan penulis adalah studi literatur dan teknik analisis data menggunakan diagram Unified Modelling Language (UML), kemudian sistem dirancang dengan menggunakan Microsoft Visual C# 2010 dan database menggunakan Microsoft Office Access 2010. Penelitian ini menghasilkan suatu aplikasi yang dapat menuntun para pemula dalam menyelesaikan kubus rubik dengan metode layer by layer sesuai dengan apa yang menjadi tujuan awal dari penelitian ini. Kesimpulan dari penelitian ini adalah bahwa aplikasi yang dihasilkan dapat membantu pengguna dalam menyelesaikan rubik dengan metode layer by layer dan rubik yang dapat diselesaikan adalah kubus rubik 3x3x3. Selain itu aplikasi juga memberi pembelajaran tentang notasi dan algoritma yang digunakan dalam menyelesaikan rubik itu sendiri.

Kata Kunci: Aplikasi, Rubik, Metode Layer by Layer

1. PENDAHULUAN

Kubus rubik adalah sebuah *puzzle* mekanis yang diciptakan oleh seorang profesor arsitektur asal negara Hungaria yang bernama Erno Rubik pada tahun 1974. Kubus rubik sendiri mempunyai ukuran standar 3x3x3 yang dimana merupakan kubus rubik yang paling digemari dan paling banyak dimainkan oleh orang di seluruh dunia dan oleh sebab itulah penelitian ini memilih rubik 3x3x3 sebagai objek penelitian walaupun sebenarnya masih banyak jenis lain seperti rubik 2x2x2 (*pocket cube*), rubik 4x4x4 (*rubik's revenge*), rubik 5x5x5 (*professor's cube*) dan ukuran yang lebih besar lainnya.

Permainan rubik prinsipnya sangat sulit untuk diselesaikan dikarenakan memiliki kerumitan yang begitu kompleks. Kubus rubik mempunyai begitu banyak kombinasi posisi yang mungkin terjadi sehingga sangat sulit untuk menyelesaikannya dengan perputaran acak tanpa menggunakan metode dengan algoritma yang pasti. Sampai saat ini sudah banyak metode yang ditemukan untuk menyelesaikan kubus rubik, beberapa

diantaranya yang cukup populer adalah metode *layer by layer*, metode *fridrich*, metode *petrus* dan metode *waterman*. Masing-masing dari metode di atas mempunyai caranya sendiri untuk menyelesaikan sebuah kubus rubik dari keadaan warna yang teracak menjadi tersusun seperti semula. Metode *layer by layer* sendiri menyelesaikan kubus rubik dengan cara membentuk pola *cross* pada lapisan pertama, kemudian menempatkan biji pojoknya pada posisi yang benar, lalu menyelesaikan biji tepi pada lapisan kedua dan yang terakhir menyelesaikan lapisan ketiga dengan sempurna. Metode *layer by layer* digunakan dalam penelitian ini karena merupakan metode yang paling mudah untuk dipelajari oleh pemula dibandingkan dengan metode-metode lainnya.

Seiring dengan berkembangannya teknologi, komputer telah mampu melakukan beberapa pekerjaan seperti dan sebaik yang dilakukan oleh manusia, tak terkecuali untuk menyelesaikan permasalahan rumit sekalipun. Oleh sebab itu penulis bermaksud untuk merancang suatu aplikasi yang mampu menyelesaikan rubik dengan cara mengadopsi pengetahuan penulis guna untuk menuntun para pemula dalam menyelesaikan permainan rubik itu sendiri.

Tidak seperti penelitian sebelumnya yang membahas tentang *Optimal Solutions For Solve Rubik's Cube*^[1], penelitian yang dilakukan penulis hanya untuk menuntun para pemula dalam menyelesaikan rubik. Hal tersebut karena penulis beranggapan bahwa para pemula belum tepat untuk memikirkan solusi optimal untuk menyelesaikan rubik melainkan harus memikirkan terlebih dahulu bagaimana cara menyelesaikannya.

2. METODE PENELITIAN

2.1. Bentuk penelitian dan teknik pengumpulan data yang digunakan adalah:

2.1.1. Rancangan Penelitian

Penyusunan penelitian ini menggunakan Desain Penelitian Hubungan Kausal (Eksperimental) dimana penulis membaca, mempelajari dan memahami semua literatur yang berhubungan dengan materi penelitian.

2.1.2. Metode Pengumpulan Data

Metode pengumpulan data yang digunakan penulis adalah Studi literatur, yaitu Penulis mencari data dan teori dari berbagai literatur yang berkaitan dengan materi penelitian.

2.1.3. Teknik Analisis Data

Teknik analisis data yang digunakan oleh penulis dalam penelitian ini adalah diagram *Unified Modelling Language* (UML).

2.1.4. Teknik Perancangan Sistem

Pada penelitian ini sistem dirancang dengan menggunakan *Microsoft Visual C# 2010* dan *database* menggunakan *Microsoft Office Access 2010*.

2.2. Landasan Teori

2.2.1. Aplikasi

Aplikasi merupakan *software* yang dirancang khusus untuk keperluan tertentu.^[2]

2.2.2. Metode

Metode ialah cara untuk mencapai sesuatu.^[3]

2.2.3. User Interface

User interface/antarmuka pengguna merupakan penghubung antara aplikasi dengan pengguna, karena itu perlu dibuat antarmuka yang *user friendly*/bersahabat dengan pengguna, produktif dan mudah dikelola.^[4]

2.2.4. Perancangan Sistem

Perancangan sistem adalah proses pengembangan spesifikasi sistem baru berdasarkan hasil rekomendasi analisis sistem.^[5]

2.2.4.1. Perancangan Input

Perancangan *input* merupakan perancangan tampilan antarmuka dimana pengguna bisa memasukan input baik berupa data melalui keyboard maupun melalui perangkat lain.^[6]

2.2.4.2. Perancangan Output

Perancangan *output* merupakan perancangan yang berkaitan dengan informasi hasil pengolahan data.^[7]

2.2.4.3. Perancangan Basis Data

Perancangan basis data merupakan proses untuk menentukan isi dan pengaturan data yang dibutuhkan untuk mendukung berbagai perancangan sistem.^[8]

2.2.5. Kubus Rubik

Rubik's cube atau kubus rubik adalah sebuah *puzzle* mekanis yang diciptakan oleh seorang profesor arsitektur asal negara hungaria yang bernama erno rubik pada tahun 1974.^[9]

2.2.6. Metode Layer by Layer

Terdapat Beragam metode untuk menyelesaikan *rubik's cube*, namun metode yang paling mudah dan populer untuk dipelajari oleh pemula adalah *layer by layer method* atau metode lapis demi lapis.^[10]

2.2.7. Unified Modeling Language (UML)

Unified Modeling Language (UML) adalah bahasa pemodelan untuk sistem atau perangkat lunak yang berparadigma berorientasi objek.^[11]

2.2.8. *Microsoft Visual C#*

Bahasa Pemrograman C# merupakan gabungan dari kecanggihan bahasa keluarga C (C, C++, *Objective-C*, *Java* dan sebagainya) dengan kemudahan pemrograman bahasa *Visual Basic*.^[12]

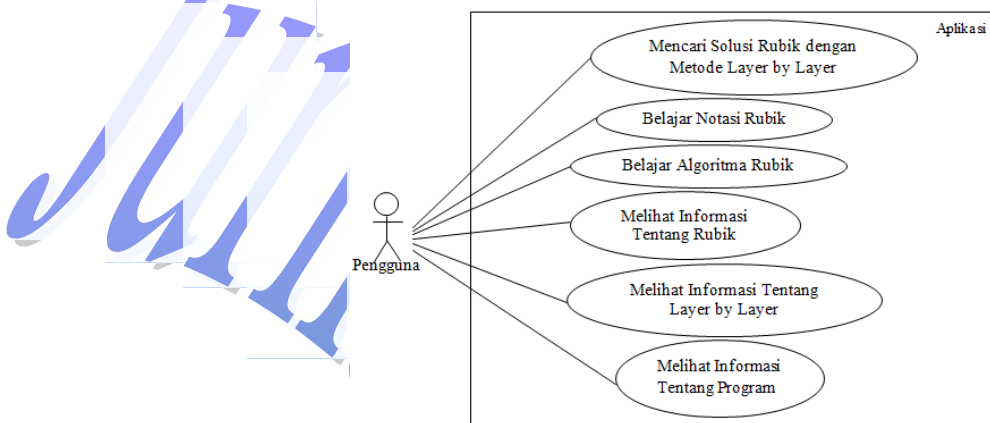
2.2.9. *Microsoft Office Access*

Microsoft Office Access merupakan sistem manajemen *database* dari *Microsoft* untuk merancang dan mengelola *database*.^[13]

3. HASIL DAN PEMBAHASAN

3.1. *Use Case Diagram*

Use case diagram menggambarkan interaksi antara pengguna sebagai actor dengan aplikasi yang dibangun. Berikut adalah *use case diagram* yang digunakan dalam perancangan aplikasi.

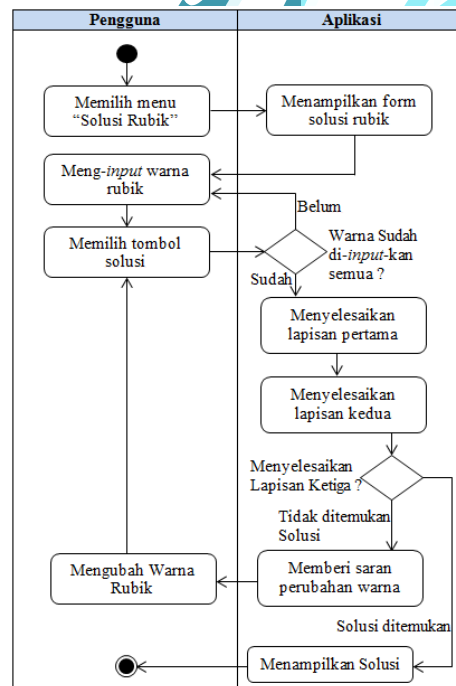


Gambar 1. *Use case diagram* aplikasi untuk menyelesaikan rubik dengan metode *layer by layer*

3.2. *Activity Diagram*

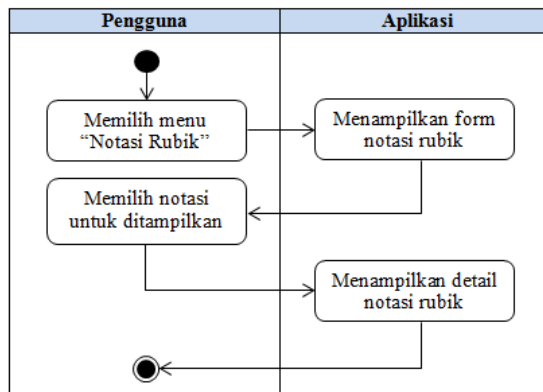
Activity diagram digunakan untuk menggambarkan aktivitas yang terjadi pada aplikasi yang dibangun. *Activity diagram* menggambarkan proses-proses yang terjadi pada saat aktivitas dimulai hingga aktivitas berhenti. Berikut adalah *activity diagram* yang digunakan dalam perancangan aplikasi:

3.2.1. *Activity Diagram* Mencari Solusi Rubik dengan Metode *Layer by Layer*



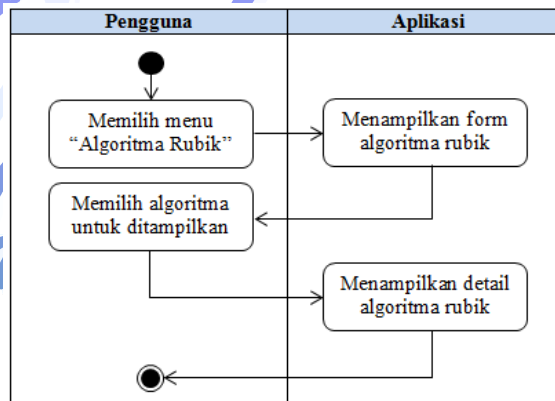
Gambar 2. *Activity Diagram* Mencari Solusi Rubik dengan Metode *Layer by Layer*

3.2.2. Activity Diagram Belajar Notasi Rubik



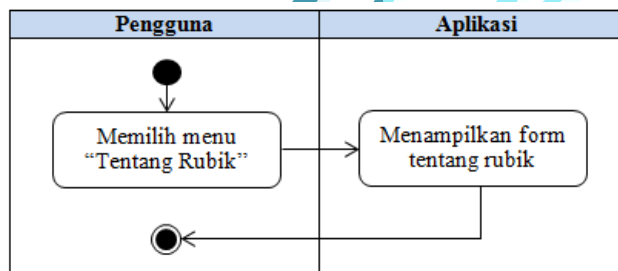
Gambar 3. Activity Diagram Belajar Notasi Rubik

3.2.3. Activity Diagram Belajar Algoritma Rubik



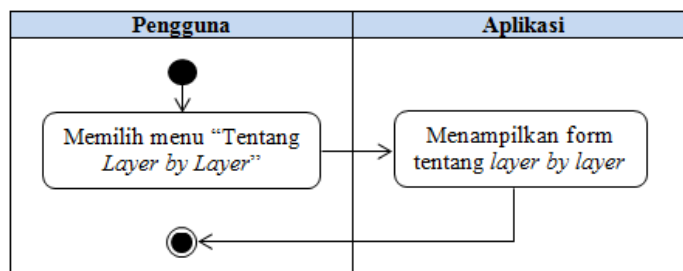
Gambar 4. Activity Diagram Belajar Algoritma Rubik

3.2.4. Activity Diagram Melihat Informasi Tentang Rubik



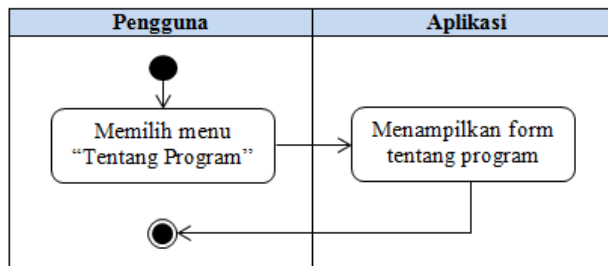
Gambar 5. Activity Diagram Melihat Informasi Tentang Rubik

3.2.5. Activity Diagram Melihat Informasi Tentang Layer by Layer



Gambar 6. Activity Diagram Melihat Informasi Tentang Layer by Layer

3.2.6. Activity Diagram Melihat Informasi Tentang Program

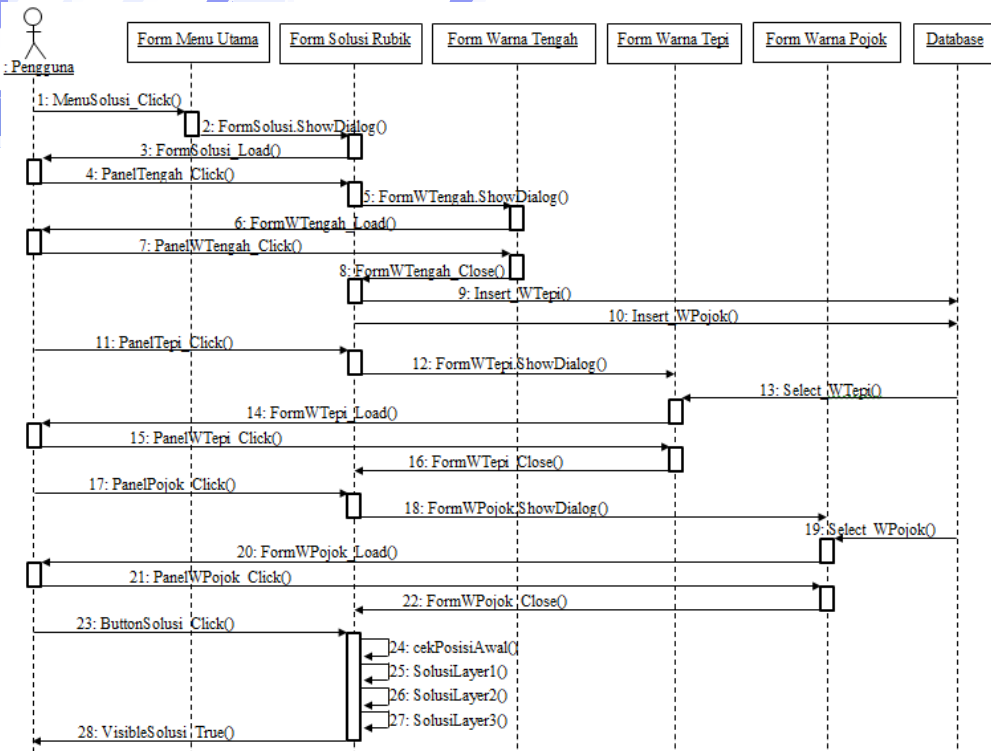


Gambar 7. Activity Diagram Melihat Informasi Tentang Program

3.3. Sequence Diagram

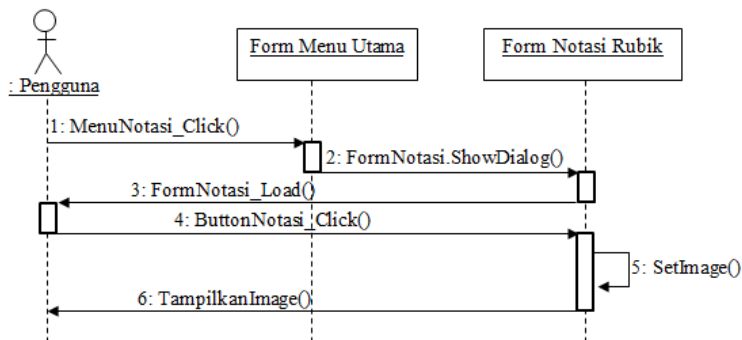
Sequence diagram akan memperlihatkan/menampilkan interaksi-interaksi antar objek di dalam sebuah aplikasi yang dibangun pada sebuah urutan atau rangkaian waktu. Berikut adalah sequence diagram yang digunakan dalam perancangan aplikasi:

3.3.1. Sequence Diagram Mencari Solusi Rubik dengan Metode Layer by Layer



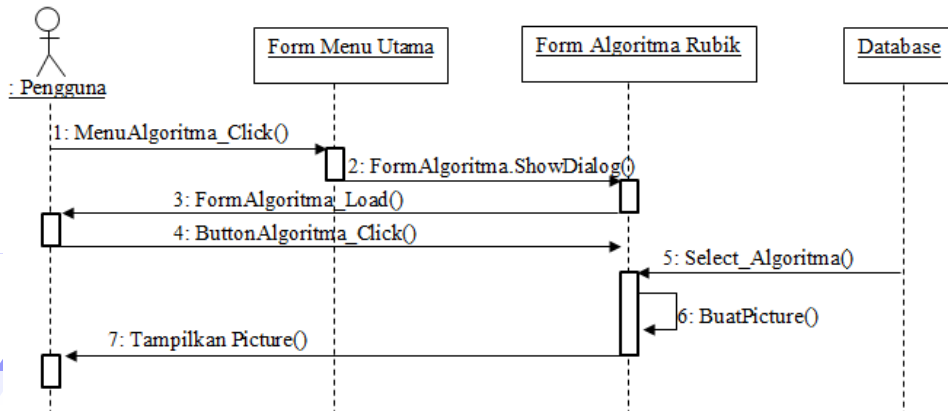
Gambar 8. Sequence Diagram Mencari Solusi Rubik dengan Metode Layer by Layer

3.3.2. Sequence Diagram Belajar Notasi Rubik



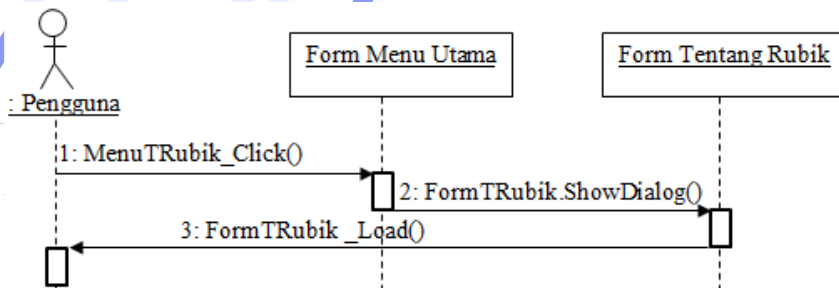
Gambar 9. Sequence Diagram Belajar Notasi Rubik

3.3.3. Sequence Diagram Belajar Algoritma Rubik



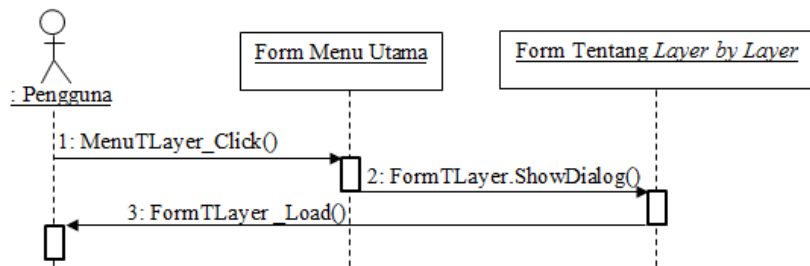
Gambar 10. Sequence Diagram Belajar Algoritma Rubik

3.3.4. Sequence Diagram Melihat Informasi Tentang Rubik



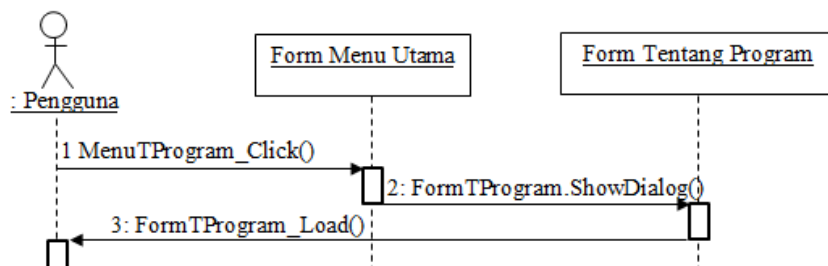
Gambar 11. Sequence Diagram Melihat Informasi Tentang Rubik

3.3.5. Sequence Diagram Melihat Informasi Tentang Layer by Layer



Gambar 12. Sequence Diagram Melihat Informasi Tentang Layer by Layer

3.3.6. Sequence Diagram Melihat Informasi Tentang Program



Gambar 13. Sequence Diagram Melihat Informasi Tentang Program

3.4. Tampilan Aplikasi

3.4.1. Tampilan Form Menu Utama

Pada saat pengguna menjalankan aplikasi, yang pertama tampil adalah form menu utama. Pada form menu utama terdapat 6 *picturebox* yang berfungsi sebagai button untuk memanggil form lainnya.



Gambar 14. Tampilan Form Menu Utama

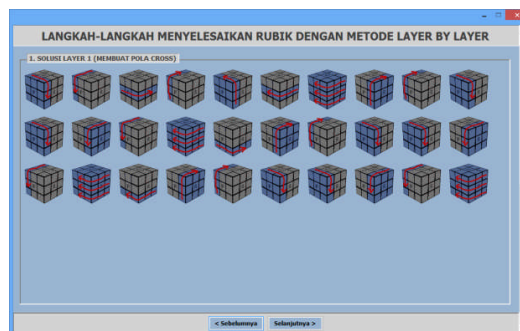
3.4.2. Tampilan Form Solusi Rubik



Gambar 15. Tampilan Form Solusi Rubik

Pada form solusi rubik terdapat 54 *panel* kosong yang dimana dapat diganti warnanya sesuai dengan warna rubik pengguna. Untuk dapat mengatur semua warna pada *panel*, pengguna harus terlebih dahulu mengatur 6 warna tengah dari masing-masing sisinya. Setelah warna diatur semua, pengguna dapat memilih tombol solusi untuk melihat solusi yang akan diberikan oleh aplikasi guna untuk menyelesaikan permasalahan rubik mereka. Pada form solusi rubik juga terdapat tombol-tombol notasi yang terletak di sebelah kanan form. Jika pengguna ingin menguji sistem dengan mengacak rubik secara *random* maka dapat menggunakan tombol-tombol notasi di sebelah kanan tersebut. Selain itu terdapat pula tombol “atur ulang warna”, tombol ini berfungsi untuk merubah warna 54 *panel* rubik ke warna dasar seperti pertama kali form ditampilkan. Lalu yang terakhir ada tombol “bantuan”, tombol ini digunakan jika pengguna bingung atau mengalami kesulitan dalam mengisi warna pada panel rubik atau bingung apa yang seharusnya dilakukan pada form solusi rubik.

3.4.3. Tampilan Solusi Pada Form Solusi Rubik



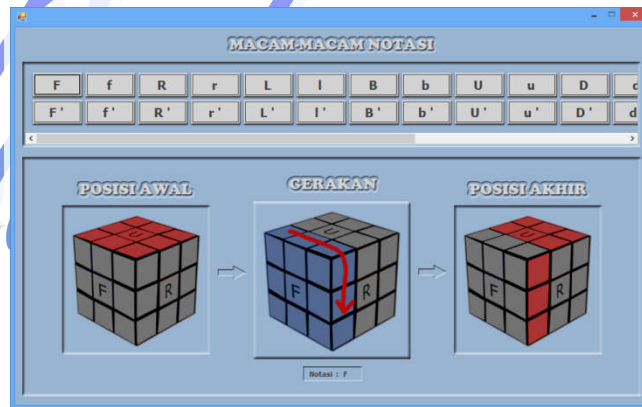
Gambar 16. Tampilan Solusi Pada Form Solusi Rubik

3.4.4. Tampilan Form Bantuan



Gambar 17. Tampilan Form Bantuan

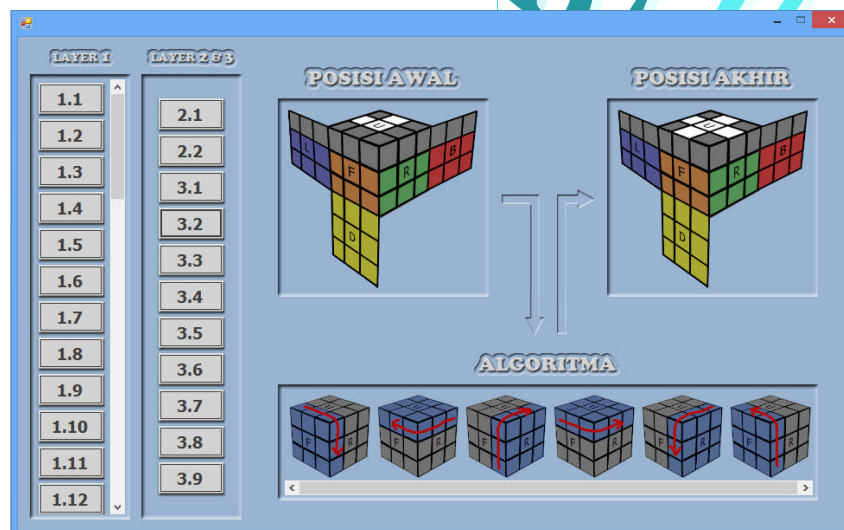
3.4.5. Tampilan Form Notasi Rubik



Gambar 18. Tampilan Form Notasi Rubik

Form notasi rubik adalah form pembelajaran notasi-notasi yang ada pada rubik, baik notasi yang digunakan maupun yang tidak digunakan dalam metode *layer by layer*. Pengguna cukup memilih notasi mana yang ingin dilihat pada bagian atas form untuk melihat bagaimana perputaran notasi dan perubahan yang terjadi setelah notasi dilakukan.

3.4.6. Tampilan Form Algoritma Rubik



Gambar 19. Tampilan Form Algoritma Rubik

Form algoritma rubik adalah form pembelajaran algoritma-algoritma yang digunakan dalam metode *layer by layer*. Pengguna cukup memilih algoritma mana yang ingin dilihat pada bagian kiri form untuk melihat bagaimana perubahan yang terjadi setelah algoritma dilakukan dan apa saja notasi yang digunakan.

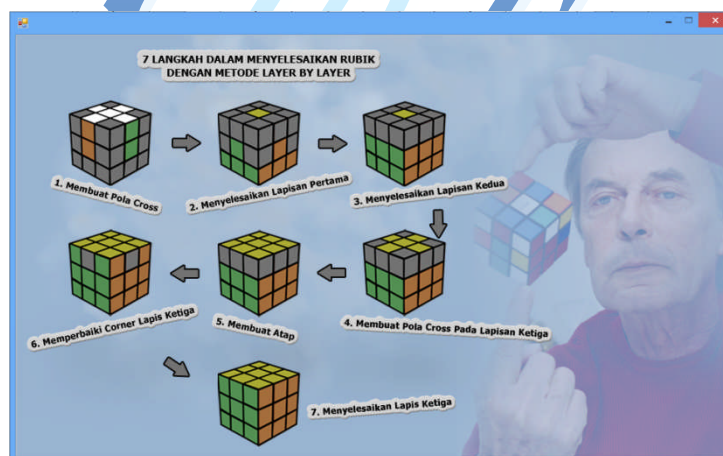
3.4.7. Tampilan Form Tentang Rubik



Gambar 20. Tampilan Form Tentang Rubik

Form tentang rubik hanya berisi *picturebox* yang menampilkan informasi mengenai siapa yang menemukan rubik, kapan dan dimana rubik itu ditemukan serta apa saja komponen penyusun rubik itu sendiri. Pengguna tidak dapat melakukan kegiatan lain pada form ini selain melihat informasi tentang rubik.

3.4.8. Tampilan Form Tentang *Layer By Layer*



Gambar 21. Tampilan Form Tentang *Layer By Layer*

Sama seperti form tentang rubik, form tentang *layer by layer* juga hanya berisi *picturebox* yang menampilkan informasi singkat kepada pengguna mengenai langkah-langkah untuk menyelesaikan rubik dengan metode *layer by layer*.

4. KESIMPULAN

Kesimpulan yang dapat diambil berdasarkan penjelasan-penjelasan sebelumnya mengenai aplikasi untuk menyelesaikan rubik dengan metode *layer by layer* adalah sebagai berikut:

- Kubus rubik mempunyai begitu banyak kombinasi posisi yang mungkin terjadi sehingga sangat sulit untuk menyelesaikannya dengan perputaran acak tanpa menggunakan metode dengan algoritma yang pasti.
- Aplikasi dapat membantu pengguna dalam menyelesaikan rubik dengan metode *layer by layer* dan rubik yang dapat diselesaikan hanya kubus rubik 3x3x3.
- Selain membantu dalam menyelesaikan rubik, aplikasi juga memberi pembelajaran tentang notasi dan algoritma yang digunakan dalam menyelesaikan rubik itu sendiri.

5. SARAN

Adapun beberapa saran yang penulis rasa perlu untuk disampaikan berdasarkan kesimpulan-kesimpulan di atas adalah sebagai berikut:

- a. Pengguna harus meng-input-kan warna dengan benar agar dapat mendapatkan solusi yang benar pula.
- b. Untuk pengembangan selanjutnya, aplikasi tidak hanya terpaku pada satu metode saja melainkan dapat memilih dari beberapa metode yang ada.
- c. Pengembangan selanjutnya juga harus dapat menyelesaikan beberapa jenis rubik, tidak hanya kubus rubik 3x3x3.

UCAPAN TERIMA KASIH

Dalam penulisan skripsi ini, penulis telah banyak mendapat bantuan berupa bimbingan, petunjuk, saran maupun dorongan moril dari berbagai pihak, maka pada kesempatan ini penulis mengucapkan terima kasih yang sebesar-besarnya kepada seluruh civitas akademika Sekolah Tinggi Manajemen Informatika dan Komputer Widya Dharma Pontianak.

DAFTAR PUSTAKA

- [1] Barzegar, Behnam, Hojjat Akbarnezhad Saravi, dan Saeid Akbaryan Saravi. (November 2011). "Optimal Solutions For Solve Rubik's Cube". *Australian Journal of Basic and Applied Sciences*. Vol. 5, No. 11: hal. 1225-1235.
- [2] Naproni. (2007). *Membangun LAN dengan Windows XP*. PT Elex Media Komputindo. Jakarta.
- [3] Gulo, W. (2007). *Strategi Belajar Mengajar*. Grasindo. Jakarta.
- [4] Kusumo, Ario Suryo. (2006). *Pemrograman Visual Basic 2005*. PT Elex Media Komputindo. Jakarta.
- [5] Kusrini, Andri Koniyo. (2007). *Tuntunan Praktis Membangun Sistem Informast Akuntansi dengan Visual Basic dan Microsoft SQL Server*. Andi. Yogyakarta.
- [6] Al Fatta, Hanif. (2009). *Rekayasa Sistem Pengenalan Wajah*. Andi. Yogyakarta.
- [7] Kusrini, Andri Koniyo. (2007). *Tuntunan Praktis Membangun Sistem Informasi Akuntansi dengan Visual Basic dan Microsoft SQL Server*. Andi. Yogyakarta.
- [8] Simarmata, Janner. (2007). *Perancangan Basis Data*. Andi. Yogyakarta.
- [9] Brata, Abel. (2010). *Langkah Mudah Menjadi Master Rubik*. Bukune. Jakarta.
- [10] Adi, Wicaksono. (2009). *Tip & Trik Jago Main Rubik*. Gradien Mediatama. Yogyakarta.
- [11] Nugroho, Adi. (2010). *Mengembangkan Aplikasi Basis Data Menggunakan C# dan SQL Server*. Andi. Yogyakarta.
- [12] _____. (2010). *Rekayasa Perangkat Lunak Berorientasi Objek dengan Metode USDP*. Andi. Yogyakarta.
- [13] Hasym. (2009). *Buku Pintar Microsoft Office*. Kriya Pustaka. Jakarta.