



JIS

JURNAL
INFORMATION SYSTEM



SUSUNAN DEWAN REDAKSI JURNAL INFORMATION SYSTEM

Pemimpin Redaksi

Elmi Devia, S.Kom., M.Kom. (Universitas Krisnadwipayana, *Indonesia*)

Tim Redaksi

Ir. Junaidi, M.Kom. (Universitas Krisnadwipayana, *Indonesia*)

M. Syarif Hartawan, S.Kom., M.Kom. (Universitas Krisnadwipayana, *Indonesia*)

Wargijono Utomo, S.T., S.Kom., M.Kom. (Universitas Krisnadwipayana, *Indonesia*)

Tim Reviewer

Dr. Fauziah, S.Kom., M.M.S.I. (Universitas Nasional, *Indonesia*)

Prof. Dr. Hoga Saragih, S.T., M.T. (Universitas Bakri, *Indonesia*)

Dr. Hendy Tannady, S.T., M.T., M.M., M.B.A. (Kalbis Institute, *Indonesia*)

Dr. Herwanto, S.Kom., M.Si. (Universitas Krisnadwipayana, *Indonesia*)

ALAMAT PENERBIT

Universitas Krisnadwipayana

Jl. Kampus UNKRIS Jatiwaringin, Jakarta 13077

Gedung G (Fakultas Teknik) Lantai 2

Ruang Sekretariat Prodi Sistem Informasi

Telepon :.021-84998529

E-Mail : sisteminformasi@unkris.ac.id

PENGANTAR REDAKSI

Puji syukur kami panjatkan kehadiran Tuhan Yang Maha Esa dengan terbitnya Jurnal Information System Vol. 3 No. 1 Mei 2023 yang dikelola Program Studi Sistem Informasi, Fakultas Teknik, Universitas Krisnadwipayana.

Dengan hadirnya Journal Information System diharapkan dapat meningkatkan kontribusi dalam mengembangkan ilmu pengetahuan khususnya dalam bidang Sistem Informasi ini menyajikan berbagai macam topik pembahasan dalam lingkup Ilmu Komputer. Untuk kesempurnaan pada terbitan volume atau nomor berikutnya, redaksi sangat mengharapkan kritik dan saran dari semua pihak yang telah memberikaan apresiasi pada jurnal ilmiah Sistem Informasi ini.

Redaksi mengucapkan terima kasih setinggi-tingginya kepada semua pihak yang telah memberikan kontribusinya sehingga jurnal ilmiah sistem informasi ini dapat terbit. Dengan harapan artikel yang dimuat pada edisi ini dapat memberikan manfaat kepada para pembaca. Harapan redaksi berikutnya adalah mohon kesediaan dari pemerhati untuk dapat menyumbangkan tulisannya sebagai materi terbitan volume atau nomor berikutnya.

Jakarta, Mei 2023

Redaksi

DAFTAR ISI

Sampul Depan	i
Dewan Redaksi	ii
Pengantar Redaksi	iii
Daftar Isi	iv
I	Implementasi Algoritma Kriptografi RC4 Untuk Keamanan Database Aplikasi Voting Pemilihan Ketua Umum Berbasis WEB	1-9
	Fadlilah Septyana Yanuba, dkk.	
II	Penerapan Algoritma Backtracking Pada <i>N-Queen</i> Problem Permainan Catur	10-16
	Santi Rahmawati, dkk.	
III	Implementasi Fuzzy Logic Dengan Metode Mamdani Untuk Sistem Pendukung Keputusan Kinerja Dosen	17-27
	Junaidi	
IV	Penerapan Decision Tree Dengan Algoritma C4.5 Untuk Menentukan Rekomendasi Kenaikan Jabatan Karyawan	28-37
	Elmi Devia	
V	Rancang Bangun Aplikasi E-Commerce Penjualan Souvenir Berbasis Android	38-48
	Junaidi	

Implementasi Algoritma Kriptografi RC4 Untuk Keamanan Database Aplikasi Voting Pemilihan Ketua Umum Berbasis WEB

Fadlilah Septyana Yanuba¹, Muhlis Tahir², M. Khamdi Fadli³, Fathin Nisa Nafasa⁴, Siti Aminatus Zahrah⁵, Afifaturoh Rohmah⁶

^{1,2,3,4,5,6} Ilmu Pendidikan, Pendidikan Informatika, Universitas Trunojoyo Madura, Bangkalan, Indonesia

Email: ¹190631100102@student.trunojoyo.ac.id, ²muhlis.tahir@trunojoyo.ac.id, ³190631100106@student.trunojoyo.ac.id, ⁴190631100109@student.trunojoyo.ac.id, ⁵190631100121@student.trunojoyo.ac.id, ⁶190631100129@student.trunojoyo.ac.id

Abstrak– Rivest code 4 (RC4) adalah perhitungan kriptografi kunci simetris canggih yang memiliki mode kerja stream chipper, sehingga dalam menangani informasi dan data pada waktu tertentu menggunakan dua kotak pengganti (s-box) sebagai tampilan dengan panjang perubahan 256 dan selanjutnya s-box kedua yang merupakan elemen dari perhitungan public key. Rivest code 4 digunakan untuk menyandikan informasi, pesan atau data. Peneliti mencoba mengimplementasikan fitur login Sistem Pemilihan Ketua Umum yang menjadi pokok bahasan penelitian ini, dengan mengimplementasikan algoritma RC4 pada aplikasi yang sudah ada. Akibatnya, data terenkripsi yang tersimpan pada database akan sulit dicari tahu arti sebenarnya. Berdasarkan temuan penelitian ini, field yang berisi informasi pengguna yang mana dalam hal ini yaitu password dapat terenkripsi dengan baik, dan juga saat implementasi login juga dapat berjalan dengan baik dengan menggunakan algoritma RC4.

Kata Kunci: Keamanan; Algoritma Rivest Code 4; Database; Voting

Abstract– Rivest code 4 (RC4) is a sophisticated symmetric key cryptographic calculation that has a stream chipper working mode, so that in handling information and data at a certain time it uses two replacement boxes (s-boxes) as displays with a change length of 256 and then the second s-box which is an element of public key computation. rivest code 4 is used to encode information, messages or data. Researchers are trying to implement the Chairman Election System login feature which is the subject of this research, by implementing the RC4 algorithm in existing applications. as a result, encrypted data stored in databases will be difficult to find out its true meaning. based on the findings of this study, the field containing user information which in this case is the password can be encrypted properly, and also when the login implementation can also run well using the RC4 algorithm.

Keywords: Safety; Algorithm Rivest Code 4; Database; Voting

1. PENDAHULUAN

Begitu banyak kehidupan modern yang bergantung pada teknologi informasi, teknologi memainkan peran penting dalam pertumbuhan sumber daya manusia. Alhasil, manusia kini dapat dengan mudah menjalin komunikasi maupun berbagi informasi, bertukar data dengan jarak jauh berkat teknologi yang semakin maju saat ini. setiap aspek kehidupan modern telah dipengaruhi oleh kemajuan teknologi informasi, yang memudahkan orang untuk bertukar data dalam berbagai format di komputer. Dampak negative dari teknologi yang semakin meningkat yakni kejahatan pencurian data ataupun penyelahgunaan data informasi[1]. Sebagai hasilnya, melindungi data yang disimpan dari akses, modifikasi, dan penghapusan yang tidak sah sangat penting jika ingin tetap aman dari berbagai ancaman.

Terdapat banyak sekali pekerjaan yang dapat diselesaikan dengan cepat, akurat, dan efektif berkat kemajuan teknologi informasi. Keamanan basis data kemudian menjadi salah satu aspek terpenting dari teknologi informasi[2]. Integritas dan keamanan data merupakan pertimbangan penting. Pentingnya kerahasiaan dan keamanan data semakin meningkat. Keamanan data melibatkan banyak kasus yang sekarang menjadi pekerjaan dimana membutuhkan banyak keamanan dan uang untuk dikelola. Mekanisme keamanan yang baik diperlukan untuk mencegah informasi data diakses oleh pihak yang tidak berwenang[3].

Hacker yang suka mengubah, mengunci, menyadap, menghapus, dan memodifikasi isi data orang lain hanyalah salah satu contoh bagaimana teknologi modern dapat digunakan dalam sistem peradilan pidana. Kriptografi adalah disiplin terkenal di bidang keamanan data[4]. Steganografi dan kriptografi adalah dua dari banyak metode yang tersedia untuk melindungi data ini[5]. Kriptografi yang bertujuan untuk mengubah pesan (plaintext) menjadi pesan yang sulit dipahami (ciphertext), memiliki kelebihan dan kekurangan[6]. Namun, orang yang membaca data terenkripsi mungkin curiga terhadap kriptografi, yang dapat menyebabkan mereka merusak enkripsi.

Salah satu pemanfaatan teknologi yakni dengan dilakukannya pemilihan dengan menggunakan media digital[7].

Hal ini membantu dalam proses pemungutan suara yang penyelenggaraannya tidak dapat dilakukan secara langsung. Seperti yang kita ketahui, selama ini proses pemungutan suara untuk pemilihan ketua umum maupun pemilihan lainnya dilakukan dengan secara manual, dengan menggunakan media kertas dan dilakukan secara serentak disatu tempat yang telah ditentukan. Hal ini juga menjadi kendala pada seseorang yang tidak dapat datang dalam pemungutan suara setempat yang telah ditentukan.

Oleh karena itu, diterapkanlah sistem pengamanan data pada proses pemilihan ketua umum yang dilakukan di Prodi Pendidikan Informatika guna mengamankan data dan informasi. Hal ini bertujuan untuk memberikan keamanan pada database yang berfungsi dalam penyimpanan hasil perolehan suara yang telah dilakukan[8]. Algoritma kriptografi yang digunakan yakni algoritma kriptografi RC4[9].

2. METODOLOGI PENELITIAN

2.1 Analisa Masalah

Kemajuan teknologi yang sangat pesat terus mengalami perubahan dan pembaruan yang lebih baik. Salah satu contohnya yakni pada sistem pemilihan umum, baik dari sebuah organisasi kecil maupun besar, yang dimana sebelumnya dilakukan dengan cara pemilihan langsung yakni dengan melakukan pemungutan suara menggunakan kertas suara. Pemilihan ini mengalami perubahan yang dilakukan melalui media online. Maka dari itu diperlukan pengamanan yang sangat baik agar data hasil pemilihan bisa tersimpan dengan baik tanpa adanya kecurangan yang dapat merusak hasil pemungutan suara. Pada saat pemilih mau menggunakan hak suaranya kerap kali ditemukan kasus jika pemilih tersebut bahwasannya sudah memilih padahal yang sebenarnya belum melaksanakan hak suaranya.

Dengan adanya permasalahan tersebut, maka diperlukan suatu algoritma enkripsi untuk mengamankan akun dari pemilih yang mana nantinya algoritma ini akan dienkripsi pada hanya satu field saja yaitu password. sehingga pihak yang tidak bertanggung jawab atas keamanan database tidak dapat membaca ataupun mengetahui karena privasi data tersebut[10]. Algoritma yang digunakan yakni algoritma RC4.

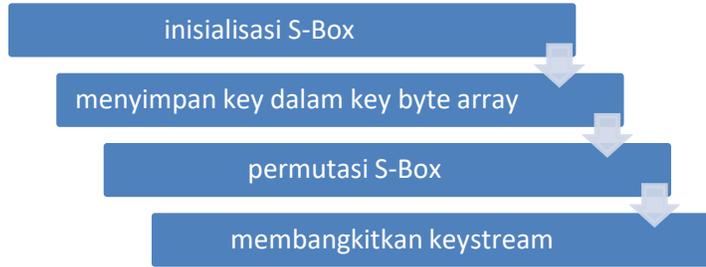
2.2 Tahapan Penelitian

Dalam penelitian ini, metode waterfall digunakan. Dimulai dengan analisis, desain, pengkodean, pengujian, dan dukungan, model waterfall menawarkan pendekatan aliran hidup perangkat lunak berurutan atau sekuensial[11]. Berikut adalah tahapan-tahapan yang dilakukan pada penelitian ini dengan menggunakan waterfall:

- a) Tahap analisis kebutuhan perangkat lunak
Pada tahap ini dimana semua masalah pengguna dipecah dan komponen sistem atau perangkat lunak, objek, hubungan antar objek, dan seterusnya diidentifikasi. Pada titik ini, penulis melakukan analisis kebutuhan yang diperlukan untuk menyusun program keamanan. Mulai dengan menganalisis kebutuhan program dengan menggunakan algoritma kriptografi RC4. Kemudian dilanjutkan ke tahap coding, dan yang terakhir dilihat setelah pengujian apakah sudah berhasil atau tidak.
- b) Perancangan
pada tahap ini merupakan proses multi langkah yang berfokus pada perancangan program perangkat lunak berdasarkan hasil data dari tahap analisis. Program perangkat lunak ini akan diimplementasikan dan mencakup diagram arsitektur perangkat lunak, kemajuan antar-representasi, dan desain basis data pembuatan kode program. Pada tahap ini, penulis menerapkan desain sistem, dimana sistem menangani pemilihan perangkat keras dan aplikasi persiapan perangkat lunak (coding atau pengkodean). Program yang didasarkan pada logika yang dirancang dalam desain diberi kode.
- c) Pengujian
Tahap pengujian menentukan apakah sistem atau perangkat lunak yang dikembangkan sesuai dengan kebutuhan pengguna dan menghilangkan atau meminimalkan cacat program (defect) sehingga sistem yang telah dikembangkan akan benar-benar membantu pengguna media dalam memperbaiki kesalahan pemrograman. Penulis menggunakan metode pengujian black box sebagai media pengujian.
- d) Pendukung (support) atau pemeliharaan (maintenance)
Pada tahapan ini dilakukan pemeliharaan pada produk dan membuat beberapa perubahan jika merasa ada sesuatu yang benar-benar perlu di kembangkan dalam produk tersebut.

2.3 Algoritma RC4

RC4 merupakan suatu algoritma yang dibuat pada tahun 1987 oleh Ron rivest di laboratorium RSA security[12]. Algoritma ini termasuk kedalam jenis stream chipper dan memiliki sifat kunci simetris. Algoritma ini melakukan proses enkripsi dengan cara byte per byte dengan operasi biner. Berikut ini rangkaian proses yang dimiliki RC 4 diantaranya :

**Gambar 1.** Alur RC4

Pada waktu proses enkripsi RC 4 membangkitkan key stream yang nantinya di XOR-kan dengan plaint text dan sebaliknya pada waktu dekripsi RC 4 keystream tersebut akan di XOR-kan dengan chipper text. RC 4 memproses data yang mana dalam satu waktu hanya berukuran byte (1 byte = 8bit). RC 4 menggunakan 2 kotak substitusi atau s box dengan array 256 byte [13]. Kotak substitusi pertama berisi permutasi dari bilangan 0 sampai 255 dan kotak substitusi kedua berisi permutasi fungsi dari kunci. Berikut ini penjelasan step by step dari algoritma RC4 [14] :

- 1) S-box di inisialisasi dengan persamaan seperti dibawah ini :

$$\text{for } m = 0 \text{ to } 255$$

$$S[m] = m$$
- 2) Kemudian alur inisialisasi S-box (Larik/array K) dengan persamaan dibawah ini:
 Panjang kunci "length" for $m = 0$ to 255

$$K[m] = \text{Kunci } [m \bmod \text{length}]$$
- 3) Setelah itu S-box diacak dengan menggunakan persamaan dibawah ini :

$$M = 0 ; n = 0$$

$$\text{for } m = 0 \text{ to } 255$$

$$n = (n + S[m] + K[n]) \bmod 256$$
 swap $S[m]$ dan $S[n]$
- 4) Terakhir buatlah pseudocode random dengan rumus persamaan dibawah ini :

$$m = (m + 1) \bmod 256$$

$$n = (n + S[m]) \bmod 256$$
 swap $S[m]$ dan $S[n]$

$$t = (S[m] + S[n]) \bmod 256$$

$$K = S[t]$$

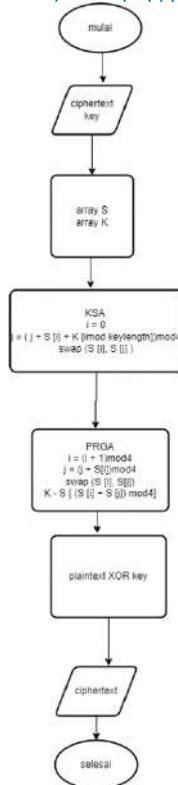
3. HASIL DAN PEMBAHASAN

3.1 Analisa Sistem

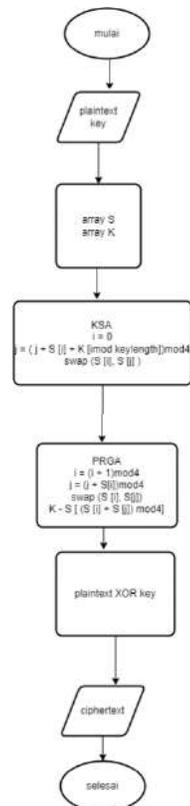
Analisa sistem merupakan sebuah proses dimana kita akan menyusun susunan dalam pembuatan proyek yang diusulkan. Dibawah ini merupakan flowchart dari proses enkripsi dan dekripsi

Flowchart Sistem

Alur proses enkripsi dan dekripsi pada database merupakan definisi dari fungsi sebuah flowchart yang dimana pada penelitian ini menggunakan RC4.



Gambar 2. Flowchart proses dekripsi RC4 pada database



Gambar 3. Flowchart proses enkripsi RC4 pada database

3.2 Implementasi dan Pengujian Aplikasi

Berikut ini kode program yang digunakan dalam keamanan data algoritma RC4 pada database aplikasi voting, antara lain:

a. Kode program inialisasi S-Box

```
// inialisasi sbox
public static function sbox()
{
    self::$state = array();
    for ($i=0; $i < 256; $i++) {
        self::$state[$i] = $i;
    }
    return self::$state;
}
```

b. Proses inialisasi S-Box (Array K)

```
//menyimpan kunci ke bentuk array
public static function menk($key)
{
    $keybyte[] = strlen($key);
    $n = strlen($key);
    $keyar = [];
    for ($a=0; $a < 256; $a++) {
        $keyar[$a]=ord($key[$a % $n]);
    }
    return $keyar;
}
```

c. Proses pengacakan S-Box (permutasi S-Box)

```
//permutasi nilai s box
public static function mtnsbox($keyq)
{
    $kunci = self::menk($keyq);
    $j = 0;
    for ($i=0; $i < 256; $i++) {
        $j = ($j+self::$state[$i]+$kunci[$i]) % 256;
        $temp = self::$state[$i];
        self::$state[$i] = self::$state[$j];
        self::$state[$j] = $temp;
    }
    return self::$state;
}
```

d. Proses pseudorandom code (keystream)

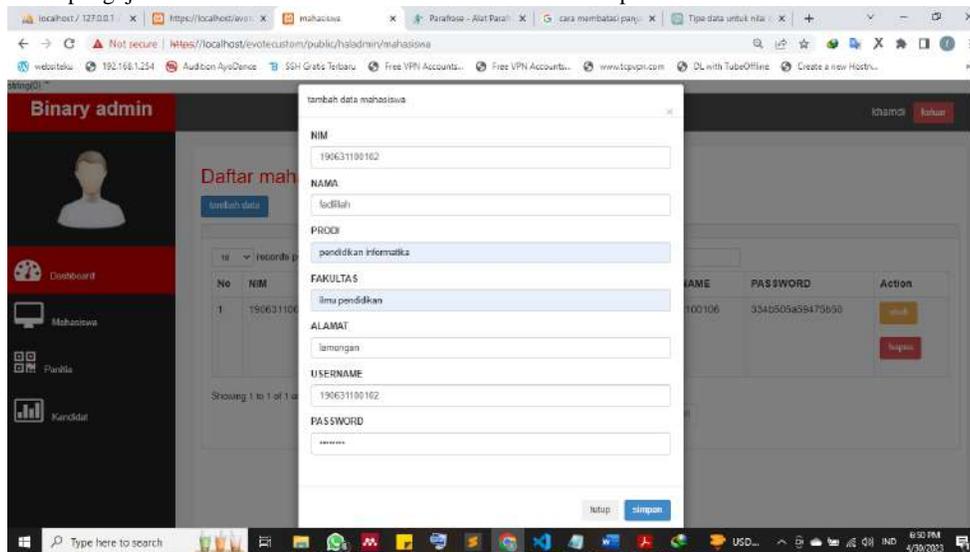
```
//membangkitkan pseudorandom (keystream)
public static function
psdrndm($keyqq,$kalimat)
```

```
{
    self::mtsnsbox($keyqq);
    $kstrm[] = strlen($keyqq);
    $strkstrm = "";
    $binkstrm[] = strlen($keyqq);

    $i = $j = 0;
    for ($c=0; $c < strlen($kalimat); $c++) {
        $i = ($i+1)%256;
        $j = ($j+self::$state[$i])%256;
        $temp = self::$state[$i];
        self::$state[$i] = self::$state[$j];
        self::$state[$j] = $temp;
        $t = (self::$state[$i]+self::$state[$j])%256;
        $strkstrm = $strkstrm . self::$state[$t];
        $kstrm[] = self::$state[$t]; //keystream
    }
    return $strkstrm;
}
```

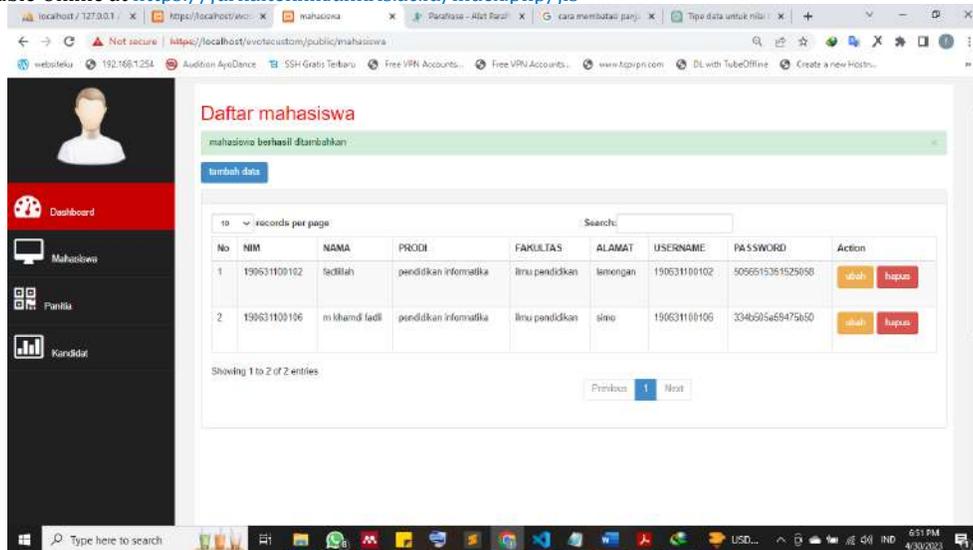
Berikut ini merupakan hasil pengujian dari aplikasi voting yang telah dibuat. Pengujian dilakukan melalui web browser dengan menjalankan alamat yang tersimpan di httdocs.

- a. Dalam pengujian ini dilakukan oleh admin untuk membuat akun pemilih



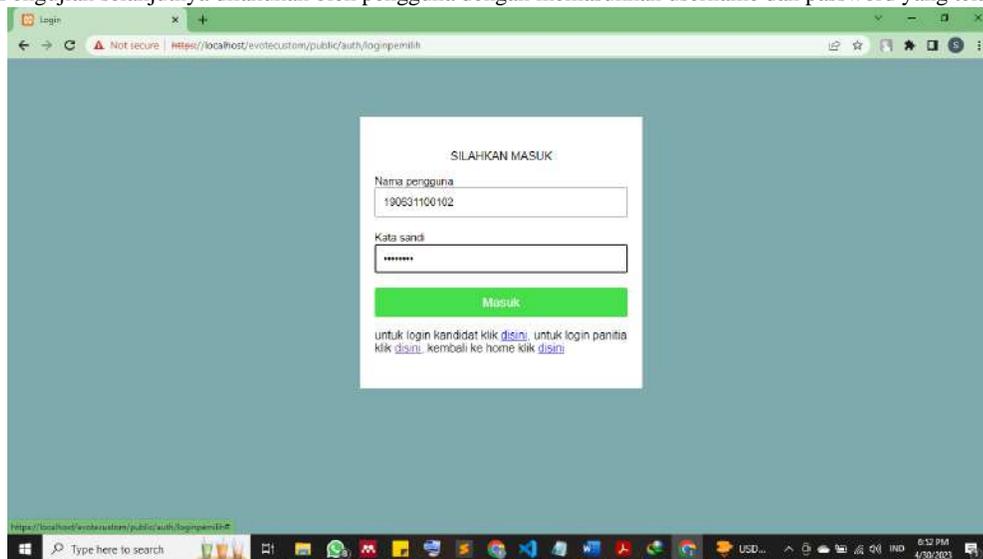
Gambar 4. Tampilan admin untuk membuat akun pemilih

Dan hasilnya adalah



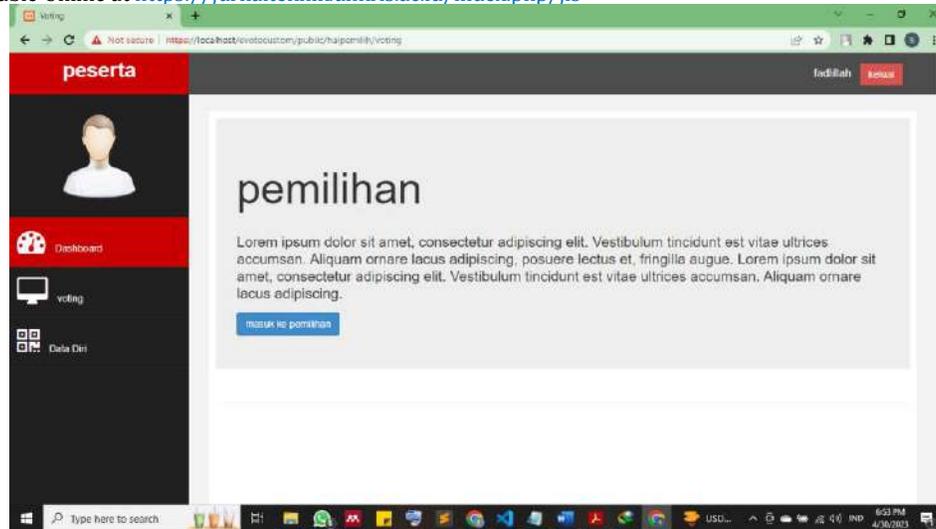
Gambar 5. Tampilan dashboard admin

b. Pengujian selanjutnya dilakukan oleh pengguna dengan memasukkan username dan password yang telah dibuat.



Gambar 6. Tampilan halaman login pemilih

Dan hasilnya



Gambar 7. Tampilan dashboard pemilih

4. KESIMPULAN

Berdasarkan hasil pengujian yang telah dilakukan, kesimpulan yang dapat ditarik berdasarkan penelitian dan analisis diatas adalah sebagai berikut :

1. Pada penelitian ini menghasilkan sebuah aplikasi *e-voting* dengan menggunakan keamanan database kriptografi RC4 yang dilakukan pada pemilihan ketua umum HIMAPIF.
2. Perancangan sistem menggunakan metode *waterfall*.
3. Akun yang dibuat oleh admin dapat dienkripsi dengan baik sehingga terjamin kerahasiaan dan keamanannya.
4. Ketika user akan masuk kedalam halaman pemilu user dapat memasukkan username dan password yang mana password tersebut sudah terenkripsi dan dapat masuk kedalam halaman pemilu.
5. Sistem sebelumnya hanya menyimpan data dalam database sesuai dengan apa yang dimasukkan ke dalam sistem informasi. Ini berarti siapa pun yang melihat isi database dapat membacanya, dan keamanannya tidak terjamin. Namun pada penelitian ini sudah tidak bisa lagi dilakukan hal yang sama dikarenakan keamanan database sudah ditingkatkan dengan menggunakan kriptografi RC4.

REFERENCES

- [1] W. P. Abdul Kodir, "IMPLEMENTASI KRIPTOGRAFI DENGAN MENGGUNAKAN METODE RC4 DAN BASE64 UNTUK MENGAMANKAN DATABASE SEKOLAH PADA SDN GROGOL UTARA 10," *Skatika*, vol. 4, no. 1, pp. 7–14, 2021, doi: 10.21856/j-pep.2021.4.08.
- [2] I. Afrianto and N. Taliasih, "Sistem Keamanan Basis Data Klien P.T. Infokes Menggunakan Kriptografi Kombinasi RC4 Dan Base64," *J. Nas. Teknol. dan Sist. Inf.*, vol. 6, no. 1, pp. 9–18, 2020, doi: 10.25077/teknosi.v6i1.2020.9-18.
- [3] A. Setiawan and T. Fatimah, "Implementasi Algoritma Kriptografi Rc4 Untuk Keamanan Database Aplikasi Penggajian Karyawan Berbasis Web Pada Pt. Trans Intra Asia," *Skatika*, vol. 4, no. 1, pp. 66–71, 2021, doi: 10.36080/skatika.v4i1.2044.
- [4] M. D. Wulandari *et al.*, "Aplikasi Pengamanan Database Berbasis Desktop dengan Algoritma AES-128 dan Rivest Code (RC4)," *Skatika*, vol. 1, no. 1, pp. 373–379, 2018.
- [5] Z. Basim, "Steganografi Dengan Algoritma Eof Untuk Keamanan Data Berbasis Desktop Pada Smk As- Su ' Udiyah," *Skatika*, vol. 3, no. 4, pp. 54–60, 2020.
- [6] R. Maulana and R. M. Simanjorang, "Implementasi Kriptografi Pengamanan Data Pribadi Siswa SMA Swasta Jaya Krama Beringin Dengan Algoritma RC4," *J. Nas. Komputasi dan Teknol. Inf.*, vol. 4, no. 6, pp. 377–383, 2021, doi: 10.32672/jnkti.v4i6.3533.
- [7] S. Jaleha and E. Suriyani, "Implementasi Sistem E-Voting Dilihat Aspek Komunikasi Dalam Rangka Pemilihan Kepala Desa Di Desa Kambitin Raya Kecamatan Tanjung Kabupaten Tabalong," *J. Adm. Publik dan Bisnis*, vol. 3, no. 2, pp. 1253–1264, 2020.



- [8] I. Anas, G. L. Ginting, E. Ndruru, A. S. Sembiring, and T. Zebua, "Perancangan Aplikasi Keamanan Data Dengan Kombinasi Algoritma Kriptografi RC4 dan One Time Pad," *J. Ris. Komput.*, vol. 8, no. 1, pp. 20–27, 2021, doi: 10.30865/jurikom.v7i5.2541.
- [9] J. S. Surbakti and S. Subandi, "Aplikasi Pengamanan Database Keuangan Berbasis Desktop Menggunakan Algoritma Rc4 Dan Vigenere Cipher," *Skatika*, vol. 1, no. 1, pp. 237–242, 2018, [Online]. Available: <https://jom.fti.budiluhur.ac.id/index.php/SKANIKA/article/view/187>
- [10] S. Susanto, "Implementasi Keamanan Data Menggunakan Algoritma Rivest Code 4 (RC4) Pada Sistem Informasi Inventory Stock Barang Pada Distributor PT.Wings Food," *Lontar Komput. J. Ilm. Teknol. Inf.*, vol. 8, no. 2, pp. 77–88, 2017, doi: 10.24843/lkjiti.2017.v08.i02.p02.
- [11] S. Murni, L. Latifah, R. Sabaruddin, and Y. L., "Penerapanan Metode Waterfall Dalam Pembangunan Aplikasi Akuntansi Kontraktor Dengan Pemrograman Php," *J. Teknol. Inf. MURA*, vol. 11, no. 1, pp. 55–67, 2019, doi: 10.32767/jti.v11i1.452.
- [12] G. Grehasen and S. Mulyati, "Pengamanan Database Pada Aplikasi Test Masuk Karyawan Baru Berbasis Web Menggunakan Algoritma Kriptografi AES-128 Dan RC4," *Budi Luhur Inf. Technol.*, vol. 14, no. 1, pp. 52–60, 2017, [Online]. Available: <https://journal.budiluhur.ac.id/index.php/bit/article/view/464%0A>
- [13] K. Fahmi, "Pengamanan Data Arsip Pada Balai Desa Sidodadi Menggunakan Kriptografi Modern RC4," *Resolusi Rekayasa Tek. Inform. dan Inf.*, vol. 2, no. 2, pp. 58–66, 2021, [Online]. Available: <http://www.djournals.com/resolusi/article/view/241>
- [14] Taronisokhi zebua, "Pengamanan citra digital berdasarkan Modifikasi algoritma rc4," *J. Teknol. Inf. dan Ilmu Komput. (JTIK)*, vol. 4, no. 4, pp. 275–282, 2017, doi: 10.25126/jtiik.201744474.

Penerapan Algoritma Backtracking Pada *N-Queen* Problem Permainan Catur

Santi Rahmawati¹, Rifda Triani Mutmainah², Raden Marissa Lestari³, Asti Herliana⁴

^{1,2,3,4}Fakultas Teknologi Informasi, Teknik Informatika, ARS University, Bandung, Indonesia

Email: ¹santirahmaaaaa@gmail.com, ²rifdatrianimutmainah21@gmail.com, ³rdmarissalestari@gmail.com, ⁴asti@ars.ac.id

Abstrak– Pada penulisan penelitian ini bertujuan untuk menerapkan Algoritma Backtracking dalam menyelesaikan masalah *N-Queen* pada permainan catur. *N-Queen* problem merupakan permasalahan yang mencari solusi penempatan bidak sebanyak N pada papan berukuran $N \times N$ dengan ilustrasi setiap bidak yang ditempatkan tidak boleh berada pada satu kolom, satu baris, dan satu diagonal yang sama sehingga tidak ada dua ratu yang saling menyerang dalam satu langkah Gerakan. Salah satu cara yang dapat menyelesaikan permasalahan *N-Queen* pada permainan catur ini yaitu dengan menggunakan Algoritma Backtracking. Dalam penerapan algoritma backtracking, solusi dicari dengan mencoba setiap kemungkinan langkah satu per satu dan kembali ke langkah sebelumnya jika langkah tersebut tidak menghasilkan solusi yang benar dan mencoba Langkah lain hingga solusi ditemukan atau semua kemungkinan telah dicoba. Penelitian ini menggunakan metode kepustakaan dimana metode ini melibatkan pencarian, peninjauan dari sumber-sumber tertulis yang berhubungan dengan topik yang sedang diteliti. Dengan membentuk pohon ruang status (State Space Tree) pada Algoritma Backtracking menghasilkan solusi pada $N = 4$ dengan jalur [2,4,1,3].

Kata Kunci: Algoritma Backtracking; Catur; *N-Queen*; State Space Tree

Abstract– In writing this study aims to apply the Backtracking Algorithm in solving the *N-Queen* problem in chess games. The *N-Queen* problem is a problem that seeks a solution to the placement of N pieces on a board measuring $N \times N$ with an illustration that each piece placed cannot be in the same column, row, and diagonal so that no two queens attack each other in one Movement Step. One way that can solve the *N-Queen* problem in this chess game is by using the Backtracking Algorithm. In the application of the backtracking algorithm, the solution is sought by trying each possible step one by one and returning to the previous step if the step does not produce the correct solution and trying another Step until the solution is found or all possibilities have been tried. This research uses the literature method where this method involves searching, reviewing written sources and other documentation related to the topic being studied. By forming a state space tree in Backtracking Algorithm, the solution at $N = 4$ is the solution path [2,4,1,3].

Keywords: Backtracking Algorithm; Chess; *N-Queen*; Pohon Ruang Status

1. PENDAHULUAN

Catur adalah permainan papan strategis untuk dua pemain. Papan catur berbentuk persegi dan dibagi menjadi 64 kotak terang dan gelap bergantian yang disusun dalam kisi berukuran 8×8 . Setiap pemain memiliki 16 buah dengan nama yang berbeda dan pergerakan yang unik. Buah catur tersebut terdiri dari Raja, Menteri/Ratu, Gajah, Benteng dan Bidak/Pion. Bidak yang paling penting adalah raja, karena tujuan setiap pemain adalah untuk mensakmat raja lawan untuk memastikan bahwa ia tidak dapat menghindari penangkapan. Jika dihitung secara matematis, untuk satu langkah saja (putih dan hitam menjalankan buah caturnya satu kali) akan menunjukkan banyak kemungkinan yang berbeda.

Permainan catur telah menjadi bagian tak terpisahkan dari sejarah dan budaya manusia selama berabad-abad. Keterampilan yang dibutuhkan untuk bermain catur bukan hanya kemampuan untuk menggerakkan buah catur di sekitar papan, tetapi juga kemampuan untuk merencanakan strategi, berpikir kritis, dan membuat keputusan yang tepat. Salah satu tantangan paling menarik dalam catur adalah menempatkan ratu di papan dengan cara yang tidak saling mengancam, yang disebut dengan *N-Queen* problem.

Salah satu upaya memenangkan permainan ini yaitu dengan cara Minimalkan kekalahan dengan menerima Informasi tentang posisi/kedudukan bidak catur yang aman dan tidak berbahaya. Satu permasalahan yang ada dan menarik untuk diteliti yaitu *N-Queen* problem. Masalah *N-Queen* terdiri dari menempatkan N - Queen pada papan catur $N \times N$ sehingga tidak ada ratu yang menyerang satu sama lain, yaitu tidak ada dua ratu yang berada di baris yang sama, di kolom yang sama, dan di diagonal yang sama.

Solusi untuk masalah ini adalah menemukan konfigurasi penempatan ratu yang memenuhi semua batasan ini. Banyak algoritma telah dikembangkan untuk memecahkan masalah *N-Queen*. Salah satu algoritma yang efektif memecahkan masalah ini adalah algoritma backtracking. Algoritma backtracking ini secara rekursif memilih langkah yang mengarah ke solusi. Jika langkah ini tidak mengarah pada solusi yang diinginkan, sistem kembali mencoba langkah alternatif. Dengan menggunakan algoritma backtracking, sistem dapat mencapai solusi optimal dalam menyelesaikan masalah *N-Queen* dengan lebih efisien dibandingkan dengan metode pencarian lainnya. Hal ini memungkinkan pemain catur untuk mempelajari strategi dan gerakan yang tepat dalam situasi permainan akhir tertentu.



Beberapa jurnal yang mengatakan bahwa permasalahan *N-Queen* dapat diselesaikan oleh algoritma backtracing seperti penelitian yang dilakukan oleh Halida Astatin "Pemanfaatan pohon Realisasi Algoritma backtracing untuk memecahkan *N-Queen* problem"[1]. Kemudian Fritno Purba "Penerapan Algoritma Ranut Balik (Backtracing) dalam *N-Queen* problem permainan catur[2].

Tujuan dari penelitian ini adalah menerapkan algoritma backtracing untuk menyelesaikan masalah *N-Queen* dalam permainan catur. Dalam penelitian ini, peneliti menjelaskan langkah-langkah implementasi algoritma backtracing dan menganalisis kinerja algoritma dalam menyelesaikan masalah *N-Queen* untuk ukuran papan tertentu.

2. TINJAUAN PUSTAKA

2.1 Catur

Catur merupakan salah satu permainan papan yang paling populer di dunia pada saat ini. Permainan catur merupakan sebuah permainan berbasis strategi abstrak yang muncul pada abad ke-6 Masehi. Permainan ini berasal dari India utara pada abad ke-6 Masehi sebagai Chaturaga dan semakin populer diseluruh India pada abad ke-7. Hingga pada tahun 1924 catur diakui sebagai permainan dunia oleh *Federation International des Echecs (FIDE)* atau Federasi Catur Internasional merupakan organisasi internasional yang bertanggung jawab atas mengatur dan mengembangkan catur di tingkat global dan telah memberikan legitiasi global bagi pemain catur sebagai olahraga intelektual dan kompetitif yang diakui secara Internasional. Catur adalah permainan papan strategis untuk dua pemain. Papan catur berbentuk persegi dan dibagi menjadi 64 kotak terang dan gelap bergantian yang disusun dalam kisi berukuran 8x8. Setiap pemain memiliki 16 buah dengan nama yang berbeda dan pergerakan yang unik. Buah catur tersebut terdiri dari Raja, Menteri/Ratu, Gajah, Benteng dan Bidak/Pion. Bidak yang paling penting adalah raja, karena tujuan setiap pemain adalah untuk mensakmat raja lawan untuk memastikan bahwa ia tidak dapat menghindari penangkapannya [3]

2.2 Algoritma Backtracking (Runut-Balik)

Algoritma Backtracking adalah algoritma yang berbasis pada DFS untuk mencari solusi persoalan secara lebih efisien, secara sistematis mencari solusi persoalan di antara semua kemungkinan solusi yang ada dengan metode runut-balik, tidak perlu memeriksa semua kemungkinan solusi. Hanya langkah yang mengarah pada solusi saja yang perlu dipertimbangkan. Akibatnya, waktu pencarian dapat dihemat. Ciri khas dalam Algoritma Backtracking yaitu adanya fungsi pemangkasan (pruning). Pencarian solusi pada masalah direpresentasikan dalam bentuk pohon solusi, proses pemangkasan akan dilakukan terhadap simpul-simpul yang tidak mengarah kepada solusi. Jika suatu simpul telah dipangkas, simpul-simpul yang menjadi anak dari simpul tersebut tidak akan diproses, karena memangkas sebuah simpul sama halnya membuang seluruh lintasan yang berada di bawah simpul tersebut. Algoritma backtracking banyak digunakan dalam membuat permainan komputer, salah satunya adalah catur [1].

2.3 *N-Queen*

N-Queen merupakan generalisasi dari masalah 4-Queen dan 8-Queen, ditempatkan pada papan catur $N \times N$ dengan masing-masing ratu tidak berada pada baris, kolom dan diagonal yang sama. Dalam permasalahan *N-Queen*, tujuan utamanya adalah menempatkan N ratu pada papan catur sehingga tidak ada dua ratu yang saling menyerang Ketika digerakkan. Ini berarti setiap baris, kolom maupun pada diagonal hanya boleh diisi oleh satu ratu. Jika solusi ditemukan, berarti semua ratu telah ditempatkan dengan aman dan tidak saling menyerang satu sama lain [4]

2.4 State Space Tree (Pohon Ruang Status)

State space tree adalah salah satu teknik dalam pemecahan masalah yang melibatkan pemodelan masalah dalam bentuk struktur pohon. Pada setiap simpul di pohon, terdapat sebuah keadaan atau state yang merepresentasikan suatu solusi dari masalah yang sedang dipecahkan, yaitu penempatan beberapa ratu pada papan catur. Setiap cabang dari simpul tersebut merepresentasikan sebuah langkah atau operasi yang dapat dilakukan untuk menghasilkan solusi baru, yaitu penempatan satu ratu pada baris berikutnya. Penerapan algoritma backtracking pada *N-Queen* problem dapat dilakukan dengan membangun state space tree yang merepresentasikan semua kemungkinan penempatan ratu pada papan catur [5].

3. METODE

3.1 Pendekatan Penelitian

Pendekatan penelitian yang digunakan dalam penelitian ini menggunakan metode kepustakaan (*library research*), penelitian ini dilakukan dengan cara mengumpulkan data dari berbagai sumber seperti jurnal, skripsi, dan dokumen lainnya yang berhubungan dengan topik yang sedang diteliti. Selain itu, peneliti juga menggunakan strategi Depth-First-Search (DFS) pada backtracking dalam proses pencarian solusi dan menerapkan pohon solusi (State Space Tree) dalam merepresentasikan langkah-langkah yang dapat diambil dalam pencarian solusi

3.2 Tahapan penelitian

Adapun tahapan dalam penelitian ini meliputi perumusan masalah, inisialiasi, membuat pohon ruang status, penerapan DFS dan penyelesaian solusi menggunakan Algoritma Backtracking. Tahapan tersebut dapat dijelaskan sebagai berikut:

a. Merumuskan masalah

Sebelum melakukan penelitian, yakni menentukan masalah yang akan diselesaikan. Pada penelitian ini masalah yang akan diselesaikan yaitu penyelesaian *N-Queen* pada catur dengan menggunakan Algoritma Backtracking.

b. Pohon ruang status (State Space Tree)

Pohon ruang status (State Space Tree) cara untuk menggambarkan status-status persoalan kedalam simpul pohon tersebut yang merupakan representasi visual dari semua kondisi, yang mungkin dalam permasalahan. Setiap simpul mewakili kondisi dan cabang-cabang pohon mewakili langkah-langkah yang mungkin untuk mencapai kondisi berikutnya dengan demikian melihat pohon ruang status dapat mengidentifikasi dan mengumpulkan semua solusi yang valid [5]

c. Penerapan Depth-First-Search (DFS)

Setelah membuat pohon ruang status, selanjutnya menerapkan Depth-First-Search (DFS) yang secara sistematis digunakan untuk pencarian solusi dengan menjelajahi pohon ruang status, melibatkan penjelajahan mendalam pada pohon ruang status dengan mengunjungi dari setiap simpul-simpul sebelum kembali ke simpul sebelumnya [6]

d. Algoritma Backtracking

Algoritma Backtracking menggunakan strategi DFS dalam proses pencarian solusi. Algoritma Backtracking menerapkan DFS untuk menjelajahi pohon ruang status, Algoritma Backtracking juga merupakan algoritma pencarian yang digunakan untuk menemukan kemungkinan solusi. Kemungkinan solusi yang ada perlu diuji secara bertahap. Algoritma ini bekerja dengan mencoba setiap langkah secara bertahap dan mundur (*Backtrack*) ketika langkah tersebut tidak menghasilkan solusi yang valid [2]

Dengan kata lain, jika lintasan pada suatu simpul yang dibuat hasilnya melanggar aturan batasan, maka simpul tersebut akan dihentikan dan melakukan *Backtrack* untuk kembali pada simpul sebelumnya, lalu diteruskan dengan mencoba simpul anak lainnya [5]

4. HASIL DAN PEMBAHASAN

4.1 Pengujian Algoritma

Algoritma Backtracking memiliki property umum, salah satunya *Bounding Function* (Fungsi Pembatas). *Bounding Function* adalah kondisi yang bernilai benar. Kondisi dapat dikatakan benar jika ratu (*Queen*) tidak pada baris, kolom dan diagonal yang sama. Fungsi pembatas ini dinyatakan sebagai $B(x_1, x_2, x_3, \dots, x_n)$ akan bernilai true jika $(x_1, x_2, x_3, \dots, x_n)$ mengarah pada solusi dan tidak melanggar batasan.

Dalam pencarian solusi, perlu dilakukan pembangkitan simpul-simpul status pada pohon sehingga menghasilkan lintasan dari akar ke daun. Dalam tahap ini, menggunakan strategi *Depth-First-Search* (DFS) untuk membangkitkan simpul. Simpul-simpul yang dibangkitkan disebut simpul hidup. Jika simpul hidup yang diperluas maka lintasan akan semakin Panjang dan jika lintasan tidak mengarah pada solusi dengan artian melanggar fungsi pembatas, maka simpul tersebut akan dimatikan yang disebut simpul mati. Dengan kondisi seperti ini maka akan dijalankan Algoritma Backtracking dengan syarat jika pembentukan lintasan berhenti karena adanya simpul mati maka, akan dilakukan pemunduran (*Backtrack*) untuk Kembali pada simpul sebelumnya (orang tua), lalu diteruskan dengan membangkitkan simpul anak lainnya. Pencarian dihentikan jika sudah mencapai simpul tujuan.

Algoritma Backtracking ini akan direpresentasikan dalam bentuk pohon ruang status yang mengandung semua kemungkinan solusi dari permasalahan 4-Queen dengan papan catur 4x4. Menempatkan 4 ratu (*Queen*) ditempat yang berbeda dengan tidak berada pada baris, kolom dan diagonal yang sama sehingga tidak ada ratu yang saling menyerang. Rincinya pada pohon ruang status (State Space Tree) tiap simpul menyatakan status dari persoalan, sedangkan sisi-sisi dilabeli nilai vektor solusi (x_1, x_2, \dots, x_n).

4.2 Implementasi Algoritma

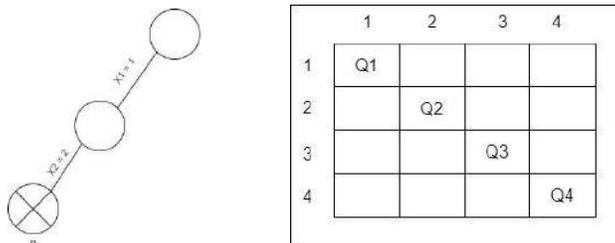
Penerapan tahapan-tahapan dalam penyelesaian permasalahan *N-Queen* akan dilakukan dimana peneliti akan menggunakan papan catur berukuran 4x4 dan $N=4$ yang berarti menempatkan 4 buah ratu pada tempat yang berbeda dimulai dengan membuat pohon ruang status dan gambaran papan dari hasil kemungkinan pada pohon ruang status tersebut sesuai fungsi pembatas.

Langkah pertama dalam membuat pohon status, peneliti akan terlebih dahulu menempatkan $Q_1, Q_2, Q_3, Q_4 =$ Queen dikolom 1,2,3,4 sebagai inialisasi awal dengan gambaran sebagai berikut

	1	2	3	4
1	Q1			
2		Q2		
3			Q3	
4				Q4

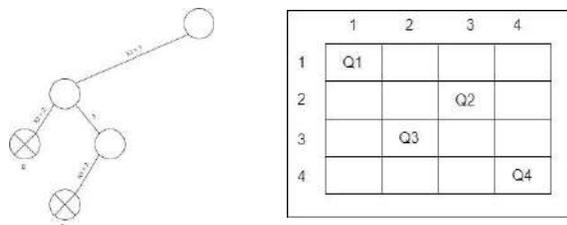
Gambar 1. Inialisasi awal penempatan ratu

Gambar diatas menunjukkan bahwa Q_1 menempati baris dan kolom ke 1, Q_2 menempati baris dan kolom 2, Q_3 menempati baris dan kolom 3, dan Q_4 menempati baris dan kolom 4. Selanjutnya, mulai percobaan memindahkan ratu satu ke kolom yang lain dengan pohon ruang status dan gambaran pada papan catur sebagai berikut



Gambar 2. Percobaan pertama pada pohon ruang status dan hasil pada papan catur 4 x 4

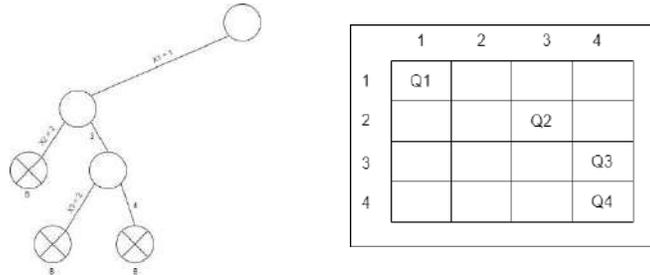
Gambar 2 menunjukkan bahwa tidak masalah menempatkan $x_1 = 1$ (Q_1 pada kolom dan baris 1), tetapi bermasalah jika menempatkan Q_2 pada kolom dan baris kedua, karena ia menjadi sejajar diagonal dengan Q_1, Q_3, Q_4 , maka hal ini melanggar fungsi batasan bahwa antara ratu tidak boleh berada pada diagonal yang sama sehingga simpul $x_2 = 2$ akan dihentikan. Selanjutnya mencoba untuk membuat langkah berikutnya



Gambar 3. Percobaan kedua pada pohon ruang status dan hasil pada papan catur 4 x 4

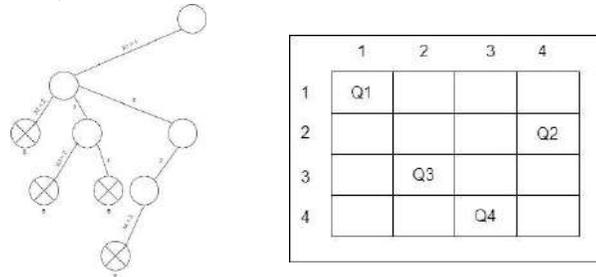
Gambar 3 menunjukkan perpindahan Q2 ke kolom 3 baris 2 maka dibuatlah simpul baru $x_2 = 3$ dan Q3 berpindah ke kolom 2 baris 3. Maka, lintasan akan dibuat di sisi kanan karena simpul kiri sudah dihentikan. Namun, Q3 melanggar fungsi batas karena sejajar diagonal dengan Q2 dengan itu simpul $x_3=2$ dihentikan dan mundur (*Backtrack*) pada simpul sebelumnya. Disinilah algoritma Backtracking dan DFS berjalan.

Percobaan masih tetap berlanjut, dengan memindahkan Q3 dan ditempatkan pada kolom 4 baris 3 dengan gambaran sebagai berikut



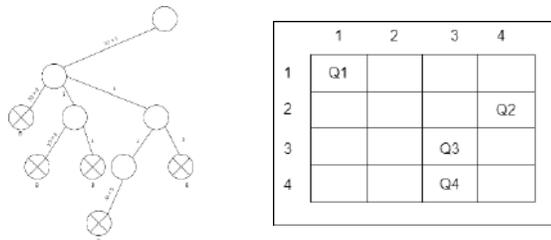
Gambar 4. Percobaan ketiga pada pohon ruang status dan hasil pada papan catur 4 x 4

Gambar 4 menunjukkan perpindahan Q3 di kolom 4 baris 3, namun percobaan ini melanggar fungsi Batasan karena menjadi sejajar diagonal dengan Q2 dan berada sejajar dengan kolom Q4. Maka simpul ini dihentikan dan kembali pada simpul sebelumnya.



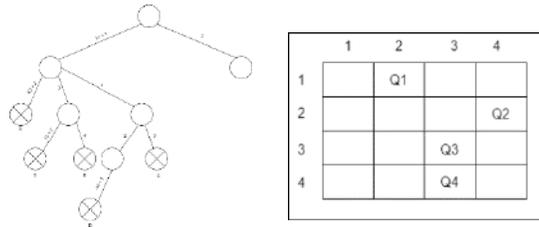
Gambar 5. Percobaan keempat pada pohon ruang status dan hasil pada papan catur 4 x 4

Gambar 5. Menunjukkan perpindahan Q2 ke dalam kolom 4 baris 2, Q3 ke dalam kolom 2 baris 3 dan Q4 ke dalam kolom 3 baris 4. Terlihat pada gambar papan catur bahwa Q4 melanggar fungsi batas dimana Q4 menjadi sejajar diagonal dengan Q3, maka simpul dihentikan dan kembali pada simpul sebelumnya yaitu simpul $x_3 = 2$. DFS disini berperan menjelajahi sejauh mungkin sebelum kembali ke simpul sebelumnya. Hal ini dapat menghasilkan pencarian yang lebih dalam pada stuktur pohon ruang status.



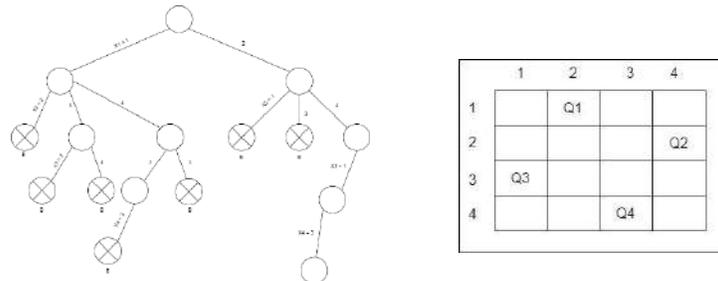
Gambar 6. Percobaan kelima pada pohon ruang status dan hasil pada papan catur 4 x 4

Pada percobaan kelima memindahkan Q3 ke dalam kolom dan baris 3, hasil percobaan tersebut melanggar fungsi batasan yaitu Q3 sejajar dengan kolom Q4 sehingga simpul $x_3=3$ dihentikan. Jika terus menerus hanya memindahkan Q2,Q3 dan Q4 maka solusi tidak akan ditemukan, dengan itu peneliti akan mencoba memindahkan Q1 dengan gambaran sebagai berikut



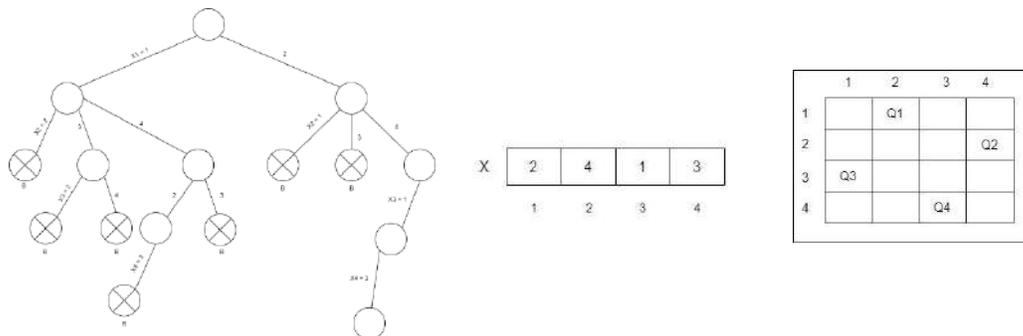
Gambar 7. Percobaan keenam pada pohon ruang status dan hasil pada papan catur 4 x 4

Pada percobaan keenam menunjukkan bahwa Q3 sejajar diagonal dengan Q2 dan berada pada kolom yang sama dengan Q4. Jika Q3 dipindahkan kedalam kolom 4 akan sejajar dengan Q2, maka Q3 dipindahkan dan menempati kolom 1.



Gambar 8. Percobaan ketujuh pada pohon ruang status dan hasil pada papan catur 4 x 4

Pada percobaan ketujuh terlihat bahwa Q3 tidak berada pada diagonal yang sama dengan Q1 dan Q4, serta tidak berada pada baris, kolom yang sama dengan Q1, Q2 dan Q4. Selanjutnya, jika melakukan percobaan pemindahan pada Q4 ditempatkan pada kolom 1 maka akan berada pada kolom yang sama dengan Q3, jika dipindahkan kedalam kolom 2 maka akan berada pada kolom yang sama dengan Q1 dan jika dipindahkan kedalam kolom 4, maka akan berada sejajar pada kolom Q2, demikian Q4 akan ditempatkan pada kolom 3 dimana Q4 tidak berada pada baris, kolom, diagonal yang sama dengan Q1, Q2 dan Q3.



Gambar 9. Hasil akhir percobaan ketujuh pada pohon ruang status dan hasil pada papan 4 x 4

Hasil dari representasi pohon ruang status, penggunaan DFS dan Algoritma Backtracking pada permasalahan *N-Queen* pada papan catur berukuran 4 x 4 menghasilkan solusi dengan tujuh percobaan pada $N = 4$ berupa jalur solusi [2,4,1,3] dimana Q1 ditempatkan pada kolom 2, Q2 ditempatkan pada kolom 4, Q3 ditempatkan pada kolom 1 dan Q4 ditempatkan pada kolom 3. Dengan ini tiap ratu tidak akan saling menyerang dalam satu langkah gerakan dikarenakan tiap ratu tidak berada pada baris, kolom maupun diagonal yang sama dan jika langkah-langkah baru dilakukan maka akan menghasilkan solusi baru dengan jalur solusi [3,1,4,2].

4.2 Implementasi Aplikasi

Pada penelitian ini, untuk menyelesaikan permasalahan *N-Queen*. Selain melalui langkah-langkah diatas, peneliti juga menggunakan program C++ dalam penyelesaiannya dengan kode utama dan output sebagai berikut



```
64 int main() {
65     int n = 4; // mengambil papan ukuran 4*4 dan 4 ratu
66     Solution obj;
67     vector < vector < string >> ans = obj.solveNQueens(n);
68     for (int i = 0; i < ans.size(); i++) {
69         cout << "Arrangement " << i + 1 << "\n";
70         for (int j = 0; j < ans[0].size(); j++) {
71             cout << ans[i][j];
72             cout << endl;
73         }
74         cout << endl;
75     }
76     return 0;
77 }
```

Arrangement 1
..Q.
Q..
...Q
.Q..

Arrangement 2
.Q..
...Q
Q..
..Q.

Gambar 9. Implementasi dan hasil program C++ dalam penyelesaian *N-Queen*

5. KESIMPULAN

Penggunaan Algoritma Backtracking pada permasalahan ini mampu menyelesaikan permasalahan pada *N-Queen* dengan menempatkan 4 ratu pada papan catur yang relatif kecil, karena untuk ukuran catur dan jumlah ratu yang lebih banyak memerlukan waktu yang cukup lama untuk mencapai pada solusi dengan kata lain implementasi saat ini belum sepenuhnya optimal untuk menangani kasus dengan nilai n yang lebih besar. Oleh karena itu, penelitian selanjutnya dapat mengatasi keterbatasan ini dengan mengusulkan strategi baru untuk meningkatkan efisiensi algoritma, serta mempertimbangkan pendekatan yang mampu menemukan semua solusi yang valid. Selain itu, penelitian mendatang dapat melibatkan analisis lebih mendalam tentang keterkaitan antar jumlah ratu dan ukuran papan catur untuk memberikan wawasan yang lebih komprehensif. Dengan memperbaiki keterbatasan-keterbatasan ini, penelitian di masa depan diharapkan dapat memberikan kontribusi yang lebih dalam pemecahan masalah *N-Queen* menggunakan *Algoritma Backtracking*.

REFERENCES

- [1] H. Astatin, "Pemanfaatan Pohon dalam Realisasi Algoritma Backtracking untuk Memecahkan N-Queens Problem."
- [2] F. Purba, "PENERAPAN ALGORITMA RANUT BALIK (BACKTRACKING) DALAM N-QUEEN PROBLEM PERMAINAN CATUR," *Jurnal Pelita Informatika*, vol. 6, no. 1, 2017, [Online]. Available: <http://www.kyma26.co.cc/2009/01/sejarahcatur>
- [3] R. E. Jeremiah, W. Sukmo Wardhono, and H. Muslimah Az-Zahra, "Analisis Pengalaman Interaksi Pengguna Terhadap Permainan Catur Sebagai Obyek Augmented Reality Menggunakan Game Experience Questionaire," 2019. [Online]. Available: <http://j-ptiik.ub.ac.id>
- [4] A. Adrifina, S. Wati, and U. Gunadarma, "Seminar Ilmiah Nasional Komputer dan Sistem Intelijen (KOMMIT 2008) Auditorium Universitas Gunadarma", [Online]. Available: <http://www.stttelkom.ac.id/staf/ZKA/>
- [5] M. Sheva and A. Sofjan, "Pemodelan Algoritme Backtracking dalam Menyelesaikan Permainan 2048 Deterministik." [Online]. Available: <http://informatika.stei.itb.ac.id/~rinaldi.munir/Matdis/2020->
- [6] M. Khoirussolih *et al.*, "Penyelesaian Masalah 8-Queen Dengan Depth First Search Menggunakan Algoritma Backtracking," vol. 4, no. 1, 2015, [Online]. Available: <http://socs.binus.ac.id/2013/04/23/uninformed->

Implementasi Fuzzy Logic Dengan Metode Mamdani Untuk Sistem Pendukung Keputusan Kinerja Dosen

Junaidi

Fakultas Teknik, Sistem Informasi, Universitas Krisnadwipayana, Jakarta, Indonesia

Email: junaidi@unkris.ac.id

Abstrak- Penilaian kinerja dosen pada perguruan tinggi sangat diperlukan untuk melihat bagaimana tingkat kinerja dosen dalam menjalankan semua kegiatan tridharma perguruan tinggi yang ada pada instansi dosen tersebut bekerja. Tujuan penelitian ini dimana ingin mengetahui bagaimana pelaksanaan penilaian kinerja dosen yang dilakukan dalam pelaksanaan kegiatan tridharma perguruan tinggi dan menentukan kinerja dosen yang terbaik dengan menggunakan *Fuzzy Logic* metode Mamdani. Masalah yang diselesaikan adalah cara menentukan tingkat kinerja dosen dalam melaksanakan tridharma perguruan tinggi dengan menggunakan tiga variabel input, yaitu pengajaran, penelitian dan pengmas. Langkah pertama penyelesaian masalah penentuan kinerja dosen dengan menggunakan metode *fuzzy logic* mamdani yaitu menentukan variabel input dan output yang merupakan himpunan tegas. Langkah kedua yaitu mengubah variabel input menjadi himpunan fuzzy dengan proses fuzzifikasi, selanjutnya langkah ketiga adalah pengolahan data himpunan fuzzy. Dan langkah terakhir adalah mengubah output menjadi himpunan tegas dengan proses defuzzifikasi dengan menggunakan perangkat lunak *MATLAB Online*, sehingga akan diperoleh hasil yang diinginkan pada variable output. Hasil dari perhitungan dengan menggunakan metode *fuzzy logic* mamdani, tingkat kinerja dosen untuk nilai variabel pengajaran 30 (rendah), nilai variabel penelitian 85 (baik) dan nilai variabel pengmas 50 (cukup) adalah 16, 4553 (cukup).

Kata Kunci: Kinerja Dosen; *Fuzzy Logic*; Metode Mamdani.

Abstract- Lecturer performance evaluation at tertiary institutions is needed to see how the level of performance of lecturers is in carrying out all the tridharma activities of higher education in the lecturer's institution. The purpose of this research is to find out how the performance assessment of lecturers is carried out in the implementation of higher education tridharma activities and to determine the best lecturer performance using the Mamdani Fuzzy Logic method. The problem to be solved is how to determine the performance level of lecturers in carrying out the tridharma of higher education by using three input variables, namely teaching, research and community service. The first step in solving the problem of determining lecturer performance using the Mamdani fuzzy logic method is to determine the input and output variables which are strict sets. The second step is to change the input variables into fuzzy sets with the fuzzification process, then the third step is processing the fuzzy set data. And the final step is to convert the output into a firm set by defuzzification process using the *MATLAB Online* software, so that the desired results will be obtained on the output variables. The results of calculations using the mamdani fuzzy logic method, the lecturer's performance level for the teaching variable value is 30 (low), the research variable value is 85 (good) and the pemmas variable value is 50 (adequate) is 16, 4553 (enough).

Keywords: Lecturer Performance; Fuzzy Logic; Mamdani method.

1. PENDAHULUAN

Sebagai seorang profesional, kinerja dosen juga perlu dinilai. Penilaian kinerja adalah penentuan secara periodik efektivitas operasional suatu organisasi, dan personalnya, berdasarkan sasaran strategi, standar, dan kriteria yang telah ditetapkan sebelumnya [1]. Pada prinsipnya penilaian kinerja merupakan cara pengukuran kontribusi-kontribusi dari individu dalam instansi yang dilakukan terhadap organisasi. Nilai penting dari penilaian kinerja adalah menyangkut penentuan tingkat kontribusi individu atau kinerja yang diekspresikan dalam menyelesaikan tugas-tugas yang menjadi tanggung jawab. Penilaian kinerja dosen pada perguruan tinggi sangat diperlukan untuk melihat bagaimana tingkat kinerja dosen dalam menjalankan semua kegiatan tridharma perguruan tinggi yang ada pada instansi dosen tersebut bekerja. Tujuan penelitian ini dimana ingin mengetahui bagaimana pelaksanaan penilaian kinerja dosen yang dilakukan dalam pelaksanaan kegiatan tridharma perguruan tinggi dan menentukan kinerja dosen yang terbaik dengan menggunakan *Fuzzy Logic* metode Mamdani.

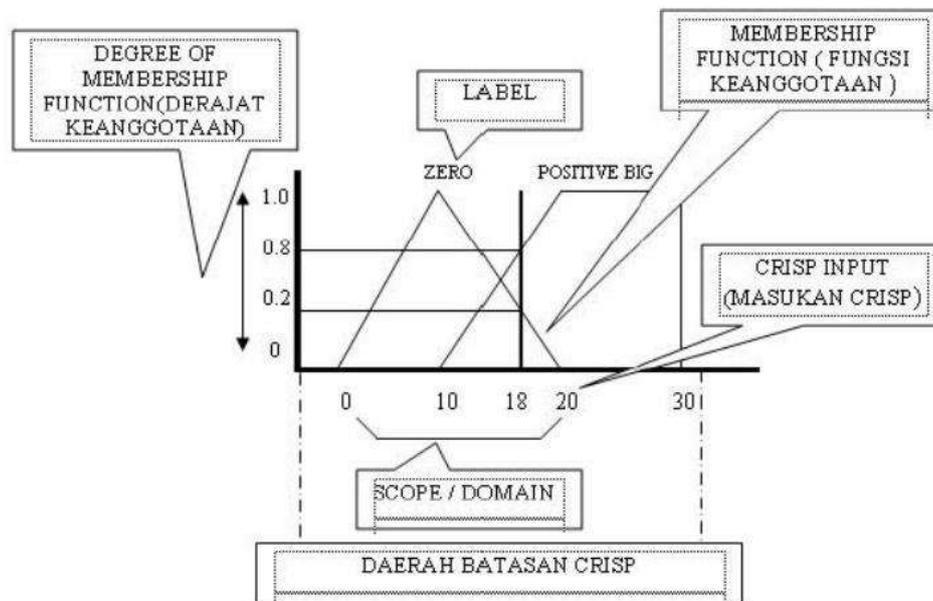
Fuzzy logic theory (teori logika fuzzy) adalah suatu cara yang tepat untuk memetakan suatu ruang input kedalam suatu ruang output. Titik awal dari konsep modern mengenai ketidakpastian adalah *paper* yang dibuat oleh Lofti A. Zadeh, dimana Zadeh memperkenalkan teori yang memiliki obyek-obyek dari himpunan *fuzzy* yang memiliki batasan yang tidak presisi dan keanggotaan dalam himpunan *fuzzy*, dan bukan dalam bentuk logika benar (*true*) atau salah (*false*), tapi dinyatakan dalam derajat (*degree*). Konsep seperti ini disebut dengan *Fuzziness* dan teorinya dinamakan *Fuzzy Set Theory*. *Fuzziness* dapat didefinisikan sebagai logika kabur berkenaan dengan semantik dari suatu kejadian, fenomena atau pernyataan itu sendiri. Seringkali ditemui dalam pernyataan yang dibuat oleh seseorang, evaluasi dan suatu pengambilan keputusan.

Sistem Pendukung Keputusan (SPK) merupakan salah satu produk perangkat lunak yang dikembangkan secara khusus untuk membantu dalam proses pengambilan keputusan. Sesuai dengan namanya, tujuan dari dipergunakannya sistem ini adalah sebagai “*second opinion*” atau “*information sources*” yang dapat dipakai sebagai bahan pertimbangan sebelum memutuskan kebijakan tertentu. Sistem pendukung keputusan merupakan suatu pendekatan untuk mendukung pengambilan keputusan [2]. Sistem pendukung keputusan menggunakan data, memberikan antarmuka pengguna yang mudah, dan dapat menggabungkan pemikiran pengambilan keputusan.

Fuzzy logic adalah cabang dari matematika dengan bantuan komputer memodelkan dunia nyata seperti yang dilakukan manusia. *Fuzzy logic* memformulasikan masalah menjadi lebih mudah, mempunyai pesisi yang tinggi, dan solusi yang akurat. *Fuzzy logic* menggunakan dasar pendekatan hukum-hukum untuk mengontrol sistem dengan bantuan model matematika.

Pada *Boolean Logic* setiap pernyataan benar atau salah, sesuai contoh pernyataan dengan 1 atau 0. Jelasnya himpunan *fuzzy* memiliki fleksibilitas keanggotaan yang diperlukan untuk keanggotaan pada suatu himpunan. Setiap kejadian dari tingkat dan alasan yang jelas adalah menunjukkan kasus terbatas pada pendekatan yang benar. Karena itu dapat disimpulkan bahwa *Boolean Logic* adalah subset dari *Fuzzy Logic*.

Profesor Lotfi A. Zadeh adalah guru besar pada *University of California* yang merupakan pencetus sekaligus yang memasarkan ide tentang cara mekanisme pengolahan atau manajemen ketidakpastian yang kemudian dikenal dengan logika *fuzzy*. Dalam penyajiannya variabel-variabel yang akan digunakan harus cukup menggambarkan ke-*fuzzy*-an tetapi di lain pihak persamaan-persamaan yang dihasilkan dari variabel-variabel itu haruslah cukup sederhana sehingga komputasinya menjadi cukup mudah. Karena itu Profesor Lotfi A Zadeh kemudian memperoleh ide untuk menyajikannya dengan menentukan “derajat keanggotaan” (*membership function*) dari masing-masing variabelnya. Fungsi keanggotaan (*membership function*), adalah suatu kurva yang menunjukkan pemetaan titik input data kedalam nilai keanggotaanya (sering juga disebut dengan derajat keanggotaan) yang memiliki interval antara 0 sampai 1. Konsep dasar logika *fuzzy* dapat dilihat pada gambar berikut ini [3]:



Gambar 1. Konsep Dasar Logika *Fuzzy*

- **Derajat Keanggotaan (*membership function*)** adalah : derajat dimana nilai *crisp* dengan fungsi keanggotaan (dari 0 sampai 1), juga mengacu sebagai tingkat keanggotaan, nilai kebenaran, atau masukan *fuzzy*.
- **Label** adalah nama deskriptif yang digunakan untuk mengidentifikasi sebuah fungsi keanggotaan.
- **Fungsi Keanggotaan** adalah mendefinisikan *fuzzy set* dengan memetakan masukan *crisp* dari domainnya ke derajat keanggotaan.
- **Masukan *Crisp*** adalah masukan yang tegas dan tertentu.
- **Lingkup/*Domain*** adalah lebar fungsi keanggotaan. Jangkauan konsep, biasanya bilangan, tempat dimana fungsi keanggotaan dipetakan.
- **Daerah Batasan *Crisp*** adalah jangkauan seluruh nilai yang dapat diaplikasikan pada variabel sistem.



Logika fuzzy dapat didefinisikan sebagai suatu jenis *logic* yang bernilai ganda dan berhubungan dengan ketidakpastian dan kebenaran parsial [4].

Ada beberapa alasan mengapa orang menggunakan *Fuzzy Logic*, antara lain [5]:

1. Konsep *Fuzzy Logic* mudah dimengerti. Konsep matematis yang mendasari penalaran *fuzzy* sangat sederhana dan mudah dimengerti.
2. *Fuzzy Logic* sangat fleksibel.
3. *Fuzzy Logic* memiliki toleransi terhadap data-data yang tidak tepat.
4. *Fuzzy Logic* mampu memodelkan fungsi-fungsi non linear yang sangat kompleks.
5. *Fuzzy Logic* dapat membangun dan mengaplikasikan pengalaman-pengalaman para pakar secara langsung tanpa harus melalui proses pelatihan.
6. *Fuzzy Logic* dapat bekerjasama dengan teknik-teknik kendali secara konvensional.
7. *Fuzzy Logic* didasarkan pada bahasan alami. Jika himpunan tegas (*crisp*), nilai keanggotaan suatu item x dalam suatu himpunan A , yang sering ditulis dengan $\mu_A(x)$, memiliki dua kemungkinan, yaitu:
 - a. Satu (1), yang berarti bahwa suatu item menjadi anggota dalam suatu himpunan.
 - b. Nol (0), yang berarti bahwa suatu item tidak menjadi anggota dalam suatu himpunan.

Himpunan adalah suatu kumpulan atau koleksi objek-objek yang mempunyai kesamaan sifat tertentu. Himpunan *fuzzy* merupakan suatu pengembangan lebih lanjut tentang konsep himpunan dalam matematika. Himpunan *fuzzy* adalah rentang nilai-nilai, masing-masing nilai mempunyai derajat keanggotaan antara 0 sampai dengan 1. Suatu himpunan *fuzzy* \tilde{A} dalam semesta pembicaraan U dinyatakan dengan fungsi keanggotaan $\mu_{\tilde{A}}$, yang nilainya berada dalam *interval* $[0,1]$, dapat dinyatakan dengan rumus (1):

$$\mu_{\tilde{A}} : U \rightarrow [0,1] \dots \dots \dots (1)$$

Himpunan *fuzzy* \tilde{A} dalam semesta pembicaraan U , biasa dinyatakan sebagai sekumpulan pasangan elemen u (u anggota U) dan derajat keanggotaannya dinyatakan sebagai rumus (2).

$$\tilde{A} = \{ (u, \mu_{\tilde{A}}(u) \mid u \in U \} \dots \dots \dots (2)$$

Ada beberapa hal yang perlu diketahui dalam memahami himpunan *fuzzy*, yaitu:

1. Variabel *fuzzy*

Variabel *fuzzy* merupakan suatu lambang atau kata yang menunjuk kepada suatu yang tidak tertentu dalam sistem *fuzzy*.

2. Himpunan *fuzzy*

Himpunan *fuzzy* merupakan suatu kumpulan yang mewakili suatu kondisi atau keadaan tertentu dalam suatu variabel *fuzzy*. Himpunan *fuzzy* memiliki dua atribut, yaitu :

- Linguistik, yaitu penamaan suatu grup yang memiliki suatu keadaan atau kondisi tertentu dengan menggunakan bahasa, seperti : Muda, Dewasa, Tua.
- Numeris, yaitu suatu nilai (angka) yang menunjukkan ukuran dari suatu variabel seperti : 3, 6, 9, dan sebagainya.

3. Semesta pembicaraan

Semesta pembicaraan adalah keseluruhan nilai yang diperbolehkan untuk dioperasikan dalam suatu variabel *fuzzy*.

4. Domain

Domain himpunan *fuzzy* adalah keseluruhan nilai yang diijinkan dalam semesta pembicaraan dan boleh dioperasikan dalam suatu himpunan *fuzzy*.

Support atau pendukung himpunan *fuzzy* \tilde{A} . $Supp(\tilde{A})$, didalam semesta X , adalah himpunan tegas dari semua anggota X yang mempunyai derajat keanggotaan lebih dari nol seperti pada rumus (3).

$$Supp(\tilde{A}) = \{ x \in X \mid \mu_{\tilde{A}}(x) > 0 \} \dots \dots \dots (3)$$

Himpunan α -cut merupakan nilai ambang batas domain yang didasarkan pada nilai keanggotaan untuk tiap-tiap domain. Himpunan ini berisi semua nilai domain yang merupakan bagian dari himpunan *fuzzy* dengan nilai keanggotaan lebih besar atau sama dengan α sedemikian hingga :

- i. Untuk α -cut dapat dinyatakan sebagai rumus (4).

$$\tilde{A}_\alpha = \{ x \in X \mid \mu_{\tilde{A}}(x) \geq \alpha \} \dots \dots \dots (4)$$

- ii. Untuk *strong* α -cut dapat dinyatakan sebagai rumus (5).

$$\tilde{A}_{+\alpha} = \{ x \in X \mid \mu_{\tilde{A}}(x) > \alpha \} \dots \dots \dots (5)$$

Tinggi (*height*) suatu himpunan *fuzzy* \tilde{A} di dalam semesta X , yang dilambangkan dengan $h(\tilde{A})$, adalah himpunan yang menyatakan derajat keanggotaan tertinggi dalam himpunan *fuzzy* tersebut dengan persamaan seperti pada rumus (6).

$$h(\tilde{A}) = \sup \{ \mu_{\tilde{A}}(X) \} \dots \dots \dots (6)$$

Metode Mamdani adalah metode yang paling sering digunakan karena metode ini merupakan metode yang pertama kali dibangun dan berhasil diterapkan dalam rancang bangun sistem kontrol

menggunakan teori himpunan *fuzzy*.

Metode Mamdani sering juga dikenal dengan nama metode *Min-Max*. Metode ini diperkenalkan oleh Ebrahim Mamdani pada tahun 1975. Untuk mendapatkan *output* diperlukan empat tahapan, yaitu [6]:

1. Pembentukan himpunan *fuzzy*, baik variabel *input* maupun variabel *output* dibagi menjadi satu atau lebih himpunan *fuzzy*.
2. Aplikasi fungsi implikasi. Fungsi implikasi yang di gunakan adalah *Min*.
3. Komposisi aturan. Tidak seperti penalaran monoton, apabila sistem terdiri dari beberapa aturan, maka inferensi diperoleh dari kumpulan dan korelasi antar aturan. Ada tiga metode yang digunakan dalam melakukan inferensi sistem *fuzzy*, yaitu: *Max*, *aditive* dan probabilistik OR (*probor*).
4. Penegasan (*defuzzification*). Pengendali logika *fuzzy* harus mengubah variabel keluaran *fuzzy* menjadi nilai- nilai tegas yang dapat digunakan untuk mengendalikan sistem. Proses ini disebut penegasan (*defuzzification*). Input dari proses penegasan ini adalah suatu himpunan *fuzzy* yang diperoleh dari komposisi aturan-aturan *fuzzy*, sedangkan *output* yang dihasilkan merupakan suatu bilangan pada *domain* himpunan *fuzzy*. Sehingga jika diberikan suatu himpunan *fuzzy* dalam *range* tertentu, maka harus dapat diambil suatu nilai *crisp* tertentu sebagai *output*.

2. METODOLOGI PENELITIAN

Metodologi yang digunakan dalam penelitian ini meliputi beberapa langkah sebagai berikut, yaitu :



Gambar 2. Metodologi Penelitian

1. Melakukan pengumpulan data yang dibutuhkan dalam melakukan perhitungan dan analisis masalah. Data yang dikumpulkan meliputi data tridharma perguruan tinggi dosen.
2. Membentuk himpunan fuzzy, pada metode mamdani baik variabel input maupun output dibagi menjadi satu atau lebih himpunan fuzzy.
3. Aplikasi fungsi Implikasi, pada metode mamdani fungsi implikasi yang digunakan untuk tiap-tiap aturan adalah fungsi min.
4. Penegasan (defuzzy), proses penegasan (defuzzyfikasi) dengan menggunakan bantuan perangkat lunak *MATLAB Online*, serta dengan menggunakan fasilitas yang disediakan pada *toolbox fuzzy*.
5. Menarik kesimpulan dari hasil pengolahan data.

3. HASIL DAN PEMBAHASAN

Dalam penerapan *Fuzzy Logic metode mamdani* untuk sistem pendukung keputusan kinerja dosen, pertama kali dilakukan adalah menentukan keputusan Kinerja Dosen (*output*), dan diperlukan beberapa kriteria (Tridarma Perguruan Tinggi) yaitu: Pengajaran, Penelitian, dan Pengmas (Pengabdian

Langkah-langkahnya adalah sebagai berikut:

1. Menyusun konsep sistem kinerja dosen.
Kriteria **Pengajaran** (*input1*) dibagi menjadi tiga kategori yaitu:

Tabel 1. Pengajaran (*Input1*)

<i>INPUT1</i>	
PENGAJARAN	KETERANGAN
Rendah	0 - 35
Sedang	31 - 85
Tinggi	81 - 100

Kriteria **Penelitian** (*input 2*) dibagi menjadi tiga kategori yaitu:

Tabel 2. Penelitian (*Input2*)

<i>INPUT 2</i>	
PENELITIAN	KETERANGAN
Kurang	0 - 35
Sedang	31 - 85
Baik	81 - 100

Kriteria **Pengmas** (*input 3*) dibagi menjadi dua kategori yaitu:

Tabel 3. Pengmas (*Input3*)

<i>INPUT 3</i>	
PENGMAS	KETERANGAN
Cukup	0 - 55
Baik	51 - 100

Keluaran **Kinerja Dosen** (*Output*) dibagi menjadi tiga kategori yaitu:

Tabel 4. Kinerja_Dosen (*Output*)

<i>OUTPUT</i>	
KINERJA DOSEN	KETERANGAN
Cukup	0 - 35
Baik	31 - 85
Baik Sekali	81 - 100

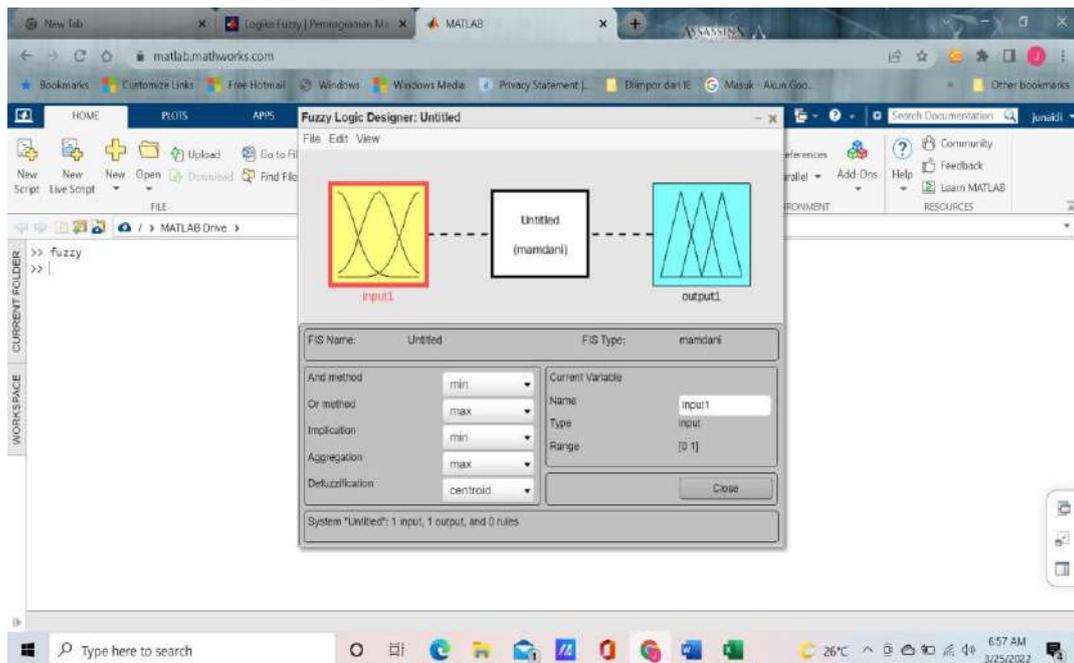
Sehubungan dengan keadaan masukan (*input*) dan keluaran (*output*) yang diinginkan maka ditetapkan beberapa aturan (*rules*) adalah sebagai berikut:

Tabel 5. Aturan (Rules)

NO.	INPUT			OUTPUT
	PENGAJARAN	PENELITIAN	PENGMAS	KINERJA_DOSEN
1	Rendah	Kurang	Cukup	Cukup
2	Rendah	Kurang	Baik	Cukup
3	Rendah	Sedang	Cukup	Cukup
4	Rendah	Sedang	Baik	Cukup
5	Rendah	Baik	Cukup	Cukup
6	Rendah	Baik	Baik	Cukup
7	Sedang	Kurang	Cukup	Cukup
8	Sedang	Kurang	Baik	Cukup
9	Sedang	Sedang	Cukup	Baik
10	Sedang	Sedang	Baik	Baik
11	Sedang	Baik	Cukup	Baik
12	Sedang	Baik	Baik	Baik
13	Tinggi	Kurang	Cukup	Baik
14	Tinggi	Kurang	Baik	Baik
15	Tinggi	Sedang	Cukup	Baik Sekali
16	Tinggi	Sedang	Baik	Baik Sekali
17	Tinggi	Baik	Cukup	Baik Sekali
18	Tinggi	Baik	Baik	Baik Sekali

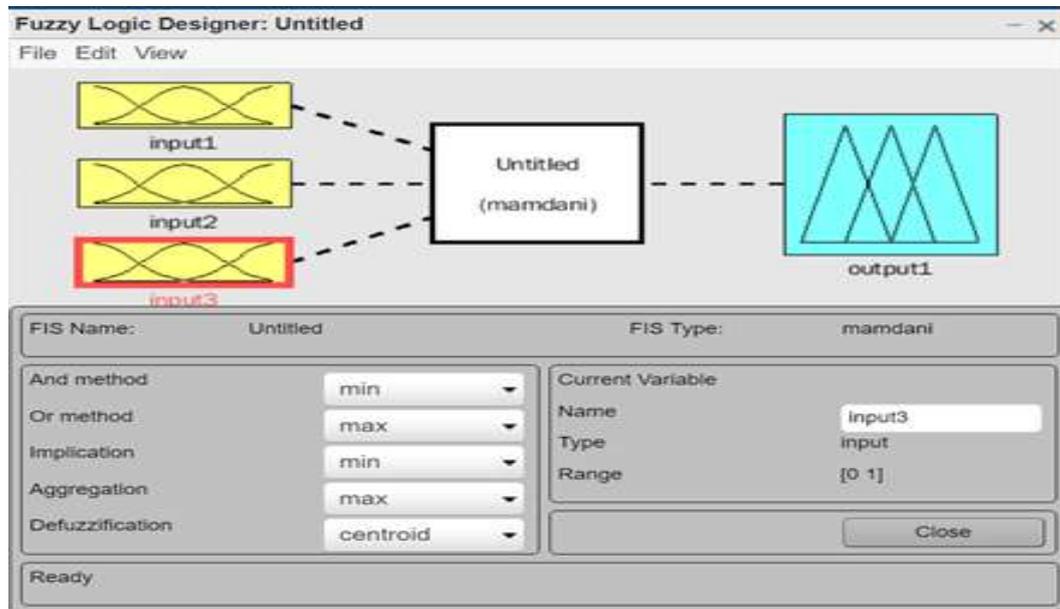
2. Setelah konsep kinerja dosen dibentuk, maka kita dapat membuat pemrogramannya dengan menggunakan *MATLAB Online*.

Ketikhlah “fuzzy” pada *command window* untuk membuka jendela *Fuzzy Inference System (FIS) editor*, sehingga muncul tampilan seperti gambar berikut ini:



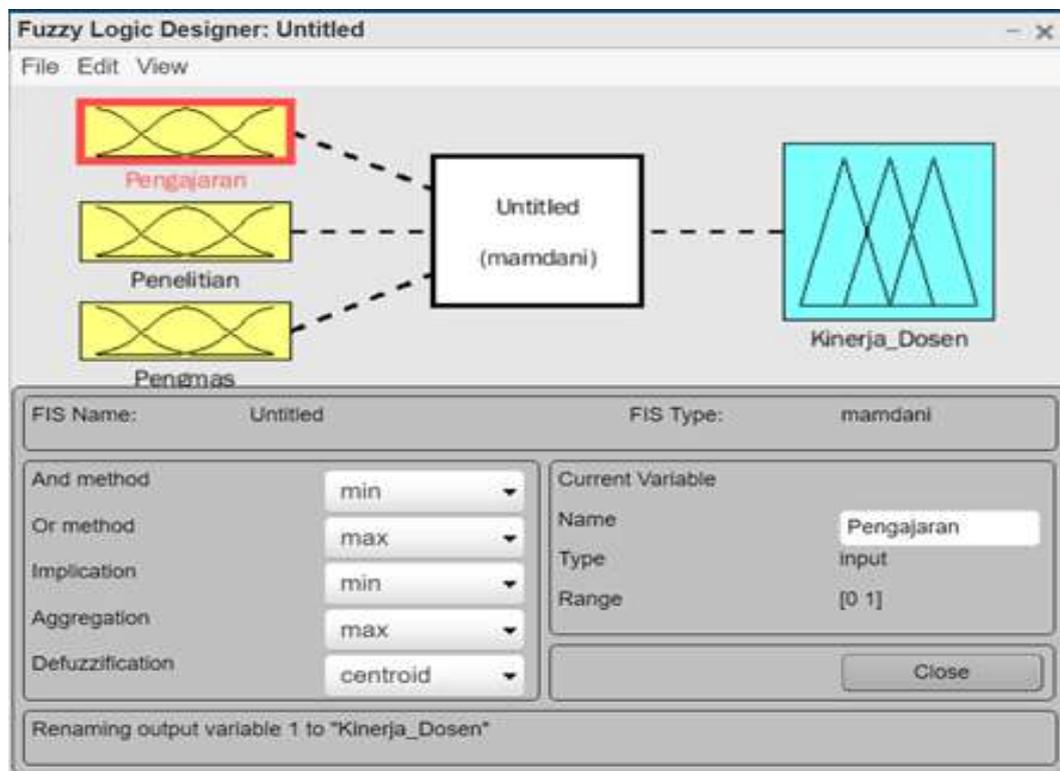
Gambar 3. Tampilan *Fuzzy Logic Designer*

- Pilih *edit* >> *add variable* >> *input* untuk menambah variabel *input* yang kedua. Ulangi untuk menambah input yang ketiga, sehingga akan muncul tampilan seperti pada gambar di bawah ini:



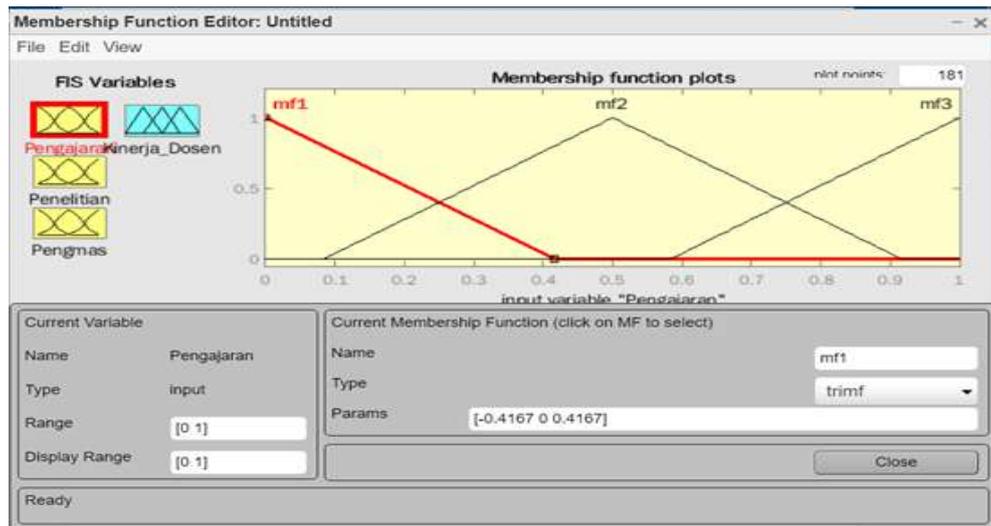
Gambar 4. Tampilan Variabel *Input* dan *Output*

- Ubahlah nama *input1* menjadi **Pengajaran**, *input2* menjadi **Penelitian**, *input3* menjadi **Pengmas**, dan *output1* menjadi **Kinerja_Dosen**



Gambar 5. Tampilan Variabel *Input* (Pengajaran, Penelitian, Pengmas) dan *Output* (Kinerja_Dosen)

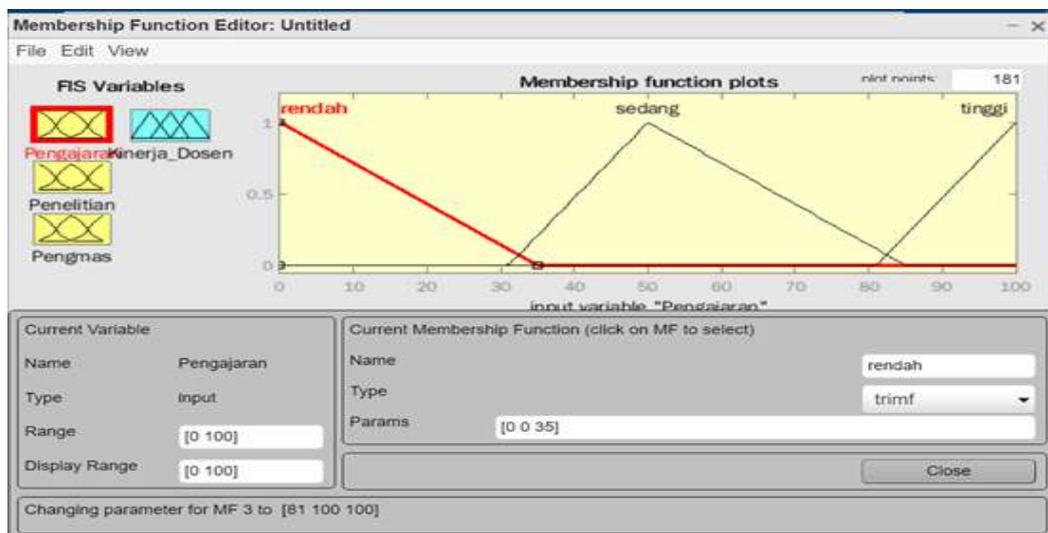
- Pilih *edit* >> *membership function* untuk membuat fungsi keanggotaan setiap variable. Sehingga akan muncul tampilan *Membership Function Editor* seperti pada gambar berikut ini:



Gambar 6. Tampilan *Membership Function Editor*

- 6. Pada variabel **Pengajaran**, ubahlah **Range** menjadi **[0 100]**,
 Name **mf1** menjadi **rendah**, Type **trimf**, Params **[0 0 35]**
 Name **mf2** menjadi **sedang**, Type **trimf**, Params **[31 50 85]**
 Name **mf3** menjadi **tinggi**, Type **trimf**, Params **[81 100 100]**

Sehingga tampilan variabel **Pengajaran** akan tampak seperti pada gambar berikut ini:



Gambar 7. Tampilan variabel **Pengajaran**

- Pada variabel **Penelitian**, ubahlah **Range** menjadi **[0 100]**,
 Name **mf1** menjadi **kurang**, Type **trimf**, Params **[0 0 35]**
 Name **mf2** menjadi **sedang**, Type **trimf**, Params **[31 50 85]**
 Name **mf3** menjadi **baik**, Type **trimf**, Params **[81 100 100]**

Pada variabel **Pengmas**, ubahlah **Range** menjadi **[0 100]**,

Pilih **mf3**, kemudian pilih **edit >> remove selected MF** untuk menghapus **membership function** karena pada variabel **Pengmas** hanya ada dua kategori. Selanjutnya ubahlah:

Name **mf1** menjadi **cukup**, Type **trimf**, Params [0 0 55]

Name **mf2** menjadi **baik**, Type **trimf**, Params [51 100 100]

Pada variabel **Kinerja_Dosen**, ubahlah **Range** menjadi

[0 100],

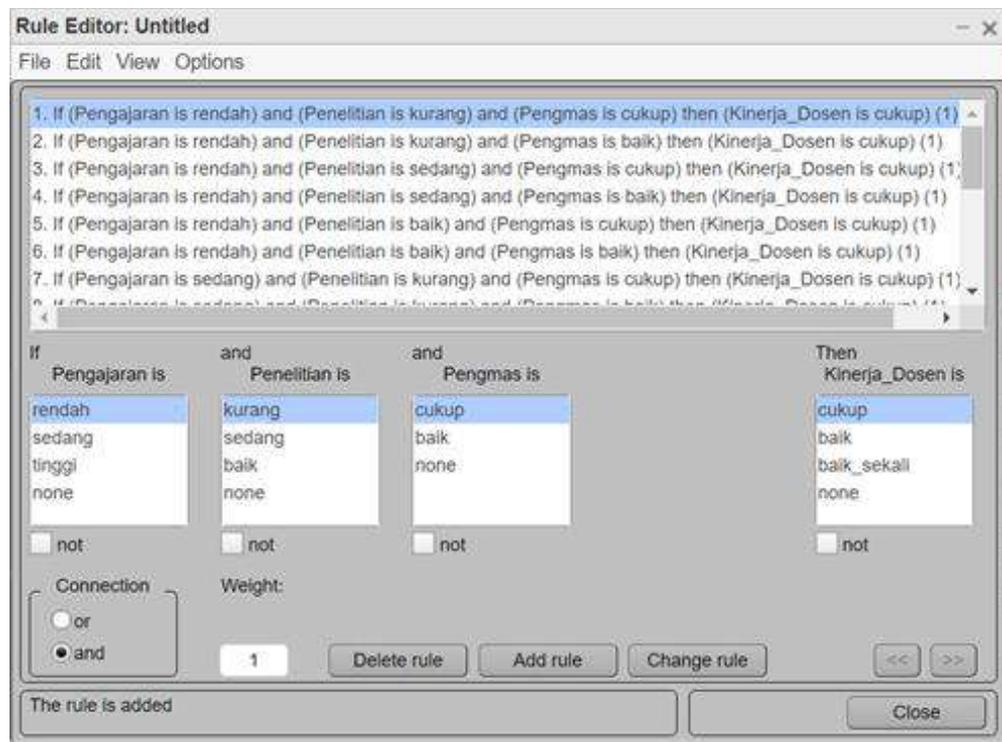
Name **mf1** menjadi **cukup**, Type **trimf**, Params [0 0 35]

Name **mf2** menjadi **baik**, Type **trimf**, Params [31 50 85]

Name **mf3** menjadi **baik sekali**, Type **trimf**, Params [81 100 100]

7. Pilih **edit >> rules** untuk membuka jendela **rule editor**

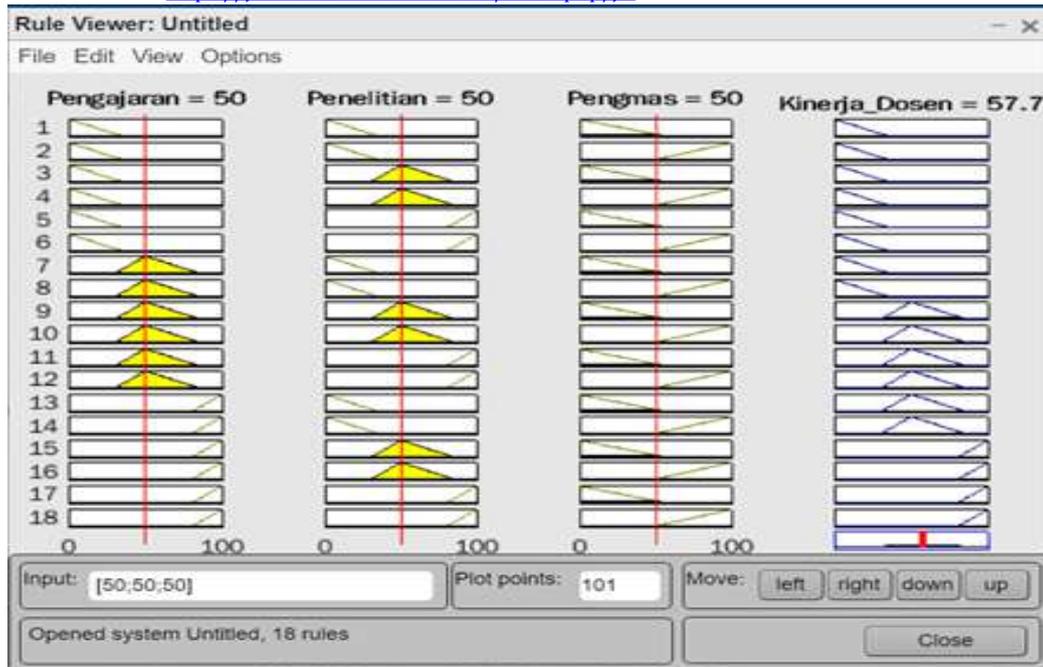
Buatlah aturan pada **rule editor** sesuai dengan konsep sistem kinerja dosen yang sebelumnya telah dibuat. Misalnya **if (Pengajaran is rendah) and (Penelitian is rendah) and (Pengmas is cukup) then (Kinerja_Dosen is cukup)**, dan seterusnya sampai dengan **18 rules**.



Gambar 8. Pembuatan Aturan-aturan (*Rules*) pada *Rule Editor*

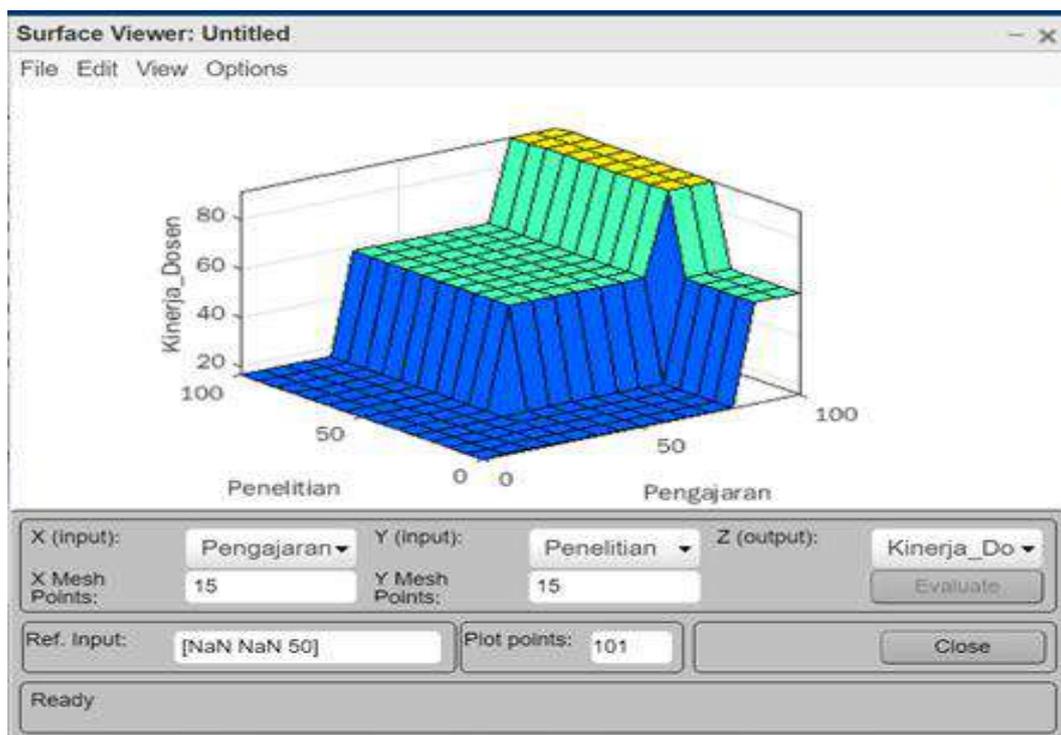
8. Pilih **view >> rules**, untuk melihat hasil **rules** yang telah kita buat.

Kita dapat menggeser-geser nilai **Pengajaran** (*input1*), nilai **Penelitian** (*input2*), dan nilai **Pengmas** (*input3*) sehingga menghasilkan nilai keluaran pada **Kinerja_Dosen** (*output*)



Gambar 9. Tampilan Hasil Rules

- Pilih view >> *surface*, untuk melihat grafik 3D antara Pengajaran, penelitian, Pengmas, dan Kinerja_Dosen, sehingga akan muncul tampilan seperti pada gambar berikut ini:

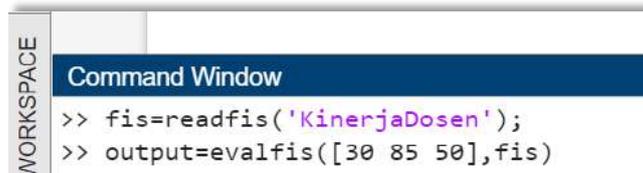


Gambar 10. Tampilan grafik 3D antara Pengajaran, penelitian, Pengmas, dan Kinerja_Dosen

- Simpanlah FIS yang telah dibuat dengan cara mengklik *file >> export >> to file*

Misalnya simpan dengan nama “KinerjaDosen.fis”.

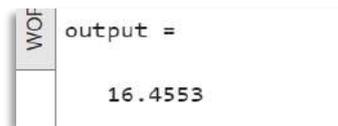
- Untuk mengecek hasil keluaran dari FIS yang telah dibuat, dapat kita lakukan dengan mengetik kode berikut pada *Command Window*:



```
WORKSPACE  
Command Window  
>> fis=readfis('KinerjaDosen');  
>> output=evalfis([30 85 50],fis)
```

Gambar 11. Kode Program

Hasilnya adalah:



```
WORKSPACE  
output =  
16.4553
```

Gambar 12. Hasil Keluaran (*Output*)

Nilai hasil keluaran diatas dapat diartikan sebagai berikut:

Jika **Pengajaran = 30 (rendah)** dan **Penelitian = 85 (baik)** dan **Pengmas = 50 (cukup)**, maka **Kinerja_Dosen = 16,4553 (cukup)**

Disini dapat diambil kesimpulan bahwa hasil keluaran **FIS** sesuai dengan konsep sistem kinerja dosen yang sebelumnya telah dibuat.

4. KESIMPULAN

Sistem pendukung keputusan yang dibuat dapat digunakan untuk membantu pengambilan keputusan penentuan kinerja dosen dengan menggunakan data dosen dengan kriteria pengajaran, penelitian dan pengmas. Dari hasil percobaan dengan menggunakan perangkat lunak *MATLAB Online* dengan *toolbox logical fuzzy* dalam penerapan terhadap masalah penentuan kinerja dosen, logika *fuzzy* metode mamdani dapat memberikan hasil yang lebih baik dan lebih mudah dibandingkan perhitungan secara manual.

Untuk meningkatkan kinerja dan menyempurnakan sistem pendukung keputusan yang telah dibuat, diharapkan dibuatnya sistem pendukung keputusan lainnya sehingga proses pengambilan keputusan semakin mudah dan cepat. Peneliti berharap agar hasil penelitian ini bisa dikembangkan dan dilengkapi lagi agar dapat digunakan sebagai alat evaluasi bagi perguruan tinggi dan instansi terkait.

REFERENCES

- [1] Mulyadi. (2007). Sistem Perencanaan dan Pengendalian Manajemen. *Jakarta : Penerbit Salemba Empat.*
- [2] Turban, E., Aronson, J. E., & Liang, T. P. (2005). Decision Support System. *Prentice Hall, 7th Edition.*
- [3] Sudradjat. (2007). Mathematical Programming Models for Portfolio Selection. *editura universităţii din Bucureşti.*
- [4] Suyanto. (2011). Artificial Intelligence Searching, Reasoning, Planning dan Learning. *Informatika Bandung*, edisi revisi, pp. 97.
- [5] Kusumadewi, S. & Purnomo, H. (2013). Aplikasi Logika Fuzzy untuk Pendukung Keputusan. *Yogyakarta : Penerbit Graha Ilmu.*
- [6] Rani M. (2018). Logika Fuzzy Untuk Menentukan Asupan Kalori pada Terapi Diet Terhadap Penderita Obesitas. *J. Inov. Tek. Inform.*, Vol. 1, No. 2.
- [7] Firmansyah, A. (2007). Dasar-dasar Pemrograman Matlab. IlmuKomputer. com.

Penerapan Decision Tree Dengan Algoritma C4.5 Untuk Menentukan Rekomendasi Kenaikan Jabatan Karyawan

Elmi Devia

Fakultas Teknik, Sistem Informasi, Universitas Krisnadwipayana, Jakarta, Indonesia

Email: elmidevia@unkris.ac.id

Abstrak- Dalam perencanaan dan usaha untuk memenuhi kebutuhan Sumber Daya Manusia (SDM) dilakukan kenaikan jabatan karyawan yang dikelola secara profesional sehingga dapat menentukan mutu dan kesuksesan perusahaan. Seleksi yang baik dan akurat dari kenaikan jabatan karyawan akan menghasilkan SDM yang berkualitas bagi perusahaan tersebut. Apabila karyawan memiliki kemampuan dan kualitas yang baik maka karyawan diberikan nilai baik. Kesalahan dalam memberikan kenaikan jabatan karyawan sangat besar dampaknya bagi perusahaan karena berpengaruh langsung pada produktivitas dan kinerja finansial perusahaan. Tujuan penelitian ini dimana ingin mengetahui bagaimana pelaksanaan penilaian kenaikan jabatan karyawan dengan menggunakan metode *Decision Tree* algoritma C4.5. Masalah yang diselesaikan adalah cara penentuan rekomendasi kenaikan jabatan karyawan dengan menggunakan data karyawan dengan beberapa indikator kinerja karyawan, yaitu : Kedisiplinan, Kerjasama, Inisiatif, dan Kepemimpinan. Langkah awal penyelesaian masalah penilaian kenaikan jabatan karyawan dengan menggunakan metode *decision tree algoritma C4.5* adalah tahap *preprocessing data*. Langkah kedua yaitu menentukan nilai atribut, dimana dalam data sampel tentukan dulu *node* terpilih, yaitu dengan menghitung nilai informasi *gain* masing-masing atribut untuk menentukan *node* terpilih, gunakan nilai informasi *gain* yang paling besar. Kemudian akan dilakukan perhitungan nilai *entropy* dan *gain* masing-masing atribut. Nilai *gain* yang tertinggi ada pada atribut Kepemimpinan dengan nilai 0.509185925 maka Kepemimpinan akan menjadi sebuah akar pertama, selanjutnya perhitungan *node* ke 2 dengan nilai *gain* tertinggi adalah pada atribut Kedisiplinan yaitu 0.566509507, maka Kedisiplinan akan menjadi akar selanjutnya. Karena tidak ditemukan nilai tertinggi dari *Gain* pada atribut Kerjasama dan Inisiatif, sehingga klasifikasi akar terakhir dari pohon keputusan yaitu atribut Kedisiplinan. Langkah ketiga dilakukan pemrosesan *decision tree* dengan menggunakan *software* RapidMiner Studio 9.10 dengan metode *decision tree*. Selanjutnya terbentuk *example set*, dimana *example set* adalah data set yang sudah terupload di aplikasi *RapidMiner Studio 9.10*. Setelah itu ditampilkan *Graph view* yang menunjukkan hasil dari klasifikasi dengan percabangan yang dapat dihasilkan kesimpulan. Hasil akhir dari *decision tree* adalah didapatnya aturan-aturan (*rules*) yang didapat dari hasil pohon keputusan dan deskripsi *tree*, untuk menganalisa faktor rekomendasi jabatan berdasarkan atribut. Dari hasil percobaan diatas untuk menganalisa faktor rekomendasi kenaikan jabatan maka dapat disimpulkan faktor yang menjadi penyebab utama karyawan di rekomendasikan untuk kenaikan jabatan adalah Kepemimpinan dan Kedisiplinan.

Kata Kunci: *Decision Tree*; Karyawan; Kenaikan jabatan; Algoritma C4.5; *RapidMiner*.

In planning and in efforts to meet the needs of Human Resources (HR), promotions are made for employees who are managed professionally so that they can determine the quality and success of the company. Good and accurate selection of employee promotions will produce quality human resources for the company. If employees have good abilities and qualities, then employees are given good grades. Mistakes in giving employee promotions have a big impact on the company because they have a direct effect on the company's productivity and financial performance. The purpose of this research is to find out how to carry out the evaluation of employee promotions using the C4.5 Algorithm Decision Tree method. The problem to be solved is how to determine recommendations for employee promotion by using employee data with several employee performance indicators, namely: Discipline, Cooperation, Initiative, and Leadership. The initial step in solving the problem of assessing employee promotion using the decision tree algorithm C4.5 method is the data preprocessing stage. The second step is to determine the attribute values, where in the sample data first determine the selected node, namely by calculating the gain information value of each attribute to determine the selected node, use the highest gain information value. Then the entropy and gain values of each attribute will be calculated. The highest gain value is in the Leadership attribute with a value of 0.509185925, then Leadership will be the first root, then the calculation of the 2nd node with the highest gain value is the Discipline attribute, namely 0.566509507, then Discipline will become the next root. Because there was no highest value of Gain on the Collaboration and Initiative attribute, the last root classification of the decision tree was the Discipline attribute. The third step is to process the decision tree using RapidMiner Studio 9.10 software with the decision tree method. Then an example set is formed, where the example set is a data set that has been uploaded to the RapidMiner Studio 9.10 application. After that, a Graph view is displayed which shows the results of the classification with branching which can be concluded. The end result of the decision tree is the obtaining of rules (rules) obtained from the results of the decision tree and description tree, to analyze the factor of job recommendations based on attributes. From the results of the experiment above to analyze the recommendation factors for promotion, it can be concluded that the factors that are the main causes of employees being recommended for promotion are Leadership and Discipline.

Keywords: Decision Tree; Employee; Promotion; C4.5 Algorithm; RapidMiner.



1. PENDAHULUAN

Salah satu elemen dalam perusahaan yang sangat penting adalah karyawan (Sumber Daya Manusia / SDM). Pengelolaan SDM dari suatu perusahaan sangat mempengaruhi banyak aspek penentu keberhasilan kerja dari perusahaan tersebut. Jika karyawan atau SDM dapat diorganisir dengan baik, maka diharapkan perusahaan dapat menjalankan semua proses usahanya dengan baik. Sebagai salah satu elemen perusahaan, manajemen SDM tidak dapat dipisahkan dari bidang manajemen lainnya untuk mencapai tujuan perusahaan.

Dalam perencanaan dan usaha untuk memenuhi kebutuhan SDM dilakukan kenaikan jabatan karyawan yang dikelola secara profesional sehingga dapat menentukan mutu dan kesuksesan perusahaan. Seleksi yang baik dan akurat dari kenaikan jabatan karyawan akan menghasilkan SDM yang berkualitas bagi perusahaan tersebut.

Apabila karyawan memiliki kemampuan dan kualitas yang baik maka karyawan diberikan nilai baik. Subjektivitas terjadi karena pengambil keputusan belum bisa mendefinisikan dengan baik dalam menilai kelayakan karyawan. Maka keputusan yang di ambil dapat kenaikan jabatan karyawan yang tidak memenuhi kualifikasi. Kesalahan dalam memberikan kenaikan jabatan karyawan sangat besar dampaknya bagi perusahaan karena berpengaruh langsung pada produktivitas dan kinerja finansial perusahaan.

Sumber daya manusia di dalam suatu organisasi perusahaan merupakan hal yang sangat penting untuk mendukung kemajuan dan kualitas perusahaan dalam mencapai tujuan. Kenaikan jabatan merupakan suatu faktor yang sangat penting bagi perencanaan karir pegawai dan juga untuk meremajakan suatu posisi jabatan agar diduduki oleh seseorang yang mempunyai kriteria-kriteria yang cocok untuk menempati suatu jabatan yang diusulkan [1]. Pengisian jabatan yang kosong pada proses kenaikan jabatan sring mengalami kesulitan karena pengajuan calon kandidat yang bisa menempati jabatan tersebut dengan cara pencocokan profil karyawan dan profil jabatan kurang terdefinisi dengan baik [2]. Aspek-aspek Penilaian sistem pendukung keputusan tersebut dirancang sedemikian rupa sehingga pengguna yang dalam hal ini adalah manajer bagian sumber daya manusia bisa menentukan aspek-aspek penilaian sendiri secara dinamis sehingga sistem pendukung keputusan tersebut bisa dipakai lebih luas [3].

Decision tree (pohon keputusan) adalah salah satu metode belajar yang sangat populer dan banyak digunakan secara praktis. Metode ini merupakan metode yang berusaha menemukan fungsi-fungsi pendekatan yang bernilai diskrit [4].

Decision tree merupakan salah satu metode klasifikasi yang kuat dan terkenal. Metode *decision tree* mengubah fakta yang besar menjadi pohon keputusan yang merepresentasikan aturan, aturan tersebut dapat dengan mudah untuk diinterpretasi oleh manusia. *Decision tree* juga berguna untuk mengeksplorasi data, menemukan hubungan tersembunyi antara sejumlah variabel input dengan sebuah variabel target [5].

Model *decision tree* terdiri dari sekumpulan aturan untuk membagi sejumlah populasi yang heterogen menjadi lebih kecil (homogen) dengan memperhatikan variabel tujuannya. Variabel tujuan biasanya dikelompokkan dengan passti dan model *decision tree* lebih mengarah pada perhitungan probabilitas dari tiap-tiap *record* terhadap kategori tersebut atau untuk mengklasifikasikan *record* dengan mengelompokkannya dalam satu kelas. Sebuah *decision tree* dapat dibangun dengan menerapkan salah satu algoritma *decision tree* untuk memodelkan himpunan data yang belum terklasifikasi kelasnya [6].

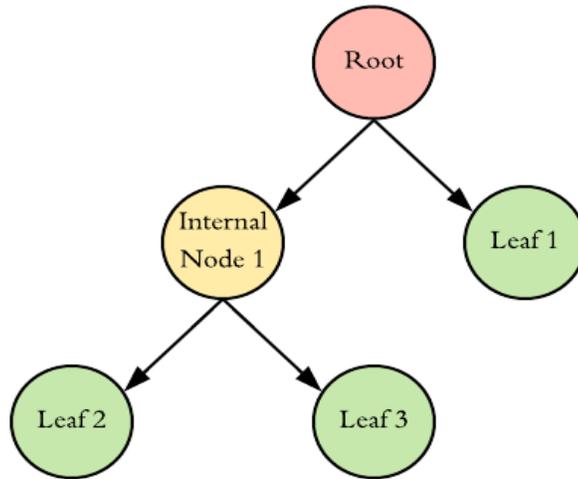
Sistem Pendukung Keputusan (SPK) adalah alat sistem informasi populer yang mendukung proses pengambilan keputusan. SPK telah didefinisikan sebagai Sistem Informasi berbasis komputer yang interaktif dan dapat disesuaikan yang juga mendukung masalah manajemen yang tidak terstruktur [7]. Melalui penggunaan SPK, para pengambil keputusan dapat menemukan solusi untuk berbagai masalah. Ini termasuk masalah semi-terstruktur hingga tidak terstruktur yang melibatkan banyak atribut, tujuan, atau sasaran [8].

Pemecahan masalah tak hanya mengacu ke solusi dari area masalah/kesulitan/kesulitan tapi mencakup juga penyelidikan mengenai kesempatan-kesempatan yang ada. *Decision tree* merupakan salah satu metode yang digunakan untuk memperoleh suatu pemecahan masalah.

Oleh karena itu, sangat penting dibangun sebuah sistem pengambilan keputusan yang terkomputerisasi yang dapat memudahkan dalam memberikan kenaikan jabatan karyawan yang sesuai kebutuhan dan kriteria perusahaan dengan menggunakan metode *decision tree* dengan algoritma C4.5.

Decision Tree (Pohon Keputusan) digunakan untuk memodelkan persoalan yang terdiri dari serangkaian keputusan yang mengarah kepada solusi. Tiap simpul dalam menyatakan keputusan sedangkan daun menyatakan solusi. Skema dan struktur *decision tree* adalah salah satu pemodelan dari struktur menurut graf. Secara umum *decision tree* adalah gambaran pemodelan dari persoalan yang terdiri beberapa serangkaian keputusan yang mengarah pada solusi, dalam menyatakan keputusan dan memberikan sebuah solusi [9].

Konsep dari *decision tree* adalah mengubah tumpukan data menjadi sebuah *decision tree* yang



Gambar 1. Konsep Dasar *Decision Tree*
(Sumber : Tahir, 2019)

Pada Gambar 1. menggambarkan konsep dasar dari *decision tree*. *Decision tree* terdiri dari *node* untuk membentuk pohon berakar. Dari pohon tersebut diarahkan oleh *node* yang disebut akar (*root*) yang tidak mempunyai masukan (*input*) tetapi memiliki dua atau lebih keluaran (*output*). Jika *node* memiliki satu masukan (*input*) yang tepat dan *node* memiliki dua atau lebih keluaran (*output*), maka disebut dengan *internal node*. *Node* yang terakhir dan hanya memiliki masukan (*input*) disebut dengan *leaf node* atau biasanya disebut juga dengan *decision node* atau *terminal node*. *Leaf node* menunjukkan sebuah label atau hasil dari klasifikasi atau kelas keputusan [10].

Ada beberapa algoritma yang bisa digunakan pada *decision tree*, dimana salah satunya adalah algoritma C4.5. Algoritma C4.5 merupakan prosedur yang digunakan dalam membangun *decision tree*. *Decision tree* dibangun dengan cara membagi data secara rekursif hingga tiap bagian terdiri dari data yang berasal dari kelas yang sama. *Decision tree* akan dibuat menggunakan algoritma C4.5 dengan langkah-langkah sebagai berikut [11]:

1. Memilih atribut sebagai akar.
2. Membuat cabang untuk tiap-tiap nilai.
3. Membagi kasus dalam cabang.
4. Ulangi proses untuk masing-masing cabang sampai semua kasus pada cabang memiliki kelas yang sama. Dalam memilih atribut sebagai akar, didasarkan pada nilai *Gain Ratio* tertinggi dari atribut-atribut yang ada. Namun sebelum menentukan *Gain Ratio*, dibutuhkan perhitungan untuk mencari nilai *Entropy* dan *Gain*. Persamaan yang digunakan untuk menentukan *Entropy* dan *Gain* adalah sebagai berikut:

Persamaan untuk mencari nilai *Entropy*:

$$Entropy(S) = \sum_{i=1}^n -p_i * \log_2 p_i$$

Keterangan:

S = Himpunan kasus

n = Jumlah partisi dalam S

p_i = Proporsi dari S_i terhadap S

Persamaan untuk mencari nilai *Gain*:

$$Gain(S,A) = Entropy(s) - \sum_{i=1}^n \frac{|S_i|}{|S|} * Entropy(S_i)$$

Keterangan:

S = Himpunan kasus

A = Atribut

n = Jumlah partisi atribut A

$|S_i|$ = Jumlah kasus pada partisi ke- i

$|S|$ = Jumlah kasus dalam S

RapidMiner adalah aplikasi atau perangkat lunak yang berfungsi sebagai alat pembelajaran dalam ilmu *data mining*. RapidMiner merupakan perangkat lunak yang bersifat terbuka (*open source*). RapidMiner adalah sebuah solusi untuk melakukan analisis terhadap *data mining*, *text mining* dan analisis prediksi. RapidMiner menggunakan berbagai teknik deskriptif dan prediksi dalam memberikan wawasan kepada pengguna sehingga dapat membuat keputusan yang paling baik. RapidMiner memiliki kurang lebih 500 operator *data mining*, termasuk operator untuk *input*, *output*, *data preprocessing* dan visualisasi.

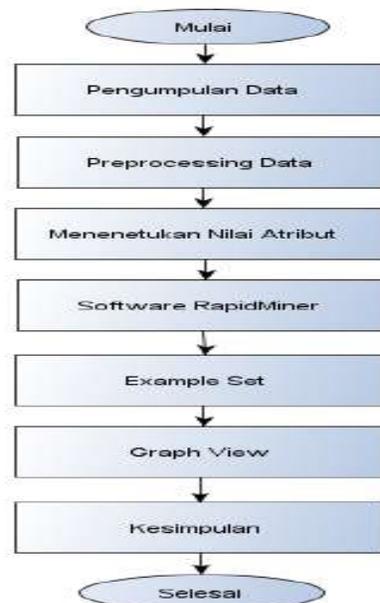
RapidMiner merupakan *software* yang berdiri sendiri untuk analisis data dan sebagai mesin *data mining* yang dapat diintegrasikan pada produknya sendiri. RapidMiner ditulis dengan menggunakan bahasa java sehingga dapat bekerja di semua sistem operasi. RapidMiner sebelumnya bernama YALE (*Yet Another Learning Environment*), dimana versi awalnya mulai dikembangkan pada tahun 2001 oleh Ralf Klinkenberg, Ingo Mierswa, dan Simon Fischer di *Artificial Intelligence Unit* dari *University of Dortmund*. RapidMiner didistribusikan di bawah lisensi AGPL (*GNU Affero General Public License*) versi 3. Hingga saat ini telah ribuan aplikasi yang dikembangkan menggunakan RapidMiner di lebih dari 40 negara. RapidMiner sebagai *software open source* untuk *data mining* tidak perlu diragukan lagi karena software ini sudah terkemuka di dunia.

RapidMiner menyediakan UI untuk mendesain pipa analisis, di mana akan menghasilkan file XML yang dapat menjelaskan proses analisis yang ingin diterapkan oleh pengguna ke data. RapidMiner akan membaca file ini untuk menjalankan analisa secara otomatis [12]. RapidMiner menyediakan tampilan (UI) yang ramah pengguna, sehingga memudahkan pengguna saat menggunakannya.

RapidMiner menyediakan GUI (*Graphic User Interface*) untuk merancang sebuah pipeline analitis. GUI ini akan menghasilkan file XML (*Extensible Markup Language*) yang mendefinisikan proses analitis keinginan pengguna untuk diterapkan ke data. File ini kemudian dibaca oleh RapidMiner untuk menjalankan analisis secara otomatis.

2. METODOLOGI PENELITIAN

Metode penelitian ini menggunakan alur penelitian yang dimulai dari pengumpulan data dan diakhiri dengan kesimpulan. Tahapan-tahapan yang harus dilakukan dalam melakukan penelitian tercakup dalam bagian metodologi penelitian yang dapat dilihat pada gambar 2.:



Gambar 2. Metodologi Penelitian



Penjelasan dari diagram alir metodologi penelitian di atas adalah:

1. Pengumpulan Data

Pengumpulan data penting untuk dilakukan sehingga diperoleh informasi yang berkaitan dengan objek penelitian guna menunjang proses penelitian.

2. Preprocessing Data

Pada tahap pra-proses dibutuhkan data keputusan terhadap rekomendasi jabatan. Berikut adalah data sampel yang berupa tabel yang disimpan dengan format Exel.

3. Menentukan Nilai Atribut

Dalam data sampel ditentukan terlebih dahulu *node* terpilih, yaitu dengan menghitung nilai informasi *gain* masing-masing atribut untuk menentukan *node* terpilih, gunakan nilai informasi *gain* yang paling besar. Kemudian akan dilakukan perhitungan nilai *entropy* dan *gain* masing-masing atribut.

4. Software RapidMiner

Pada tahap ini akan dilakukan pemrosesan *decision tree* dengan menggunakan *software RapidMiner Studio 9.10* dengan metode *decision tree*. Data yang diolah menggunakan *MS. Excel 2019*, sehingga di aplikasi *RapidMiner Studio 9.10* menggunakan operator *read excel* yang berfungsi untuk membaca file yang akan diolah.

5. Example Set

Example set adalah data set yang sudah terupload di aplikasi *RapidMiner Studio 9.10*.

6. Graph View

Graph view menunjukkan hasil dari klasifikasi dengan percabangan yang dapat dihasilkan kesimpulan.

7. Kesimpulan

Aturan-aturan (*rules*) yang didapat dari hasil pohon keputusan dan deskripsi *tree* diatas untuk menganalisa faktor rekomendasi jabatan berdasarkan atribut.

3. HASIL DAN PEMBAHASAN

Data yang akan digunakan dalam bentuk data sampel untuk menganalisa rekomendasi kenaikan jabatan. Data rekomendasi tersebut selanjutnya akan idakka pra proses untuk menghasilkan data khusus yang siap untuk dibentuk menjadi sebuah *decision tree*.

Penilaian kinerja dibutuhkan untuk mengetahui tingkat keberhasilan kinerja setiap karyawan. Penilaian kinerja digunakan untuk menilai keberhasilan atau kegagalan pelaksanaan kegiatan atau kebijaksanaan sesuai dengan tujuan yang telah ditetapkan dalam rangka mewujudkan misi dan visi perusahaan, ada beberapa hal yang dapat digunakan sebagai indikator penilaian kinerja, antara lain:

- Kedisiplinan
- Kerjasama
- Inisiatif
- Kepemimpinan

Preprocessing Data

Pada tahap preprocessing (pra-proses) dibutuhkan data keputusan terhadap rekomendasi kenaikan jabatan. Data yang dibutuhkan terdiri dari data karyawan dengan indikator kinerja yang terdiri dari : Kedisiplinan, Kerjasama, Inisiatif, dan Kepemimpinan, serta Rekomendasi. Berikut adalah data sampel yang berupa tabel yang disimpan dengan format Exel.

Tabel 1. Data Sampel

NO	KEDISIPLINAN	KERJASAMA	INISIATIF	KEPEMIMPINAN	REKOMENDASI
1	Cukup	Kurang Mampu	Bagus	Baik	YA
2	Bagus	Kurang Mampu	Bagus	Baik	YA
3	Bagus	Mampu	Bagus	Baik	YA
4	Bagus	Mampu	Bagus	Baik	YA
5	Kurang	Kurang Mampu	Cukup	Cukup	TIDAK
6	Bagus	Mampu	Kurang	Cukup	TIDAK

7	Cukup	Kurang Mampu	Bagus	Baik	YA
8	Bagus	Kurang Mampu	Bagus	Cukup	TIDAK
9	Bagus	Mampu	Bagus	Baik	YA
10	Bagus	Mampu	Bagus	Baik	YA
11	Cukup	Mampu	Cukup	Cukup	TIDAK
12	Bagus	Mampu	Kurang	Baik	YA
13	Cukup	Mampu	Bagus	Baik	YA
14	Bagus	Mampu	Bagus	Baik	YA
15	Bagus	Mampu	Bagus	Baik	YA
16	Kurang	Kurang Mampu	Cukup	Baik	TIDAK
17	Kurang	Kurang Mampu	Cukup	Baik	TIDAK
18	Bagus	Kurang Mampu	Cukup	Baik	YA
19	Cukup	Mampu	Cukup	Cukup	TIDAK
20	Cukup	Kurang Mampu	Bagus	Baik	YA

Menentukan Nilai Atribut

Dalam data sampel tentukan dulu *node* terpilih, yaitu dengan menghitung nilai informasi *gain* masing-masing atribut untuk menentukan *node* terpilih, gunakan nilai informasi *gain* yang paling besar. Kemudian akan dilakukan perhitungan nilai *entropy* dan *gain* masing-masing atribut, dapat dilihat pada tabel dibawah ini.

Tabel 2. Perhitungan *Entropy* dan *Gain* Node 1

Klasifikasi Penilaian	Jml. Kasus (S)	YA (S1)	TIDAK (S2)	ENTROPY	GAIN
TOTAL	20	13	7	0.934068055	
KEDISIPLINAN					0.282358166
Bagus	11	9	2	0.684038436	
Cukup	6	4	2	0.918295834	
Kurang	3	0	3	0	
KERJASAMA					0.023140813
Mampu	11	8	3	0.845350937	
Kurang Mampu	9	5	4	0.99107606	
INISIATIF					0.390771219
Bagus	12	11	1	0.41381685	
Cukup	6	1	5	0.650022422	
Kurang	2	1	1	1	
KEPEMIMPINAN					0.509185925
Baik	15	13	2	0.566509507	0.509185925
Cukup	5	0	5	0	

Dari tabel diatas dilihat nilai *gain* yang tertinggi ada pada atribut **Kepemimpinan** dengan nilai **0.509185925** maka Kepemimpinan akan menjadi sebuah akar pertama, kemudian akan dilanjutkan perhitungan *node* ke 2 dengan perhitungan pada tabel dibawah ini.



Tabel 3. Perhitungan *Entropy* dan *Gain* Node 2

Klasifikasi Penilaian	Jml. Kasus (S)	YA (S1)	TIDAK (S2)	ENTROPY	GAIN
TOTAL	15	13	2	0.566509507	
KEDISIPLINAN					0.566509507
Bagus	9	9	0	0	
Cukup	4	4	0	0	
Kurang	2	0	2	0	
KERJASAMA					0.163719908
Mampu	8	8	0	0	
Kurang Mampu	7	5	2	0.863120569	
INISIATIF					0.38285034
Bagus	11	11	0	0	
Cukup	3	1	2	0.918295834	
Kurang	1	1	0	0	

Tabel diatas menunjukkan perhitungan *node* 2 dengan nilai *gain* tertinggi adalah pada atribut **Kedisiplinan** yaitu **0.566509507**, maka Kedisiplinan akan menjadi akar selanjutnya, perhitungan akan dilanjut ke *node* 3 dibawah ini.

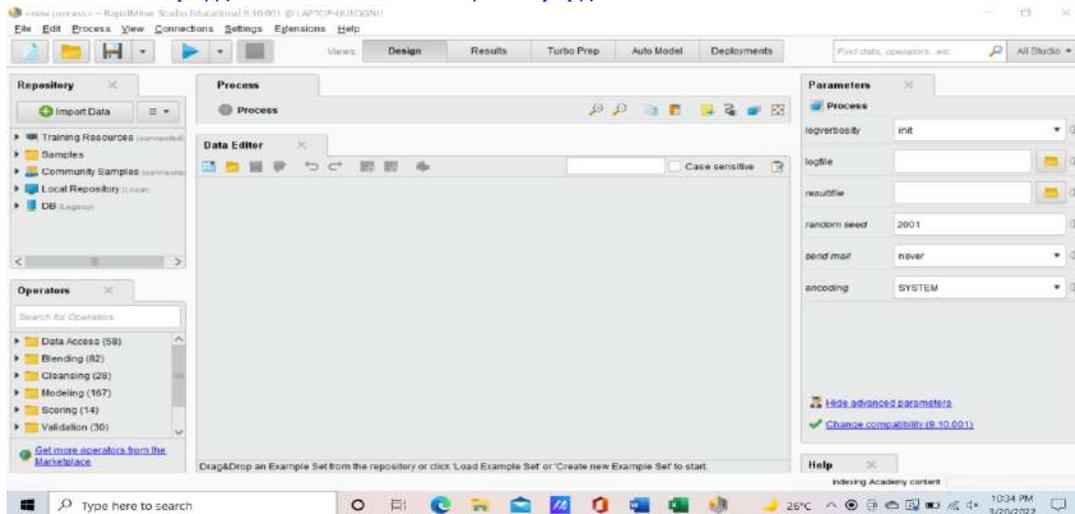
Tabel 4. Perhitungan *Entropy* dan *Gain* Node 3

Klasifikasi Penilaian	Jml. Kasus (S)	YA (S1)	TIDAK (S2)	ENTROPY	GAIN
TOTAL	13	13	0	0	
KERJASAMA					0
Mampu	8	8	0	0	
Kurang Mampu	5	5	0	0	
INISIATIF					0
Bagus	11	11	0	0	
Cukup	1	1	0	0	
Kurang	1	1	0	0	

Pada tabel diatas merupakan perhitungan terakhir dari algoritma C4.5 karena tidak ditemukan nilai tertinggi dari *Gain* pada atribut Kerjasama dan Inisiatif, sehingga klasifikasi akar terakhir dari pohon keputusan yaitu atribut Kedisiplinan (lihat tabel 3.).

Penggunaan Software RapidMiner

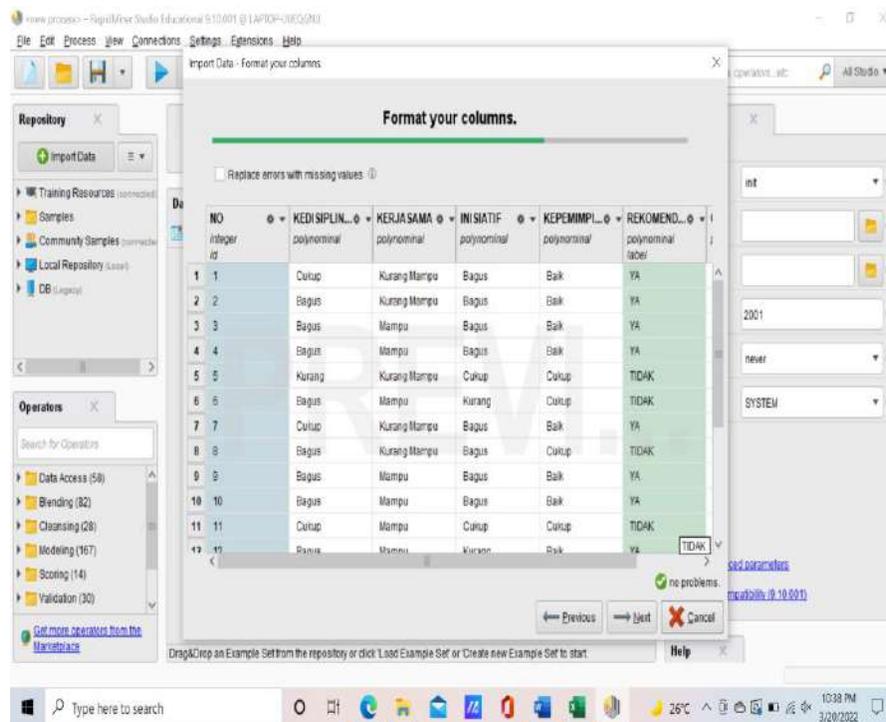
Pada tahap ini akan dilakukan pemrosesan *decision tree* dengan menggunakan *software RapidMiner Studio 9.10* dengan metode *decision tree*. Data yang diolah menggunakan *MS. Excel 2019*, sehingga di aplikasi *RapidMiner Studio 9.10* menggunakan operator *read excel* yang berfungsi untuk membaca file yang akan diolah. Kemudian masukan operator *Decision Tree* untuk membuat data yang diolah menghasilkan *decision tree*.



Gambar 3. Tampilan Work Area dari RapidMiner

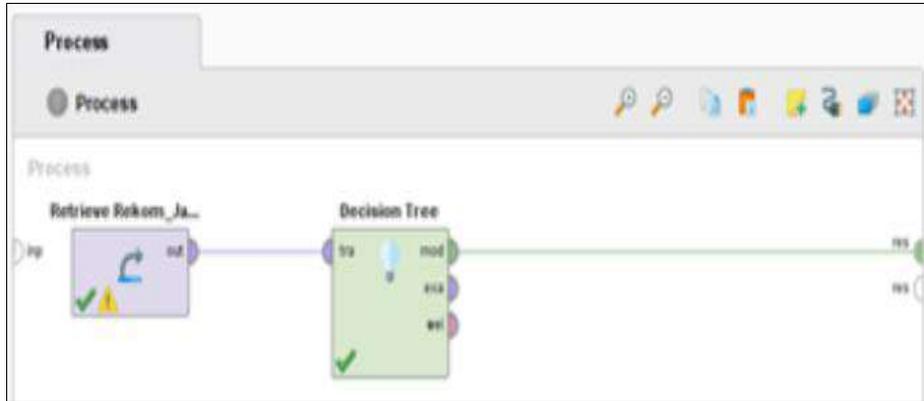
Example Set

Example set adalah data set yang sudah terupload di aplikasi RapidMiner Studio 9.10. Berikut adalah data set yang siap untuk diproses dengan aplikasi RapidMiner Studio 9.10.



Gambar 4. Example Set

Langkah berikutnya yang akan dilakukan yaitu melihat hasil pengujian, dimana nantinya proses akan dimulai dari koneksi basis data, operator rapidminer dan pohon keputusan yang dihasilkan, untuk lebih jelasnya dapat dilihat pada gambar berikut.

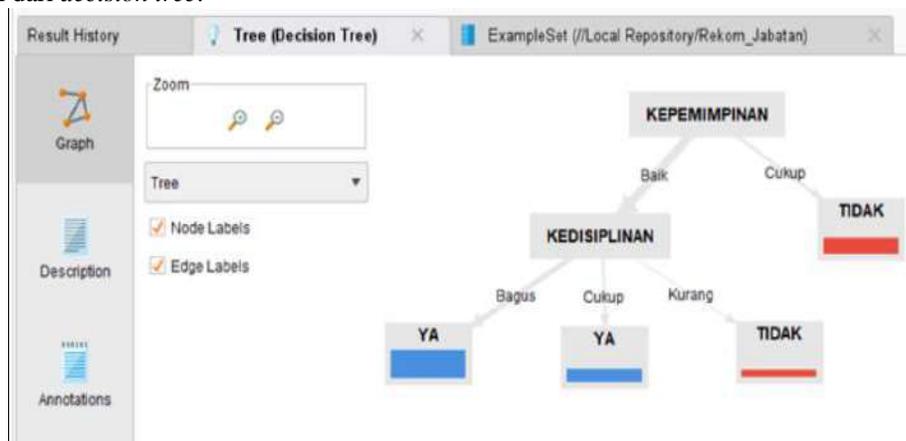


Gambar 5. Koneksi Proses *RapidMiner*

Proses ini dilanjutkan dengan menjalankan (*run*) dan dapat kita lihat hasil pohon keputusan yang dibentuk dan sama dengan hasil perhitungan manual yang telah dilakukan diatas.

Graph View

Graph view menunjukkan hasil dari klasifikasi dengan percabangan yang dapat dihasilkan kesimpulan. Berikut adalah hasil dari *decision tree*.



Gambar 6. Pohon Keputusan

Dari pohon keputusan diatas maka pada *RapidMiner* di dapat diperoleh aturan-aturan (*rules*) sebagai berikut:



Gambar 7. *Rules* Yang Didapat

Hasil Akhir

Aturan-aturan (*rules*) yang didapat dari hasil pohon keputusan dan deskripsi *tree* diatas untuk



menganalisa faktor rekomendasi jabatan berdasarkan atribut adalah:

1. Jika KEPEMIMPINAN = Cukup maka REKOMENDASI = Tidak.
2. Jika KEPEMIMPINAN = Baik, KEDISIPLINAN = Bagus, maka REKOMENDASI = Ya.
3. Jika KEPEMIMPINAN = Baik, KEDISIPLINAN = Cukup, maka REKOMENDASI = Ya.
4. Jika KEPEMIMPINAN = Baik, KEDISIPLINAN = Kurang, maka REKOMENDASI = Tidak.

Dari hasil percobaan diatas untuk menganalisa faktor **Rekomendasi** kenaikan jabatan maka dapat disimpulkan faktor yang menjadi penyebab utama karyawan di rekomendasikan untuk kenaikan jabatan adalah **Kepemimpinan** dan **Kedisiplinan**.

4. KESIMPULAN

Sistem pendukung keputusan yang dibuat dapat digunakan untuk membantu pengambilan keputusan penentuan rekomendasi kenaikan jabatan karyawan dengan menggunakan data karyawan dengan beberapa indikator kinerja karyawan, yaitu : Kedisiplinan, Kerjasama, Inisiatif, dan Kepemimpinan.

Dari hasil percobaan dengan menggunakan perangkat lunak RapidMiner dalam penerapan terhadap masalah penentuan rekomendasi kenaikan jabatan karyawan, *decision tree* dengan algoritma C4.5 dapat memberikan hasil yang lebih baik dan lebih mudah dibandingkan perhitungan secara manual.

Untuk meningkatkan kinerja dan menyempurnakan sistem pendukung keputusan yang telah dibuat, diharapkan dibuatnya sistem pendukung keputusan lainnya sehingga proses pengambilan keputusan semakin mudah dan cepat. Peneliti berharap agar hasil penelitian ini bisa dikembangkan dan dilengkapi lagi agar dapat digunakan sebagai sarana untuk rekomendasi kenaikan jabatan karyawan bagi perusahaan.

REFERENCES

- [1] Dwijaya, I., F. (2010). Sistem Pendukung Keputusan Kenaikan Jabatan pada PT. Sysmex menggunakan Metode Profile Matching. Skripsi, Fakultas Teknik dan Ilmu Komputer Jurusan Teknik Informatika, Bandung: Universitas Komputer Indonesia.
- [2] Muqtadir, A., Purdianto, I. (2013). Sistem Pendukung Keputusan Kenaikan Jabatan Menggunakan Metode Profile Matching (Studi Kasus di PT. Industri Kemasan Semen Gresik). Yogyakarta : Seminar Nasional Aplikasi Teknologi Informasi (SNATI).
- [3] Hidayat, A. L., Pinandita, T. (2013). Sistem Pendukung Keputusan Evaluasi Kinerja Karyawan untuk Promosi Jabatan Struktural pada Bimbingan Belajar SCIENCEMASTER Menggunakan Metode GAP Kompetensi (Profile Matching). Jurnal Teknologi Technocientia, Vol. 5, No. 2, Hal 211-220.
- [4] Suyanto. (2007). Artificial Intelligence. Informatika Bandung, pp. 139.
- [5] Berry, Michael J.A. & Gordon S. Linoff. (2004). Data Mining Techniques For Marketing, Sales, Costumer, Relationship Managemen. Second Edition. Wiley Publishing, Inc.
- [6] Kusriani, & Emha Taufiq Luthfi (2009). Algoritma Data Mining. Yogyakarta: Andi Publisher.
- [7] Turban, E., Aronson, J. E., & Liang, T. P. (2005). Decision Support System. 7th Edition.
- [8] Nemati, H. R., Steiger, D. M., Iyer, L. S., & Herschel, R. T. (2002). Knowledge warehouse: an architectural integration of knowledge management, decision support, artificial intelligence and data warehousing. Decision Support Systems, 33(2), 143-161.
- [9] Sunarko, D., & Pakaja, F. (2009). Study Decision Tree/Decision tree Sebagai Sebuah Alat Bantu Pendukung Sistem Dalam Proses Pengambilan Keputusan Penjualan Pada CV. Khan Setia Utama. Pondok Cabe Depok. Jurnal Ilmiah Teknologi Informasi Asia, 3(2), pp. 51-69.
- [10] Tahir, M. A. (2019). Perancangan Aplikasi Data Mining Menggunakan Metode Decision Tree Untuk Analisa Pemberian Kredit Pada BRI Unit Lalabata Rilau. Jurnal Ilmiah Sistem Informasi dan Teknik Informatika "JISTI", 2(1), pp. 1-10.
- [11] Ginting, S. L. B., Zarman, W., & Hamidah, I. (2014). Analisis Dan Penerapan Algoritma C4.5 Dalam Data Mining Untuk Memprediksi Masa Studi Mahasiswa Berdasarkan Data Nilai Akademik. Prosiding SNAST.
- [12] Aprillia, D., Baskoro, D. A, Ambarwati, L., dan Wicaksana, I. W. S. (2013). Belajar Data Mining dengan RapidMiner. Jakarta : Gramedia Pustaka Utama.

Rancang Bangun Aplikasi E-Commerce Penjualan Souvenir Berbasis Android

Junaidi

Fakultas Teknik, Sistem Informasi, Universitas Krisnadwipayana, Jakarta, Indonesia

E-mail: junaidi@unkris.ac.id

Abstrak- Indonesia memiliki berbagai jenis cinderamata dari berbagai daerah. Cinderamata merupakan ciri khas daerah asal yang mungkin tidak dapat diperoleh di daerah lain namun tidak menutup kemungkinan untuk diperoleh di daerah lain. Untuk mendapatkan oleh-oleh dari suatu daerah kita mungkin perlu pergi ke daerah tersebut untuk mendapatkannya, hal ini menjadi salah satu pertimbangan untuk mendapatkan oleh-oleh khas dari suatu daerah karena jarak yang harus ditempuh dan berapa biaya yang harus dikeluarkan untuk menuju ke tempat tersebut. daerah. Dengan adanya teknologi E-Commerce dapat membantu mengatasi kendala jarak dan biaya untuk mendapatkan oleh-oleh khas dari suatu daerah. Pada umumnya penggunaan teknologi E-Commerce digunakan untuk menjual kebutuhan sehari-hari seperti sepatu, makanan, sepeda, dll namun terkadang juga menjual souvenir. Di beberapa situs jual beli souvenir online, hanya sedikit atau tidak ada yang menjual souvenir. Dari permasalahan tersebut maka dibuatlah sebuah aplikasi E-Commerce berbasis android yang nantinya akan membantu dalam proses penjualan oleh-oleh dari setiap nusantara. Pada aplikasi ini juga akan diimplementasikan sebuah algoritma pencarian yaitu algoritma sekuensial.

Kata Kunci: Android, Cinderamata, E-Commerce, Algoritma Sekuensial.

Abstract- Indonesia has various types of souvenirs from various regions.. The souvenir is a characteristic of the area of origin which may not be able to be obtained in other areas but does not close the possibility to be obtained in other areas. To get a souvenir from an area we might need to go to the area to get it, this is one of the considerations to get a unique souvenir from an area because of the distance that must be traveled and how much it costs to go to the area. With the existence of E-Commerce technology can help overcome distance and cost constraints to get a typical souvenir from an area. In general, the use of E-Commerce technology is used to sell daily necessities such as shoes, food, bicycles, etc. but sometimes also sells souvenir. In some online souvenir buying and selling sites, there are only a few or no one selling souvenir. From these problems, an Android-based E-Commerce application was made which later will help in the process of selling souvenirs from each archipelago. In this application also will implement a search algorithm, namely sequential algorithm.

Keywords: Android, Souvenir, E-Commerce, Sequential Algorithm.

1. PENDAHULUAN

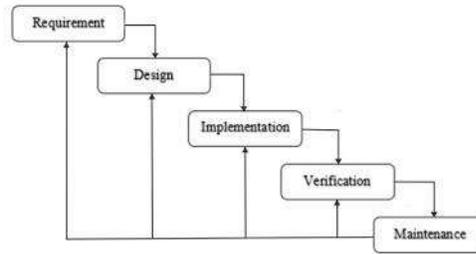
Indonesia memiliki berbagai jenis cinderamata dari berbagai daerah. Cinderamata tersebut merupakan ciri khas dari daerah asalnya yang mana mungkin tidak dapat diperoleh didaerah lain akan tetapi tidak menutup kemungkinan untuk didapatkan didaerah lain.

Untuk mendapatkan *souvenir* dari suatu daerah mungkin kita perlu pergi ke daerah tersebut untuk mendapatkannya, hal tersebut merupakan salah satu pertimbangan untuk mendapat/memiliki *souvenir* khas dari suatu daerah karena jarak yang harus ditempuh dan berapa besar biaya yang harus dikeluarkan untuk pergi ke daerah tersebut.

Dengan adanya teknologi *E-Commerce* dapat membantu mengatasi batasan jarak dan biaya untuk mendapatkan *souvenir* khas dari suatu daerah. Menurut Laudon (2012:49), *E-Commerce* adalah suatu proses membeli dan menjual produk-produk secara elektronik oleh konsumen dan dari perusahaan ke perusahaan dengan komputer sebagai perantara transaksi bisnis. Sedangkan menurut David Baum (2000:2), *E-Commerce* adalah serangkaian teknologi, aplikasi, dan proses bisnis dinamis yang menghubungkan perusahaan, konsumen, dan komunitas melalui transaksi elektronik dan pertukaran barang, layanan, dan informasi elektronik.

2. METODOLOGI PENELITIAN

Metode yang diterapkan dalam penelitian ini adalah model *Waterfall*. Tahapan dari metode *waterfall* adalah sebagai berikut :



Gambar 1. Tahapan Metode Waterfall

1. Requirement Analysis

Tahap ini pengembang sistem diperlukan komunikasi yang bertujuan untuk memahami perangkat lunak yang diharapkan oleh pengguna dan batasan perangkat lunak tersebut.

2. System Design

Spesifikasi kebutuhan dari tahap sebelumnya akan dipelajari dalam fase ini dan desain sistem disiapkan.

3. Implementation

Pada tahap ini, sistem pertama kali dikembangkan di program kecil yang disebut *unit*, yang terintegrasi dalam tahap selanjutnya.

4. Integration & Testing

Seluruh *unit* yang dikembangkan dalam tahap implementasi diintegrasikan ke dalam sistem setelah pengujian yang dilakukan masing-masing *unit*.

5. Operation & Maintenance

Pemeliharaan termasuk dalam memperbaiki kesalahan yang tidak ditemukan pada langkah sebelumnya.

E-Commerce

E-Commerce merupakan suatu transaksi saling tukar menukar barang antar satu dengan yang lainnya untuk memenuhi kebutuhan sehari-hari secara digital [1]. Penggunaan e-commerce pada saat ini merupakan syarat bagi sebuah organisasi atau perusahaan, agar perusahaan itu dapat bersaing secara global [2]. Pandangan populer dari *E-Commerce* adalah penggunaan internet dan komputer dengan browser Web untuk membeli dan menjual produk.

E-Commerce atau kependekan dari elektronik *commerce* (perdagangan secara elektronik), merupakan transaksi bisnis yang terjadi dalam jaringan elektronik, seperti internet. Siapapun yang dapat mengakses komputer, memiliki sambungan ke internet, dan memiliki cara untuk membayar barang-barang atau jasa yang mereka beli, dapat berpartisipasi dalam *E-Commerce*.

Jadi pengertian *E-Commerce* adalah proses transaksi jual beli yang dilakukan melalui internet dimana *website* digunakan sebagai wadah untuk melakukan proses tersebut.

Android

Android merupakan perangkat bergerak pada sistem operasi untuk telepon seluler yang berbasis *linux* [3]. *Android* merupakan OS (*Operating System*) *Mobile* yang tumbuh ditengah OS lainnya yang berkembang dewasa ini [4]. OS lainnya seperti *Windows Mobile*, *i-Phone OS*, *Symbian*, dan masih banyak lagi. Akan tetapi, OS yang ada ini berjalan dengan memprioritaskan aplikasi inti yang dibangun sendiri tanpa melihat potensi yang cukup besar dari aplikasi pihak ketiga. Oleh karena itu, adanya keterbatasan dari aplikasi pihak ketiga untuk mendapatkan data asli ponsel, berkomunikasi antar proses serta keterbatasan distribusi aplikasi pihak ketiga untuk *platform* mereka.

Algoritma Sequential

Algoritma Sequential Searching Sequential Search adalah proses membandingkan setiap elemen array satu persatu secara beruntun dimulai dari elemen pertama hingga elemen yang dicari ditemukan atau hingga elemen terakhir dari array[4]. Adalah suatu teknik pencarian data dalam array (1 dimensi) yang akan menelusuri semua elemen-elemen array dari awal sampai akhir, dimana data-data tidak perlu diurutkan terlebih dahulu. Pencarian berurutan menggunakan prinsip sebagai berikut : data yang ada dibandingkan satu per satu secara berurutan dengan yang dicari sampai data tersebut ditemukan atau tidak ditemukan.

Algoritma pencarian berurutan dapat dituliskan sebagai berikut :

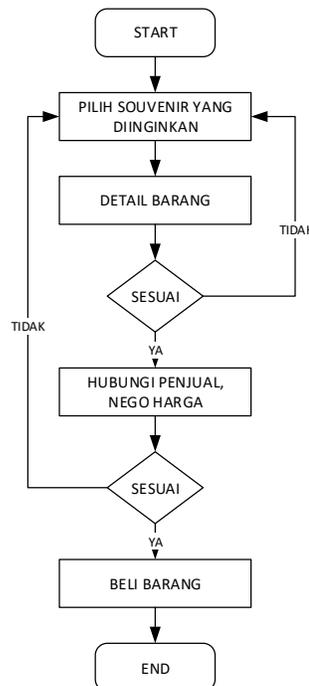
1. $i \leftarrow 0$
2. ditemukan \leftarrow false
3. Selama (tidak ditemukan) dan ($i \leq N$) kerjakan baris
4. Jika ($Data[i] = x$) maka ditemukan \leftarrow true, jika tidak $i \leftarrow i + 1$
 Jika (ditemukan) maka i adalah indeks dari data yang dicari, jika tidak data tidak ditemukan

3. HASIL DAN PEMBAHASAN

Analisis Sistem

Dari hasil analisa sistem berjalan diatas, dapat dibuat sebuah rancangan sistem yang nantinya akan diterapkan pada sebuah aplikasi *e-commerce* penjualan *souvenir* nusantara yang mana analisa sistem yang diusulkan memiliki proses dimana *user* (pembeli) memilih *souvenir* yang diinginkannya, setelah memilih maka *user* (pembeli) dapat mengecek harga dari *souvenir* tersebut. Jika harga dari *souvenir* tersebut tidak cocok maka *user* (pembeli) dapat memilih *souvenir* yang lain dengan harga yang lebih murah, dan jika harga sudah sesuai maka *user* (pembeli) dapat langsung melakukan proses pembayaran. Jika telah melakukan proses pembayaran maka *user* (pembeli) akan menerima bukti pembayaran dari pembelian *souvenir*nya.

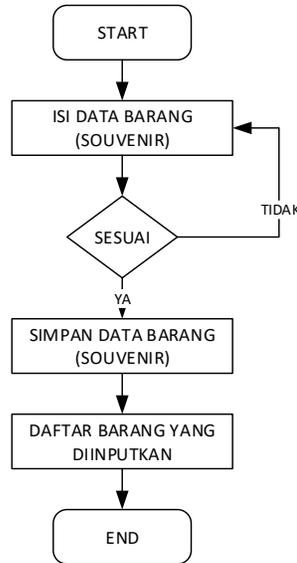
Dari penjabaran analisa sistem usulan untuk pembeli dapat dibuat sebuah *flowchart*, berikut *flowchart* analisa sistem usulan untuk pembeli :



Gambar 1. Rancangan Sistem Usulan-Pembeli

Selain analisa sistem usulan untuk pembeli terdapat pula analisa sistem untuk penjual yang mana prosesnya bahwa *user* (penjual) dapat mengisi data barang (*souvenir*) yang akan dijual olehnya. Dalam prosesnya *user* (penjual) mengisi data barang (*souvenir*) yang akan dijual. Jika data tersebut tidak cocok maka *user* (penjual) dapat memperbaiki data yang diisinya, jika data tersebut sudah cocok maka sistem akan menampilkan daftar barang (*souvenir*) yang telah diinputkan.

Dari penjabaran analisa sistem usulan diatas dibuatlah sebuah *flowchart*, berikut *flowchart* dari analisa sistem usulan untuk penjual :

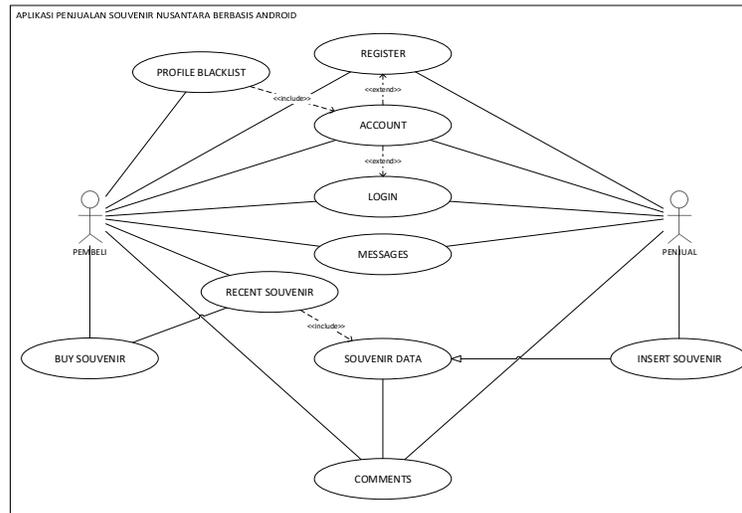


Gambar 2. Rancangan Sistem Usulan-Penjual

Perancangan Sistem

1. Use Case Diagram

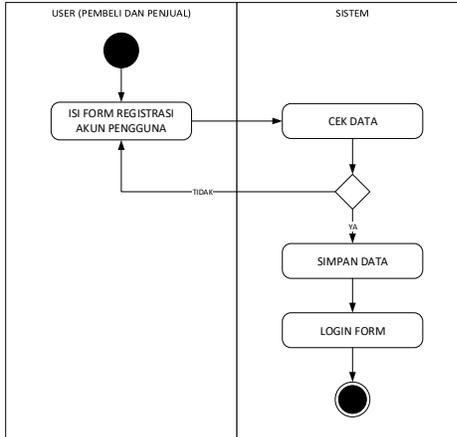
Adapun *Use Case Diagram* yang dibuat untuk menggambarkan fungsi-fungsi apa saja yang tersedia di aplikasi penjualan *souvenir* nusantara ini serta siapa saja yang berhak menggunakan fungsi-fungsi tersebut :



Gambar 3. Use Case Diagram Aplikasi Penjualan Souvenir Nusantara

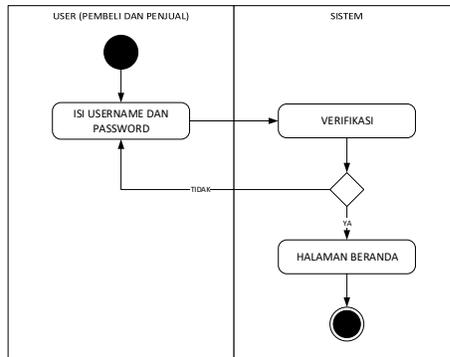
2. Activity Diagram

Dalam gambar 4 *user* (pembeli dan penjual) harus melakukan *register account* sebelum dapat menggunakan aplikasi *mobile e-commerce* penjualan *souvenir* ini. Pada *activity diagram* ini dapat dijelaskan bahwa *user* (pembeli/penjual) melakukan *register*/pendaftaran pada *form* yang telah disediakan, disini *user* tidak perlu melakukan verifikasi akun pengguna.

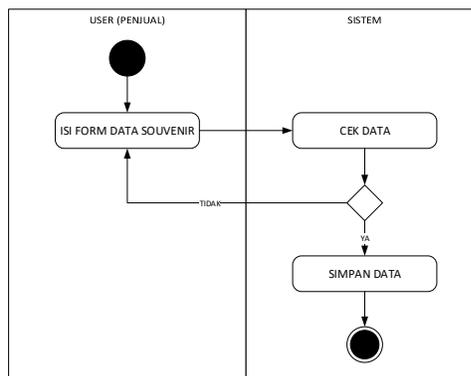


Gambar 4. Activity Diagram Register-Pembeli dan Penjual

Pada gambar dibawah menjelaskan bahwa sebelum dapat menggunakan aplikasi *mobile e-commerce* penjualan *souvenir nusantara* harus melewati proses *log in* pengguna yang ditujukan untuk *user* (pembeli). Jika *username* dan *password* tidak ada (tidak cocok) maka akan menampilkan halaman *log in*, dan jika *username* dan *password* ada (cocok) maka akan menampilkan halaman beranda (daftar *souvenir*).



Gambar 5. Activity Diagram Log In-Pembeli dan Penjual

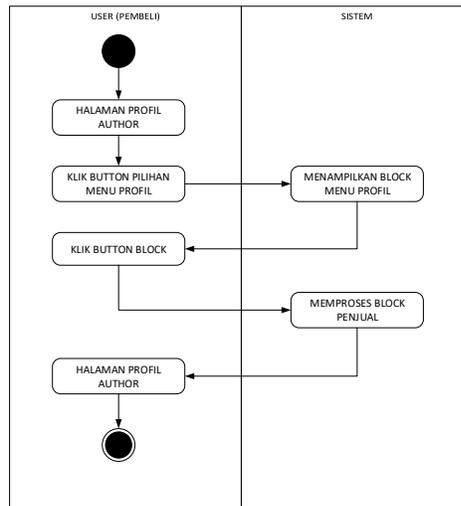


Gambar 6. Activity Diagram Insert Souvenir-Pembeli

Pada gambar diatas menjelaskan bahwa dalam *insert souvenir user* (pembeli) harus mengisi data pada *form insert data souvenir*, setelah *user* mengisi data *souvenir* maka *user* tersebut dapat melakukan rekap data dengan bantuan sistem yang akan mengkoreksi data *inputan* apakah benar atau tidak, jika data sudah benar semua maka *user* dapat menekan *button save* maka secara otomatis sistem akan menyimpan ke data tersebut

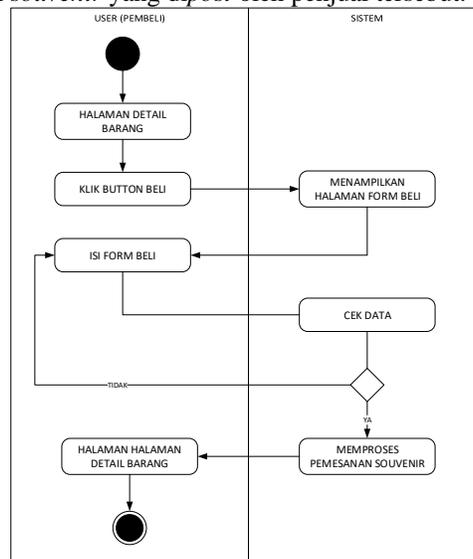


kedalam *database* dan akan menampilkan kembali ke halaman *recent items (recent souvenir)*.



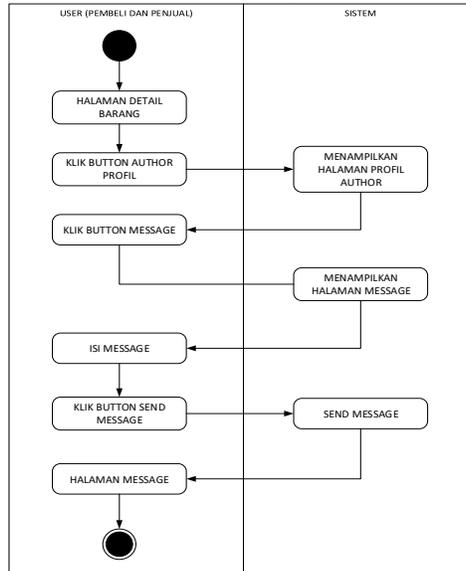
Gambar 7. Activity Diagram Profil Blacklist-Pembeli

Pada gambar diatas menjelaskan bahwa dalam aplikasi memiliki sebuah fitur dapat melakukan *block* penjual untuk tidak ditampilkan *souvenir yang dipost oleh penjual tersebut*.



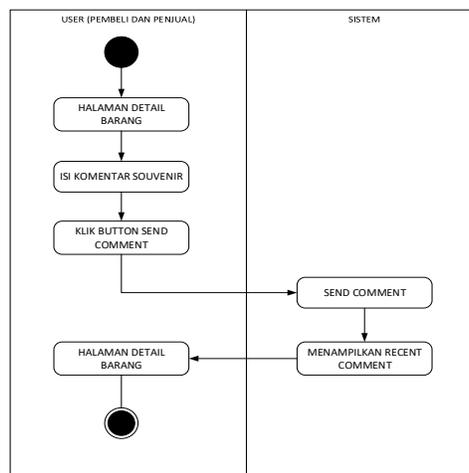
Gambar 8. Activity Diagram Buy Souvenir-Pembeli

Pada gambar diatas dapat disimpulkan bahwa proses dalam pembelian *souvenir* memiliki proses dari *user* (pembeli) dapat mengakses dari halaman detail barang dan menekan *button* beli, maka sistem akan menampilkan halaman *form* pembelian maka *user* dapat mengisi *form* pembelian, jika *form* tersebut telah terisi semua dan menekan *button* lanjut maka sistem akan menampilkan kembali ke halaman detail barang.



Gambar 9. Activity Diagram Messages-Pembeli

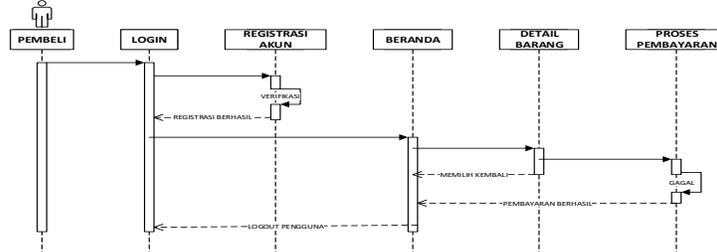
Dari gambar diatas dapat dijelaskan bahwa proses *messages* dari mengakses detail barang sampai tahap *send message*.



Gambar 10. Activity Diagram Comments-Pembeli dan Penjual

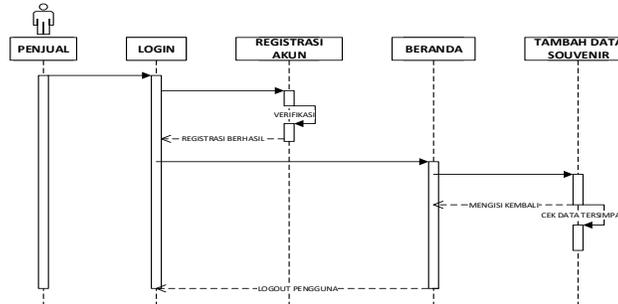
Gambar diatas dapat dijelaskan bahwa proses yang terjadi dari halaman detail barang hingga kembali ke halaman detail barang.

3. Sequential Diagram



Gambar 11. Sequence Diagram Aplikasi Mobile E-Commerce Penjualan Souvenir Nusantara-Pembeli

Pada gambar diatas dapat dijelaskan bahwa proses yang terdapat pada aplikasi tersebut dari *user* (pembeli) melakukan registrasi pengguna hingga logout pengguna. Dalam *sequence diagram* ini terdapat beberapa *object* yang saling berkaitan dengan yang lainnya diantaranya login pengguna, registrasi pengguna, beranda, detail barang, dan proses pembayaran.

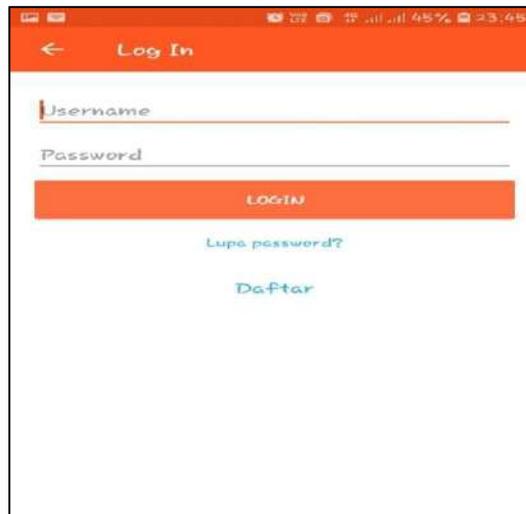


Gambar 12. Sequence Diagram Aplikasi Mobile E-Commerce Penjualan Souvenir Nusantara-Penjual

Pada gambar diatas dapat dijelaskan bahwa proses yang terdapat pada aplikasi tersebut dari *user* (penjual) melakukan registrasi pengguna hingga logout pengguna. Dalam *sequence diagram* ini terdapat beberapa *object* yang saling berkaitan dengan yang lainnya diantaranya login, registrasi akun, beranda, dan tambah data *souvenir*.

Implementasi Dan Pengujian Sistem

1. Halaman Login

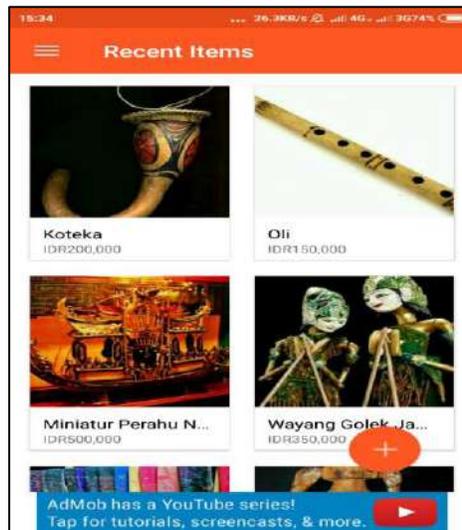


Gambar 13. Halaman Login

Pada halaman login *user* harus mengisi *username* dan *password* terlebih dahulu sebelum dapat menggunakan aplikasi ini. Jika *user* belum memiliki akun maka ia dapat membuatnya dengan memilih tombol daftar yang tersedia pada halaman tersebut. Pada halaman ini juga menyediakan fitur lupa *password* yang mana akan membantu *user* untuk mendapatkan *password*nya kembali.

2. Halaman Beranda

Pada halaman ini *user* disuguhkan dengan *list* dari produk-produk yang ditawarkan, untuk melihat detail produk *user* hanya perlu meng-klik gambar dari produk yang diinginkan maka secara otomatis aplikasi akan menampilkan halaman detail produk. Pada halaman ini juga terdapat beberapa tombol yang mana fungsi dari masing-masing tombol adalah membuka *side-menu* aplikasi dan membuka halaman tambah produk.



Gambar 14. Halaman Beranda

3. Halaman Kategori



Gambar 15. Halaman Kategori

Pada halaman ini *user* disuguhkan list dari kategori produk yang ditawarkan seperti kategori real estate dan mainan & hobi. Produk-produk yang tersimpan pada database akan dikelompokkan berdasarkan kategorinya. Dalam penggunaan aplikasi ini pada halaman kategori *user* hanya perlu meng-klik kategori mana yang diinginkan. Jika sudah memilih kategori produk yang diinginkan maka aplikasi akan menampilkan list dari produk yang ditawarkan berdasarkan kategori produk.

4. Halaman Detail Produk

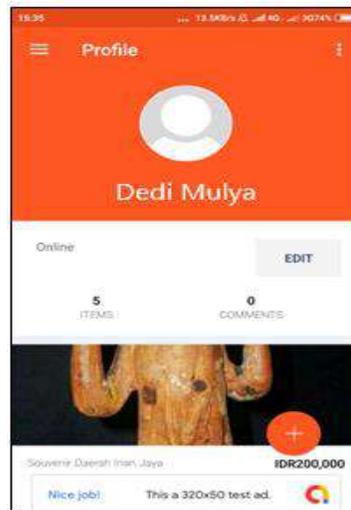
Pada halaman ini aplikasi menampilkan halaman yang berisikan gambar, harga, dan kategori dari sebuah produk. Pada halaman ini terdapat beberapa tombol yang berfungsi untuk melihat profil penjual serta melakukan panggilan ke penjual untuk melakukan nego harga. Pada halaman ini *user* dapat melakukan negosiasi harga dengan melakukan panggilan ataupun melalui fitur komentar.



Gambar 16. Halaman Detail Produk

5. Halaman Profil User

Pada halaman ini *user* akan disuguhkan dengan tampilan profil pengguna yang berisikan foto profil pengguna, status pengguna, jumlah produk yang di *posting*, jumlah komentar yang telah ditulis, serta *list* dari produk-produk yang telah di *posting*. Pada halaman ini terdapat beberapa fitur diantaranya tambah produk, *edit* akun pengguna, ganti foto profil pengguna, serta ganti *cover* latar profil pengguna.



Gambar 17. Halaman Profil User

4. KESIMPULAN

Telah dibuat sebuah aplikasi penjualan *souvenir* Nusantara. Metode penelitian yang digunakan dalam penelitian ini adalah studi literatur, wawancara, dan observasi. Metode pengembangan yang digunakan dalam penelitian ini adalah SWOT (*Strength, Weakness, Opportunities, and Threats*). Dalam perancangan aplikasi menggunakan UML (*Unified Modeling Language*) yang berfungsi untuk proses pengembangan aplikasi berdasarkan *object oriented*.

Dalam penelitian ini juga telah dibuat sebuah rancangan pengujian menggunakan *BlackBox Testing* yang mana



berfungsi untuk mengujikan apakah fungsi-fungsi pada aplikasi sudah sesuai dengan apa yang telah direncanakan.

Pada tahap implementasi sistem telah mengimplementasikan rancangan *interface* dan *database*. Telah dilakukannya pengujian aplikasi dengan menggunakan *BlackBox Testing* untuk melihat apakah fungsi-fungsi dari aplikasi sudah sesuai dengan fungsi yang telah direncanakan.

REFERENCES

- [1] Riswandi, D., (2019). Transaksi On-Line (E-Commerce) : Peluang Dan Tantangan Dalam Perspektif Ekonomi Islam. Jurnal Econetica Vol.1 No.1 Mei 2019
- [2] Alwendi. (2020). Penerapan E-Commerce Dalam Meningkatkan Daya Saing Usaha. Jurnal Manajemen Bisnis Vol. 17, No. 3.
- [3] Teguh, A. (2011), *Membuat Interface Aplikasi Android Lebih Keren Dengan LWIT*, Yogyakarta : Andi Offset.
- [4] Hermawan S., Stephanus, 2011, *Mudah Membuat Aplikasi, Android*, Yogyakarta : Andi Offset.
- [5] Sitorus, Lamhot, (2015). *Algoritma dan Pemrograman*, Andi, Yogyakarta.

ISSN 2807-7849

